GEOMETRY-AWARE DIFFUSION POLICIES FOR ONLINE REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Diffusion models have emerged as a powerful class of expressive policies in Reinforcement Learning (RL). In this paper, we propose an improved diffusion policy method for online RL, where the diffusion process is driven entirely by a learned Q-function and adapts to the geometry of the action-value landscape. Unlike conventional diffusion policies that rely on separately trained score networks, our formulation approximates the score function by the gradient of the Q-function, motivated by the Boltzmann exploration. Furthermore, we introduce a state-dependent preconditioner, motivated by the geometry of Q function, using the covariance of Q-gradients across parallel sampling chains without additional network evaluations, which yields structured-anisotropic exploration grounded in information geometry. The algorithm is evaluated on the Gymnasium benchmarks and attains the highest aggregate normalized return among diffusion reinforcement learning baselines.

1 INTRODUCTION AND RELATED WORKS

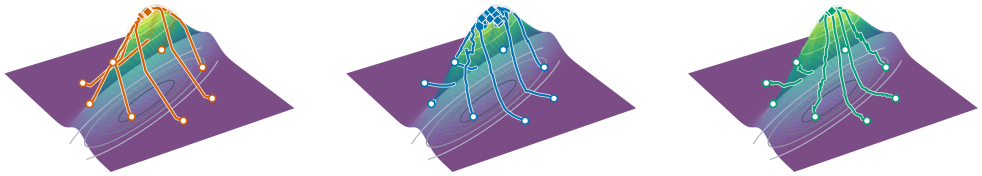
Model-free Reinforcement Learning (RL) has demonstrated remarkable effectiveness in solving nonlinear control problems, ranging from robotic manipulation to autonomous navigation. Within this domain, Entropy-regularized RL (Ziebart et al., 2008; Haarnoja et al., 2018; Geist et al., 2019; Levine, 2018) optimizes the trade-off between exploration and reward maximization by augmenting the return with an entropy regularizer. The optimal policy is a Boltzmann distribution over the action-value function. Practical algorithms such as Soft Actor-Critic (SAC) (Haarnoja et al., 2018) approximate this target with unimodal parametric distributions, such as Gaussian distributions, to model the policy. While computationally efficient, these unimodal approximations exhibit shortcomings in expressiveness when confronted with the multi-modal reward landscapes in complex control tasks.

Diffusion models have emerged as a powerful tool in image generation because of their ability to capture multimodality, such as Denoising Diffusion Probabilistic Models (DDPM) (Ho et al., 2020). Their success extends to policy representation. In imitation learning, where an agent learns a policy from expert demonstrations, Diffusion Policy (Chi et al., 2025) and Diffuser (Janner et al., 2022) represent complex multimodal behaviours through diffusion processes and surpass the classical imitation methods. In the context of offline RL, the expressive power of diffusion models has been utilized not only to clone the behaviour distributions but also to improve the performance of the agents through action value functions (Wang et al., 2022; Hansen-Estruch et al., 2023).

In online reinforcement learning, however, there are no expert demonstrations available as target distributions to sample from. Agents must learn through interactions with the environment, which brings difficulties to traditional diffusion training. To address this, researchers have proposed methods such as Q-score matching (Psenka et al., 2023; Dong et al., 2025), utilizing Q-values to guide the score function (Yang et al., 2023; Ding et al., 2024), employing reweighted score matching (Ma et al., 2025a) based on the change of measure, or direct soft policy sampling based on the noise-conditioned action value functions (Ki et al., 2026) for policy updates. These approaches demonstrate that diffusion models maintain significant advantages even in online RL.

However, standard diffusion samplers and Euclidean Langevin dynamics converge slowly when the target distribution is ill-conditioned with poor convergence when its curvature differs strongly across directions (Zhang et al., 2020), as illustrated in Figure 1. In online RL, the action-value landscape of a learned critic is typically anisotropic when some action dimensions affect the return more than others, and the sampler operates under a small, fixed step budget. Hessian Riemannian Langevin Monte Carlo

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



(a) Standard Diffusion (b) Hessian Riemannian Diffusion (c) Covariance-Preconditioned Diffusion

Figure 1: Comparison of sampling dynamics for various diffusion models on an anisotropic (ill-conditioned) 2D target distribution. (a) Standard Diffusion: Constrained by an isotropic Gaussian prior, trajectories rapidly collapse toward the high-variance axis and exhibit slower convergence characterized by inefficient random walk. (b) Hessian Riemannian Diffusion: Leverages local second-order geometric information. (c) Covariance-Preconditioned Diffusion (ours): uses the cross-chain gradient variance at each noise level as a per-direction step-size normalization.

was introduced by considering Langevin diffusions on a Hessian-type manifold (Zhang et al., 2020; Lobashev et al., 2025), adapting to the local geometry of the log-density. This methodology naturally aligns with the broader framework of Riemannian Diffusion Models (Huang et al., 2022; Betancourt, 2013; De Bortoli et al., 2022), which enables generation directly on Riemannian manifolds. In reinforcement learning, the action value function is parameterized by a Neural Network and the related Boltzmann distributions may exhibit complex non-Euclidean or ill-posed geometries. While Riemannian approaches provide a rigorous geometric foundation for adapting to non-flat manifolds, they often involve intensive computations, which can be prohibitive in high-dimensional action spaces. Preconditioned Diffusion Sampling (PDS) (Ma et al., 2022; 2025b) has emerged as a model-agnostic acceleration framework to alleviate these computational bottlenecks. Similar methods like Elucidating the Design Space of Diffusion Models (EDM) (Karras et al., 2022), Variational Diffusion Models (VDM) (Kingma et al., 2021), and VDM++ (Kingma & Gao, 2023) have explored these SNR schedule preconditioner extensively. Despite this, there is a gap in the literature that few studies have explored the impact of the appropriate preconditioner for the improvement of diffusion policies.

Motivated by this, we investigate whether the geometry of the Q function can improve online diffusion policies. We propose **Geometry-Aware Diffusion Policy** . Our contributions are:

- We leverage the Q-gradient-driven reverse diffusion process, eliminating the actor and score networks, and introduce geometry-aware Langevin sampling to handle with ill-posed Q function in online reinforcement learning.
- We introduce a state-dependent preconditioner for Q-guided diffusion sampling, estimated from the covariance of Q-gradients across parallel sampling chains without requiring additional network evaluations. We relate the preconditioner to Hessian Riemannian sampling and to the Fisher information of the diffused target. We demonstrate the effectiveness of the anisotropic diffusion steps in the Gymnasium environment.

2 Preliminaries

2.1 Reinforcement Learning Foundations

We consider an infinite-horizon discounted Markov Decision Process (MDP) $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the continuous action space, $\mathcal{P}(s'|s, a)$ are the transition dynamics, $r(s, a)$ is the reward function, and $\gamma \in [0, 1)$ is the discount factor. A policy $\pi(a|s)$ maps states to probability distributions over actions. The goal of reinforcement learning is to find a policy π that maximizes the expected cumulative discounted reward $J(\pi) = \mathbb{E}_\pi[\sum_{\tau=0}^{\infty} \gamma^\tau r(s_\tau, a_\tau)]$.

The action-value function $Q^\pi(s,a)$ and the state-value function $V^\pi(s)$ are defined related to the policy π:

$$Q^\pi(s,a) = \mathbb{E}_\pi \left[\sum_{l=0}^{\infty} \gamma^l r(s_{\tau+l}, a_{\tau+l}) \Big| s_\tau = s, a_\tau = a \right] \tag{1}$$

$$V^\pi(s) = \mathbb{E}_{a\sim\pi(\cdot|s)}[Q^\pi(s,a)]. \tag{2}$$

These functions satisfy the Bellman equations. The optimal policy π^* maximizes the action value function at every state and the corresponding optimal action-value function Q^* satisfies the Bellman optimality equation:

$$Q^*(s_\tau, a_\tau) = r(s_\tau, a_\tau) + \gamma \mathbb{E}_{s_{\tau+1}\sim\mathcal{P}(\cdot|s_\tau,a_\tau)}[\max_{a_{\tau+1}} Q^*(s_{\tau+1}, a_{\tau+1})]. \tag{3}$$

2.2 Diffusion Models

Diffusion models are a class of generative models that learn a data distribution $p(x)$ by simulating a continuous-time stochastic process (Song et al., 2020b). The forward process gradually corrupts data $x_0 \sim p(x)$ into standard Gaussian noise over a continuous time variable $t \in [0,1]$. This corruption can be described by a Stochastic Differential Equation (SDE):

$$dx_t = b(t,x_t)dt + g(t)dW_t, \tag{4}$$

where $b(t,x_t)$ is the drift term, $g(t)$ is the diffusion coefficient, and W_t is a standard Wiener process. The reverse-time SDE is given by:

$$dx_t = \left[b(t,x_t) - g(t)^2 \nabla_x \log p_t(x_t) \right] dt + g(t) d\bar{W}_t, \tag{5}$$

where $d\bar{W}_t$ is a standard Wiener process running backward in time. The score function $\nabla_x \log p_t(x_t)$ in Equation 5 points towards high-density regions of the marginal data distribution at time t. Traditional diffusion models train a neural network $s_\theta(x_t,t)$ via denoising score matching to approximate this gradient field.

Riemannian Diffusion Models and Hessian Riemannian Langevin Monte Carlo While standard diffusion models are usually formulated in Euclidean space $\mathbb{R}^d$, many data distributions are supported on a Riemannian manifold $(\mathcal{M}, G)$. In continuous control tasks such as locomotion, certain action dimensions have a much stronger impact on the return than others, leading to highly anisotropic sensitivity of the Q-function, as shown in Figure 1. To respect the underlying geometry, the forward process 4 in terms of the Riemannian Brownian motion is associated to the metric tensor G (De Bortoli et al., 2022). In local coordinates, a geometry-aware Itô diffusion may be written as

$$dX_t^i = \left[b^i(t,X_t) + \frac{g(t)^2}{2} \frac{1}{\sqrt{|G(X_t)|}} \partial_j \left(\sqrt{|G(X_t)|}\, G^{ij}(X_t) \right) \right] dt + g(t)\, \sigma_k^i(X_t)\, dW_t^k, \tag{6}$$

where $G^{ij}(x)$ denotes the entries of the inverse metric $G(x)^{-1}$, $|G(x)| = \det G(x)$, and the matrix $\sigma(x)$ satisfies $\sigma(x)\sigma(x)^\top = G(x)^{-1}$. The second term in the drift is the geometric correction induced by the Laplace–Beltrami operator.

Let p_t denote the density of X_t with respect to the Riemannian volume measure $d\,\mathrm{vol}_G = \sqrt{|G(x)|}\,dx$. The corresponding Riemannian score is defined as $\mathrm{grad}_G \log p_t(x) = G(x)^{-1}\nabla_x \log p_t(x)$, where grad_G is the Riemannian gradient, which is a vector field tangent to $\mathcal{M}$. Then the reverse-time dynamics can be written formally as

$$dX_t = \left[b_t(X_t) - g(t)^2\, \mathrm{grad}_G \log p_t(X_t) \right] dt + g(t)\, d\bar{B}_t^G, \tag{7}$$

where $\bar{B}_t^G$ denotes a reverse-time Riemannian Brownian motion. In local Itô coordinates, this expression must be interpreted together with the corresponding divergence correction induced by the position-dependent diffusion matrix $g(t)^2 G(x)^{-1}$.

A particularly important example of such geometry-aware stochastic dynamics is the Hessian Riemannian Langevin Monte Carlo (HRLMC). Given a target distribution $\pi(x) \propto \exp(-f(x))$, one

defines a position-dependent metric through a strictly convex potential ϕ, $G(x) = D^2\phi(x)$. The HRLMC diffusion is then given by

$$dX_t = \left[\theta(X_t) - G(X_t)^{-1}\nabla f(X_t)\right] dt + \sqrt{2G(X_t)^{-1}}\, dB_t, \tag{8}$$

where the correction drift θ is

$$\theta_i(x) = -\sum_{j,k} G^{ij}(x)\partial_k G^{jk}(x) = -G(x)^{-1}\,\mathrm{Tr}\left(D^3\phi(x)G(x)^{-1}\right), \tag{9}$$

when $G(x) = D^2\phi(x)$. This correction term is necessary for the diffusion to preserve π as its invariant distribution.

From the perspective of diffusion models, HRLMC can be interpreted as replacing the Euclidean score field with a geometry-preconditioned score field,

$$\mathrm{grad}_G \log p_t(x) = [D^2\phi(x)]^{-1}\nabla_x \log p_t(x). \tag{10}$$

Thus, the metric induced by ϕ reshapes the sampling dynamics, producing a preconditioning effect that can improve efficiency in anisotropic landscapes.

Preconditioned Diffusion Models Preconditioned diffusion sampling (PDS) introduces a preconditioning operator into the sampling dynamics without requiring the Hessian whose practical deployment is often hindered by the prohibitive cost of computing (Ma et al., 2025b; 2022). Concretely, the reverse-time dynamics can be modified by replacing the Euclidean score field with a preconditioned version,

$$\nabla_x \log p_t(x) \to MM^\top \nabla_x \log p_t(x),\ d\bar{W}_t \to M d\bar{W}_t \tag{11}$$

while simultaneously injecting anisotropic noise with $MM^\top$. A key theoretical result is that, as long as the preconditioning matrix M is invertible, this transformation preserves the target distribution of the diffusion process. Instead of relying on local curvature information, PDS constructs M using statistics of the data distribution to reshape the second-order structure (i.e., second moments) of the sampling updates, effectively balancing directions in the sample space that exhibit anisotropic scales. In practice, this can be achieved using simple data-dependent filters that approximate the covariance structure of the data, thereby mitigating the ill-conditioning of the diffusion dynamics.

3 Problem Setting: Entropy-Regularized Policy Optimization

In online Reinforcement Learning, an entropy term is introduced to balance action value maximization and stochastic exploration,

$$J(\pi) = \mathbb{E}_\pi\left[\sum_{\tau=0}^{\infty}\gamma^\tau(r(s_\tau, a_\tau) + \alpha\mathcal{H}(\pi(\cdot|s_\tau)))\right], \tag{12}$$

where $\alpha > 0$ is the temperature. The policy improvement step minimizes the KL divergence to the exponentiated Q-function,

$$\pi_{\text{new}} = \arg\min_\pi D_{\text{KL}}\left(\pi(\cdot|s)\middle\|\frac{\exp(Q(s,\cdot)/\alpha)}{Z(s)}\right), \tag{13}$$

whose solution is the Boltzmann distribution $\pi(a|s) \propto \exp(Q(s,a)/\alpha)$. The Q-function is trained by a standard Bellman update.

Our objective is to sample from $p_0(a|s) \propto \exp(Q(s,a)/\alpha)$ induced by the learned critic, using a diffusion sampler, without training an explicit policy network. The clean score of this target is available in closed form,

$$\nabla_a \log p_0(a|s) = \frac{1}{\alpha}\nabla_a Q(s,a), \tag{14}$$

where $Q(s,\cdot)$ is continuously differentiable. The following method applies to any target of the form $\exp(Q/\alpha)$, regardless of how Q is trained.

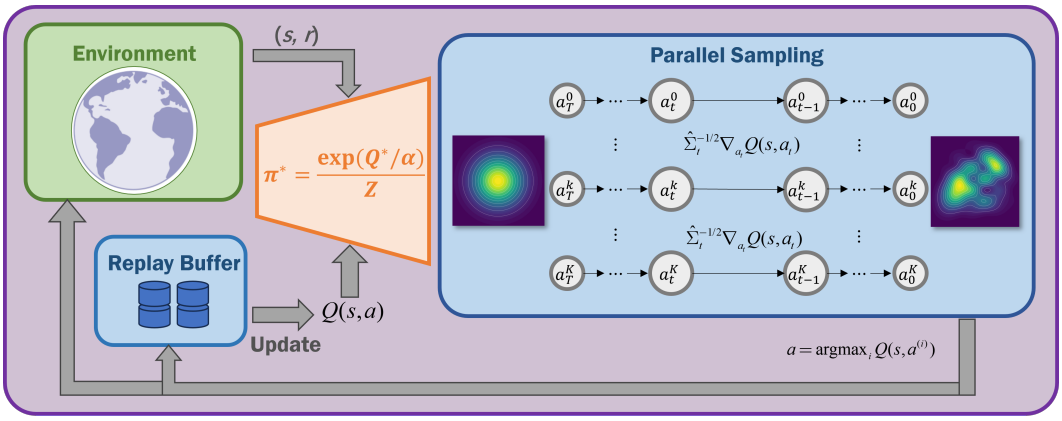


Figure 2: **Overview of Geometry-Aware Diffusion Policies.** This architecture replaces traditional score networks with the gradient of a learned Q-function to drive parallel action sampling. A covariance-based preconditioner ($\hat{\Sigma}_t^{-1/2}$), estimated across the parallel chains at each noise level, adapts per-direction step sizes to the anisotropy of the action-value landscape, yielding structured anisotropic action sampling.

4 Methodology

4.1 Policy Optimization via Q-Guided Diffusion

Our sampler draws actions from the target $p_0(a|s) \propto \exp(Q(s,a)/\alpha)$ by simulating a reverse diffusion process, with no separately trained score network. With the DDPM forward kernel $q_t(a_t|a_0) = \mathcal{N}(a_t; \sqrt{\bar{\alpha}_t}a_0, (1-\bar{\alpha}_t)I)$, the intermediate marginal is the convolution $p_t(a_t|s) = \int q_t(a_t|a_0)p_0(a_0|s)\,da_0$, whose exact score is, by Tweedie's formula,

$$s_t^*(s,a_t) = \nabla_{a_t}\log p_t(a_t|s) = \frac{\sqrt{\bar{\alpha}_t}\,\mathbb{E}[a_0 \mid a_t, s] - a_t}{1-\bar{\alpha}_t}. \tag{15}$$

Computing s_t^* requires an intractable posterior expectation. Following the (Psenka et al., 2023), we instead use the clean score equation 14 evaluated at the noisy action,

$$\widehat{s}_t(s,a_t) = \frac{1}{\alpha}\nabla_{a_t}Q_\phi(s,a_t) \;\approx\; s_t^*(s,a_t). \tag{16}$$

Equation 16 is an approximation, and we make the following assumption.

Assumption 4.1 (Noisy-action score accuracy) *For each diffusion level t there exists $\varepsilon_t \geq 0$ such that $\mathbb{E}_{s\sim\rho,\ a_t\sim p_t(\cdot|s)}\left[\|\widehat{s}_t(s,a_t) - s_t^*(s,a_t)\|_2^2\right] \leq \varepsilon_t^2$.*

This implies critic gradient error at noisy actions, which is nontrivial because the TD regression only sees executed clean actions and relies on generalization elsewhere where broader action coverage during training reduces it. Under local smoothness of Q it vanishes as the noise level $1-\bar{\alpha}_t \to 0$, so equation 16 is most accurate at the late, low-noise steps that determine the final sample, and least accurate at early high-noise steps where the gradient only needs to supply a coarse direction toward high-value regions. The diagnostics of the approximations is provided in Appendix E.

The direct gradient may be ill-conditioned when the critic landscape has different curvature across action directions. To address this we consider a Hessian–Riemannian formulation and use it to motivate a tractable surrogate.

4.2 Hessian Riemannian Diffusion for Q-Guided Policy Sampling

To capture the local geometry of the target distribution, we first consider a Hessian–Riemannian diffusion in action space. Let the metric be the Hessian of a strictly convex potential ϕ_s; for the Boltzmann target the natural choice is $\phi_s(a) = -Q(s,a)$, so that $G_s(a) = \nabla_a^2\phi_s(a) = -\nabla_a^2 Q(s,a)$.

Assumption 4.2 (Local strong concavity) *Within a neighbourhood $\mathcal{U}$ of the current action, $Q(s,\cdot) \in C^3$ and $mI \preceq -\nabla_a^2 Q(s,a) \preceq LI$ for constants $0 < m \le L < \infty$.*

Assumption 4.2 makes G_s positive definite and invertible, so it is a valid Riemannian metric on $\mathcal{U}$. Substituting the proxy score equation 16 into the Riemannian reverse dynamics equation 7 with $\mathrm{grad}_G \log p_t = G_s^{-1} \nabla \log p_t$ gives

$$da_t \approx \Big[b_G(t,a_t) - g(t)^2 \frac{1}{\alpha} G_s(a_t)^{-1} \nabla_{a_t} Q(s,a_t) \Big] dt + g(t) G_s(a_t)^{-1/2}\, dB_t, \tag{17}$$

where b_G contains the forward drift and the divergence correction induced by the position-dependent metric. Equation 17 is a curvature-preconditioned score field which it takes conservative steps along high-curvature directions and larger steps along flat ones. Detailed derivation is provided in Appendix D.

The Hessian metric is impractical in the sampler. Although conceptually appealing, the exact metric requires a Hessian per chain and per step, is unstable to invert where Q is not strongly concave, and does not scale to large neural critics. We therefore do not use equation 17 directly.

4.3 Covariance-based Surrogate as Preconditioner

Our surrogate exploits statistics that the sampler already produces. We run K chains in parallel at a shared noise level and use the spread of their Q-gradients as a curvature proxy. We first give the local argument that motivates the specific functional form.

Consider chains concentrated around a reference action a_0 with isotropic spread at noise level σ_t, $a_t = a_0 + \sigma_t \epsilon$, $\epsilon \sim \mathcal{N}(0, I)$. If Q is locally quadratic (Assumption 4.2), then

$$\nabla_a Q(s,a_t) \approx \nabla_a Q(s,a_0) + \sigma_t H_s(a_0)\epsilon, \qquad H_s(a_0) = \nabla_a^2 Q(s,a_0), \tag{18}$$

so the gradient covariance satisfies

$$\mathrm{Cov}\big[\nabla_a Q(s,a_t)\big] \approx \sigma_t^2 H_s(a_0) H_s(a_0)^\top, \tag{19}$$

and, under local concavity, $|H_s(a_0)|^{-1} = \sigma_t \,\mathrm{Cov}[\nabla_a Q(s,a_t)]^{-1/2}$. This inspires the inverse square root of the gradient covariance as a surrogate for the inverse Hessian metric. It describes refinement inside a single basin, where the chains' positional spread is dominated by the diffusion noise.

On the distribution level, the chains share one matrix per state and noise level, so the estimated object is a property of the target rather than of any individual point. For the smooth density p_t decaying at infinity, we can have

$$\mathrm{Cov}_{p_t}\big[\nabla \log p_t\big] = \mathbb{E}_{p_t}\big[\nabla \log p_t\, \nabla \log p_t^\top\big] = \mathbb{E}_{p_t}\big[-\nabla^2 \log p_t\big] = \mathcal{F}(p_t), \tag{20}$$

the Fisher information of p_t with respect to a location shift, i.e., the expected curvature of the diffused target. Under Assumption 4.1, the cross-chain covariance of $\widehat{s}_t$ is an estimate of $\mathcal{F}(p_t)$. The diagnostics of the approximations is provided in Appendix E.

Estimator and preconditioned dynamics. Let $\{a_t^{(k)}\}_{k=1}^K$ be the K noisy actions at level t for state s. We compute

$$\bar{r}_t = \frac{1}{K}\sum_{k=1}^K \nabla_{a_t} Q\big(s, a_t^{(k)}\big), \qquad \widehat{\Sigma}_t = \frac{1}{K-1}\sum_{k=1}^K \Big(\nabla_{a_t} Q\big(s, a_t^{(k)}\big) - \bar{r}_t\Big)\Big(\nabla_{a_t} Q\big(s, a_t^{(k)}\big) - \bar{r}_t\Big)^\top, \tag{21}$$

and use its diagonal as a lightweight preconditioner in place of the exact inverse metric,

$$da_t = \left[b_{\Sigma_t}(t,a_t) - g(t)^2 \widehat{\Sigma}_t^{-1/2} \frac{1}{\alpha} \nabla_a Q(s,a_t) \right] dt + g(t) \widehat{\Sigma}_t^{-1/4} d\bar{W}_t, \tag{22}$$

where b_{Σ_t} is the drift in the preconditioned coordinates. Appendix C discusses that equation 22 preserves the target distribution under an exact score. To prevent near-singular estimates from producing extreme step-size ratios we use

$$\widehat{\Sigma}_t^{-1/2} \;\leftarrow\; \Big(\widehat{\Sigma}_t + \epsilon I\Big)^{-1/2}, \tag{23}$$

Algorithm 1: Geometry-Aware Diffusion Policy

Input: State s, critic Q_ϕ, target critic $Q_{\bar{\phi}}$, temperature α, diffusion steps T, parallel chains K, critic target update factor ρ, stabilization constant ϵ

Output: Action a_0, updated critic Q_ϕ

for $\tau = 1, 2, \ldots, \mathcal{T}$ **do**
- Observe current state s_τ
- Sample minibatch $(s, a, r, s') \sim \mathcal{D}$
- Obtain the target action a'_0 by running the geometry-aware sampler below at state s' with the target critic $Q_{\bar{\phi}}$
- Update Q_ϕ by minimizing $\mathcal{L}_Q(\phi) = \big(Q_\phi(s,a) - [r + \gamma Q_{\bar{\phi}}(s', a'_0)]\big)^2$
- Update target critic $\bar{\phi} \leftarrow \rho\bar{\phi} + (1-\rho)\phi$
- Initialize K noisy actions $\{a_T^{(k)}\}_{k=1}^K \sim \mathcal{N}(0, I)$
- **for** $t = T, T-1, \ldots, 1$ **do**
  - **for** $k = 1, \ldots, K$ **do**
    - $r_t^{(k)} \leftarrow \nabla_a Q_\phi(s_\tau, a_t^{(k)})$
  - $\bar{r}_t \leftarrow \frac{1}{K}\sum_{k=1}^K r_t^{(k)}$
  - $\hat{\Sigma}_t \leftarrow \frac{1}{K-1}\sum_{k=1}^K \big(r_t^{(k)} - \bar{r}_t\big)\big(r_t^{(k)} - \bar{r}_t\big)^\top$
  - $\hat{\Sigma}_t \leftarrow \mathrm{diag}(\hat{\Sigma}_t) + \epsilon I$ (shared by all K chains at this step)
  - **for** $k = 1, \ldots, K$ **do**
    - $a_{t-1}^{(k)} \leftarrow$ Discretization of Eq. 22
- $a_\tau \leftarrow \arg\max_k Q_\phi(s_\tau, a_0^{(k)})$
- Execute a_τ and store $(s_\tau, a_\tau, r_\tau, s_{\tau+1})$ in $\mathcal{D}$

which also implements the graceful fallback discussed above: where the gradient spread is small or uninformative, the update degrades to the isotropic schedule. Intuitively, the covariance at a fixed noise level measures how much the gradient field varies across chains in each direction, and its inverse square root is an adaptive anisotropic step size.

With F the cost of one action gradient, each reverse step costs $O(KF + Kd)$ with the diagonal estimator (the Kd term being covariance construction and application), for a total of $O(T(KF+Kd))$ per sampled action; the preconditioner adds $O(TKd)$ on top of the parallel Q-gradient sampling it reuses and requires no additional network evaluations. The general framework is shown in Figure 2. The pseudocode of the algorithm is shown in Algorithm 1.

5 Numerical Experiments

In this section, we evaluate the performance of Geometry-Aware Diffusion policy on a suite of continuous control tasks. Our experiments are designed to answer the following research questions: (1) How does the proposed method compare with maximum-entropy and online diffusion-based baselines in aggregate return, stability across seeds, and training cost? (2) Is the gain attributable to the covariance preconditioner, as opposed to parallel Q-gradient sampling alone? (3) Does the method exhibit broader state visitation?

5.1 Experimental Setup

5.1.1 Environments

We select several continuous control environments from the MuJoCo suite (Todorov et al., 2012), as provided in Gymnasium (Towers et al., 2024): `Swimmer-v5`, `Ant-v5`, `HalfCheetah-v5`, `Walker2d-v5`, `Hopper-v5`, and `Humanoid-v5` which are standard benchmarks and enable direct comparison with recent online diffusion-RL methods under a common protocol. Detailed experimental settings can be found in Appendix H.

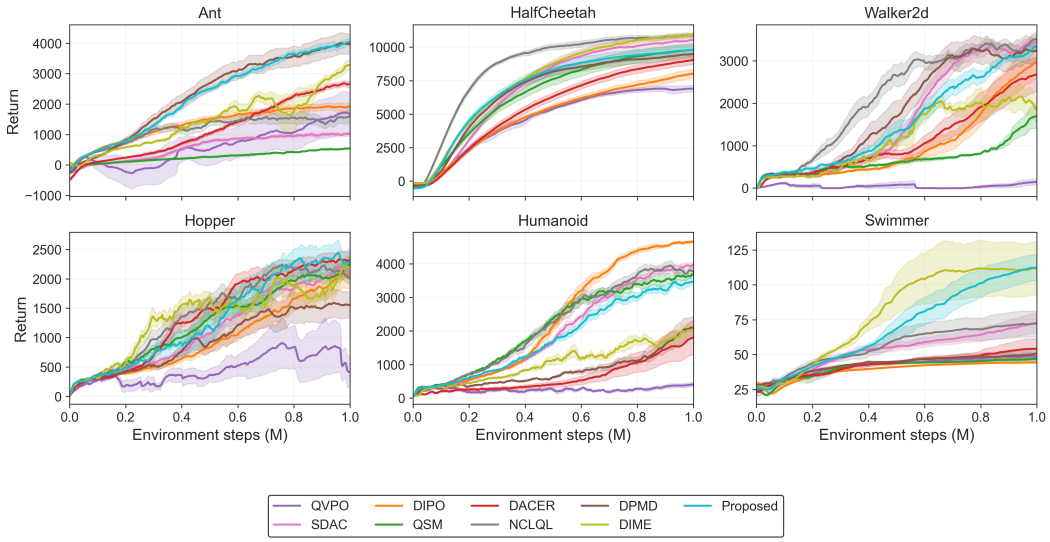


Figure 3: **Training curves of the proposed method and baselines on the continuous control tasks with mean (line) and standard error (shaded).** Specifically, DIPO performs well on Humanoid but fails in Ant; QSM and DACER exhibit lower returns in Ant and Walker2d; NCLQL, despite a strong performance in HalfCheetah, lags behind in Ant. DPMD suffers performance degradation in Humanoid; SDAC struggles to maximize returns in Ant and Swimmer; The results indicate consistently competitive performance.

We compare our proposed algorithm against existing distinct algorithms in the field of Maximum entropy reinforcement learning and diffusion-based reinforcement learning: **DIPO** (Yang et al., 2023), using Q-gradient-guided actions as the target to train the diffusion policies; **QSM** (Psenka et al., 2023), explicitly match the Q-gradient as the score function; **DACER** (Wang et al., 2024), a diffusion actor-critic trained by SAC with entropy approximated by Gaussian; **QVPO** (Ding et al., 2024), Q weighted score matching; **DPMD** and **SDAC** (Ma et al., 2025a), exponential Q reweighted score matching considering the change of measure; **NC-LQL** (Ki et al., 2026), generating actions via Langevin dynamics driven by the gradient of a noise-conditioned Q-function; and **DIME** (Celik et al., 2025), diffusion-based maximum-entropy RL. These recent algorithms represent the state-of-the-art in integrating diffusion policies into the online actor-critic framework.

5.2 Comparative Evaluation

We evaluate the return over 1 million environment steps, aggregated across environments after normalizing each environment's maximum observed return. Following (Agarwal et al., 2021), we report over **ten seeds** the normalized mean, the interquartile mean (IQM), the optimality gap, and stratified-bootstrap 95% confidence intervals (Table 1). Detailed results are provided in Appendix I.

The proposed method attains the highest aggregate normalized return and the smallest optimality gap among the nine algorithms, with the lowest standard error. It is the best method on Ant, Hopper and Swimmer, competitive on Walker2d, HalfCheetah and Humanoid; no baseline is consistently strong across the suite (Table 7). Because the method trains no actor or score network, its model size is $0.66\times$ that of SAC (versus $1.69\times$ for SDAC, DPMD and QSM) and its normalized training time is 1.59 (versus 2.43 for SDAC and 3.74 for NC-LQL), even though sampling requires critic gradients rather than a single forward pass (Additional runtime analysis is provided in Appendix G). Swimmer's reward is dominated by forward velocity and requires precisely coordinated joint torques. Policies are easily trapped in low-return oscillatory modes so that better exploration could contribute to this task. Appendix F further evaluates the capabilities of exploration based on the state visitation metrics and shows the advantages of exploring.

Table 1: Aggregate normalized return over six MuJoCo tasks and ten seeds, with interquartile mean (IQM), optimality gap and stratified-bootstrap 95% confidence intervals (Agarwal et al., 2021).

Algorithm	Mean ± SD [95% CI]	IQM [95% CI]	Opt. Gap [95% CI]
Proposed	$\mathbf{91.6 \pm 18.7}$ [87.3, 95.9]	**91.0** [86.3, 95.7]	**12.4** [9.6, 15.3]
SDAC	77.1 ± 29.8 [73.4, 80.7]	83.9 [78.6, 88.8]	24.9 [21.9, 27.9]
NCLQL	76.4 ± 27.7 [72.0, 81.1]	79.2 [73.0, 85.3]	25.6 [21.7, 29.4]
DIME	74.2 ± 27.7 [68.3, 79.9]	77.1 [67.9, 86.0]	27.1 [21.8, 32.5]
DPMD	72.4 ± 31.3 [66.6, 78.0]	74.5 [66.5, 82.4]	30.1 [25.2, 35.3]
DIPO	72.3 ± 30.2 [67.2, 77.2]	72.0 [64.8, 78.5]	30.5 [26.5, 34.9]
DACER	69.3 ± 32.0 [62.8, 75.7]	73.6 [65.5, 81.6]	32.1 [26.0, 38.4]
QSM	62.5 ± 34.8 [58.5, 66.6]	63.1 [56.7, 69.6]	39.1 [35.4, 42.7]
QVPO	31.5 ± 33.1 [22.6, 39.6]	29.8 [22.2, 37.8]	68.9 [61.2, 77.6]

5.3 Ablation Study: The Impact of the Covariance Preconditioner

To prove that the performance improvement of the proposed method stems from the geometry based preconditioner as opposed to just benefiting from Parallel Q-gradient Sampling, we conduct an ablation study.

We define two explicit algorithmic variants by isolating the preconditioning mechanism and evaluate them on all environments: **w/o preconditioner:** This variant utilizes the exact same parallel sampling to compute the score function $\nabla_{a_t} \log p_t \approx \nabla_{a_t} Q\big(s, a_t^{(k)}\big)$. The sampling SDE uses a standard, cosine variance schedule. **w/ preconditioner:** The complete proposed framework, where the preconditioner is included in the sampling process.

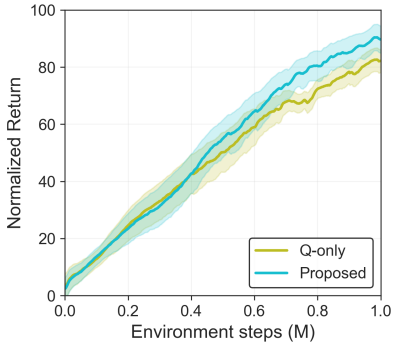


Figure 4: Normalized Return training curve comparisons between with and without preconditioner.

The two variants share the same settings while they differ only in whether equation 22 or the isotropic schedule is used. The preconditioned variant reaches higher normalized aggregated return (Figure 4), so the gap is attributable to the preconditioner alone. Parallel Q-gradient sampling is beneficial but not sufficient when the critic landscape is ill-conditioned. Appendix J reports sensitivity to the number of chains K and steps T for the method.

6 Conclusion, Limitations, and Future Work

We introduced Geometry-Aware Diffusion Policy for online RL: a training-free diffusion sampler driven by the gradient of the learned critic, equipped with a covariance preconditioner estimated from parallel sampling chains. The preconditioner estimates the expected curvature (Fisher information) of the diffused Boltzmann target, is justified locally as a Hessian surrogate inside a basin, preserves the target distribution as a linear preconditioning, and adds no network evaluations. On Gymnasium MuJoCo tasks, it attains the highest aggregate normalized return among all diffusion reinforcement learning baselines with the smallest model, and an ablation attributes the gain to the preconditioner.

Limitations: The current design relies on a diagonal, state-level curvature estimate across chains. Therefore, it cannot capture position-dependent geometry that differs between basins. Additionally, the Q-gradient serves only as an approximate score for the noisy intermediate marginals, and the sampler targets the Boltzmann distribution over the learned critic rather than the exact optimum. Natural extensions include per-chain curvature estimates, full covariance matrices, adaptive diffusion schedules derived from ensemble statistics, offline RL with data-defined priors, and visual control via latent action spaces. A detailed discussion of limitations is in Appendix A.

AI USE STATEMENT

In this work, we used generative AI tools for code assistance and language polishing. AI-assisted code was reviewed and tested by the authors, and AI-assisted text was carefully reviewed and revised to ensure accuracy and consistency with the intended meaning. We take full responsibility for the final content of this work, including all text, claims, and code produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, we provide an anonymized implementation of Q-gradient-based sampling as the supplementary material. A detailed description of the proposed method is provided in Section 4 and the experimental setup is described in Section 5 and Appendix H. We report additional experimental details in Appendix I. Additional ablation studies and analysis are also provided in the Appendix J.

REFERENCES

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron C Courville, and Marc Bellemare. Deep reinforcement learning at the edge of the statistical precipice. *Advances in neural information processing systems*, 34:29304–29320, 2021.

Anurag Ajay, Yilun Du, Abhi Gupta, Joshua Tenenbaum, Tommi Jaakkola, and Pulkit Agrawal. Is conditional generative modeling all you need for decision-making? *arXiv preprint arXiv:2211.15657*, 2022.

Fan Bao, Chongxuan Li, Jun Zhu, and Bo Zhang. Analytic-dpm: an analytic estimate of the optimal reverse variance in diffusion probabilistic models. *arXiv preprint arXiv:2201.06503*, 2022.

Michael Betancourt. A general metric for riemannian manifold hamiltonian monte carlo. In *International conference on geometric science of information*, pp. 327–334. Springer, 2013.

Max Braun, Noémie Jaquier, Leonel Rozo, and Tamim Asfour. Riemannian flow matching policy for robot motion learning. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5144–5151. IEEE, 2024.

Onur Celik, Zechu Li, Denis Blessing, Ge Li, Daniel Palenicek, Jan Peters, Georgia Chalvatzaki, and Gerhard Neumann. Diffusion-based maximum entropy reinforcement learning. In *The Exploration in AI Today Workshop at ICML 2025*, 2025.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

Petros Christodoulou. Soft actor-critic for discrete action settings. *arXiv preprint arXiv:1910.07207*, 2019.

Valentin De Bortoli, Emile Mathieu, Michael Hutchinson, James Thornton, Yee Whye Teh, and Arnaud Doucet. Riemannian score-based generative modelling. *Advances in neural information processing systems*, 35:2406–2422, 2022.

Haoran Ding, Noémie Jaquier, Jan Peters, and Leonel Rozo. Fast and robust visuomotor riemannian flow matching policy. *IEEE Transactions on robotics*, 2025.

Shutong Ding, Ke Hu, Zhenhao Zhang, Kan Ren, Weinan Zhang, Jingyi Yu, Jingya Wang, and Ye Shi. Diffusion-based reinforcement learning via q-weighted variational policy optimization. *Advances in Neural Information Processing Systems*, 37:53945–53968, 2024.

Xiaoyi Dong, Jian Cheng, and Xi Sheryl Zhang. Maximum entropy reinforcement learning with diffusion policy. *arXiv preprint arXiv:2502.11612*, 2025.

Jingliang Duan, Yang Guan, Shengbo Eben Li, Yangang Ren, Qi Sun, and Bo Cheng. Distributional soft actor-critic: Off-policy reinforcement learning for addressing value estimation errors. *IEEE transactions on neural networks and learning systems*, 33(11):6584–6598, 2021.

Matthieu Geist, Bruno Scherrer, and Olivier Pietquin. A theory of regularized markov decision processes. In *International conference on machine learning*, pp. 2160–2169. PMLR, 2019.

Mark Girolami and Ben Calderhead. Riemann manifold langevin and hamiltonian monte carlo methods. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 73(2):123–214, 2011.

Junyu Guo, Zhi Zheng, Donghao Ying, Ming Jin, Shangding Gu, Costas Spanos, and Javad Lavaei. Don't trade off safety: Diffusion regularization for constrained offline rl. *arXiv preprint arXiv:2502.12391*, 2025.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1861–1870. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/haarnoja18b.html.

Philippe Hansen-Estruch, Ilya Kostrikov, Michael Janner, Jakub Grudzien Kuba, and Sergey Levine. Idql: Implicit q-learning as an actor-critic method with diffusion policies. *arXiv preprint arXiv:2304.10573*, 2023.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Chin-Wei Huang, Milad Aghajohari, Joey Bose, Prakash Panangaden, and Aaron C Courville. Riemannian diffusion models. *Advances in Neural Information Processing Systems*, 35:2750–2761, 2022.

Zhiao Huang, Litian Liang, Zhan Ling, Xuanlin Li, Chuang Gan, and Hao Su. Reparameterized policy learning for multimodal trajectory optimization. In *International conference on machine learning*, pp. 13957–13975. PMLR, 2023.

Haque Ishfaq, Guangyuan Wang, Sami Nur Islam, and Doina Precup. Langevin soft actor-critic: Efficient exploration through uncertainty-driven critic learning. *arXiv preprint arXiv:2501.17827*, 2025.

Michael Janner, Yilun Du, Joshua B Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. *arXiv preprint arXiv:2205.09991*, 2022.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022.

Donghyeon Ki, Hee-Jun Ahn, Kyungyoon Kim, and Byung-Jun Lee. Direct soft-policy sampling via langevin dynamics. *arXiv preprint arXiv:2602.07873*, 2026.

Diederik Kingma and Ruiqi Gao. Understanding diffusion objectives as the elbo with simple data augmentation. *Advances in Neural Information Processing Systems*, 36:65484–65516, 2023.

Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. *Advances in neural information processing systems*, 34:21696–21707, 2021.

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

Chunyuan Li, Changyou Chen, David Carlson, and Lawrence Carin. Preconditioned stochastic gradient langevin dynamics for deep neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016.

Alexander Lobashev, Dmitry Guskov, Maria Larchenko, and Mikhail Tamm. Hessian geometry of latent space in generative models. *arXiv preprint arXiv:2506.10632*, 2025.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in neural information processing systems*, 35:5775–5787, 2022.

Cheng Lu, Huayu Chen, Jianfei Chen, Hang Su, Chongxuan Li, and Jun Zhu. Contrastive energy prediction for exact energy-guided diffusion sampling in offline reinforcement learning. In *International Conference on Machine Learning*, pp. 22825–22855. PMLR, 2023.

Haitong Ma, Tianyi Chen, Kai Wang, Na Li, and Bo Dai. Efficient online Reinforcement Learning for Diffusion Policy. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 41837–41853. PMLR, 13–19 Jul 2025a. URL https://proceedings.mlr.press/v267/ma25d.html.

Hengyuan Ma, Li Zhang, Xiatian Zhu, and Jianfeng Feng. Accelerating score-based generative models with preconditioned diffusion sampling. In *European conference on computer vision*, pp. 1–16. Springer, 2022.

Hengyuan Ma, Xiatian Zhu, Jianfeng Feng, and Li Zhang. Preconditioned score-based generative models. *International Journal of Computer Vision*, 133(7):4837–4863, 2025b.

Xiaoteng Ma, Junyao Chen, Li Xia, Jun Yang, Qianchuan Zhao, and Zhengyuan Zhou. Dsac: Distributional soft actor-critic for risk-sensitive reinforcement learning. *Journal of Artificial Intelligence Research*, 83, 2025c.

Sam Patterson and Yee Whye Teh. Stochastic gradient riemannian langevin dynamics on the probability simplex. *Advances in neural information processing systems*, 26, 2013.

Michael Psenka, Alejandro Escontrela, Pieter Abbeel, and Yi Ma. Learning a diffusion model policy from rewards via q-score matching. *arXiv preprint arXiv:2312.11752*, 2023.

Zebang Shen, Alejandro Ribeiro, Hamed Hassani, Hui Qian, and Chao Mi. Hessian aided policy gradient. In *International conference on machine learning*, pp. 5729–5738. PMLR, 2019.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pp. 2256–2265. pmlr, 2015.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020a.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020b.

Jean Tarbouriech, Tor Lattimore, and Brendan O'Donoghue. Probabilistic inference in reinforcement learning done right. *Advances in Neural Information Processing Systems*, 36:33687–33725, 2023.

Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033. IEEE, 2012. doi: 10.1109/IROS.2012.6386109.

Mark Towers, Ariel Kwiatkowski, Jordan Terry, John U Balis, Gianluca De Cola, Tristan Deleu, Manuel Goulão, Andreas Kallinteris, Markus Krimmel, Arjun KG, et al. Gymnasium: A standard interface for reinforcement learning environments. *arXiv preprint arXiv:2407.17032*, 2024.

Yinuo Wang, Likun Wang, Yuxuan Jiang, Wenjun Zou, Tong Liu, Xujie Song, Wenxuan Wang, Liming Xiao, Jiang Wu, Jingliang Duan, et al. Diffusion actor-critic with entropy regulator. *Advances in Neural Information Processing Systems*, 37:54183–54204, 2024.

Zhendong Wang, Jonathan J Hunt, and Mingyuan Zhou. Diffusion policies as an expressive policy class for offline reinforcement learning. *arXiv preprint arXiv:2208.06193*, 2022.

Long Yang, Zhixiong Huang, Fenghao Lei, Yucun Zhong, Yiming Yang, Cong Fang, Shiting Wen, Binbin Zhou, and Zhouchen Lin. Policy representation via diffusion probability model for reinforcement learning. *arXiv preprint arXiv:2305.13122*, 2023.

Jichen Zhang, Liqun Zhao, Antonis Papachristodoulou, and Jack Umenberger. Constrained diffusers for safe planning and control. *arXiv preprint arXiv:2506.12544*, 2025.

Kelvin Shuangjian Zhang, Gabriel Peyré, Jalal Fadili, and Marcelo Pereyra. Wasserstein control of mirror langevin monte carlo. In *Conference on learning theory*, pp. 3814–3841. PMLR, 2020.

Yinan Zheng, Jianxiong Li, Dongjie Yu, Yujie Yang, Shengbo Eben Li, Xianyuan Zhan, and Jingjing Liu. Safe offline reinforcement learning with feasibility-guided diffusion model. *arXiv preprint arXiv:2401.10700*, 2024.

Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, Anind K Dey, et al. Maximum entropy inverse reinforcement learning. In *Aaai*, volume 8, pp. 1433–1438. Chicago, IL, USA, 2008.

A LIMITATIONS

(i) Score approximation. The Q-gradient is the exact score of the clean Boltzmann target but only an approximation to the score of the noisy intermediate marginals (Assumption 4.1); exact target preservation holds in the limit of vanishing approximation error, and our noise-conditioned diagnostic shows moderate, improving agreement rather than equivalence.

(ii) Local geometry. The covariance is one matrix per state and noise level and estimates an expected curvature while it does not represent curvature that differs between basins. The within-basin Hessian interpretation in Sec. 4.3 describes late-stage refinement and does not explain transitions between modes. If chains fall into several basins at low noise, the pooled covariance mixes between-mode statistics and the stabilizer ϵI governs the fallback.

(iii) Diagonal approximation and cost. The diagonal of the covariance is used to simplify the computation while introduce the bias. Sampling requires critic gradients, and target actions for the TD update are produced by the same sampler at s', so per-update cost grows linearly with minibatch size, T and K. Real-time deployment may require fewer steps or distillation.

B RELATED WORKS

This section provides a comprehensive review of the literature related to our proposed method.

B.1 ENTROPY-REGULARIZED REINFORCEMENT LEARNING AND PROBABILISTIC INFERENCE

RL traditionally aims to find a policy that maximizes the expected cumulative reward. Entropy-Regularized RL (Ziebart et al., 2008) extends this objective by incorporating an entropy regularization term, encouraging the agent to explore widely and capturing a distribution of optimal behaviours. This approach lies in the framework of casting RL and control as probabilistic inference (Levine, 2018; Tarbouriech et al., 2023), where the optimal policy naturally emerges as a Boltzmann distribution proportional to the exponentiated Q-function. This theoretical backbone supports actor-critic algorithms like Soft Actor-Critic (SAC) (Haarnoja et al., 2018), with its extension to the discrete action setting (Christodoulou, 2019), and to the distributional reinforcement learning considering the risk-sensitive case (Duan et al., 2021; Ma et al., 2025c). However, practical implementations often parameterize the policy using restricted, unimodal distributions (e.g., isotropic Gaussians). While computationally tractable, these approximations fundamentally limit the agent's expressive capacity when navigating the highly non-convex, multimodal action-value landscapes inherent to complex continuous control tasks (Huang et al., 2023; Celik et al., 2025).

B.2 DIFFUSION MODELS

Diffusion models, initially conceptualized through nonequilibrium thermodynamics (Sohl-Dickstein et al., 2015), have recently emerged as the state-of-the-art framework for generative modeling (Ho et al., 2020). From the perspective of Stochastic Differential Equations (SDEs) (Song & Ermon, 2019; Song et al., 2020b), the reverse sampling process relies on a score function which is the gradient of the log-marginal likelihood. While initially designed for high-dimensional image synthesis, their capability to map simple Gaussian noise to highly complex, multimodal distributions makes them an ideal choice for modeling continuous actions. Recognizing the computational bottleneck of sequential sampling in standard diffusion, subsequent works such as Denoising Diffusion Implicit Models (DDIM) (Song et al., 2020a) and fast ODE solvers like DPM-Solver (Lu et al., 2022) have significantly improved sampling efficiency. These advancements benefit the integration of diffusion processes into online RL.

B.3 DIFFUSION MODELS FOR OFFLINE AND ONLINE RL

Offline Reinforcement Learning. In the offline setting, diffusion models primarily perform as expressive behaviour cloning and sequence planners. Diffuser (Janner et al., 2022) framed trajectory generation as a sequence-level diffusion process, while Decision Diffuser (Ajay et al., 2022) demonstrated that conditional generative modeling could replace traditional imitation learning methods.

Other works, such as Diffusion Policy (Chi et al., 2025), showcased that action diffusion outperforms other imitation learning policies in visuomotor control. To incorporate value function to surpass the expert data, methods like (Wang et al., 2022), IDQL (Hansen-Estruch et al., 2023), and Contrastive Energy Prediction (CEP) (Lu et al., 2023) treat the offline dataset as a prior, utilizing Q-values to guide or reweight the diffusion sampling towards high-reward regions. Safety considerations also emerge as a critical challenge in these approaches (Guo et al., 2025; Zhang et al., 2025; Zheng et al., 2024).

Online Reinforcement Learning. In online RL, the expert demonstrations are not available during training. Recent advancements aim to bridge this gap. QSM (Psenka et al., 2023) matches a parameterized score network to the Q-gradient, while DIPO (Yang et al., 2023) treats Q-gradient-updated actions as pseudo-ground-truth targets. QVPO (Ding et al., 2024) introduces Q-weighted variational policy optimization, and methods like SDAC, DPMD (Ma et al., 2025a) incorporate exponential Q-reweighting to address the change of measure. NC-LQL (Ki et al., 2026) directly performs Langevin dynamics on a noise-conditioned Q-function. (Ishfaq et al., 2025) focuses on improving exploration via stochastic posterior sampling of the critic function.

B.4 Riemannian and Geometry-Aware Diffusion Models

Standard diffusion models assume the target data lies in a flat Euclidean space. However, many real-world distributions reside on complex manifolds. The necessity of adapting stochastic sampling to local geometry was addressed in Stochastic Gradient Riemannian Langevin Dynamics (SGRLD) (Patterson & Teh, 2013). This was extended to generative modeling via Riemannian Diffusion Models (Huang et al., 2022; De Bortoli et al., 2022), which simulate Brownian motion with respect to the Laplace-Beltrami operator to ensure geometric adherence. In the context of energy-based sampling, Hessian Riemannian Langevin Monte Carlo (HRLMC) (Zhang et al., 2020) dynamically constructs a local metric based on the Hessian of a strictly convex potential, accelerating convergence in ill-conditioned regions. Riemannian MALA (Girolami & Calderhead, 2011) uses the Fisher information as the metric, and preconditioned SGLD (Li et al., 2016) uses running gradient second moments. Our preconditioner is estimated per state and noise level from gradient rather than positional statistics and applied inside a diffusion schedule. More recently, similar geometric perspectives have been adopted in robotic trajectory generation, where trajectories are modeled as evolving on underlying Riemannian manifolds (Ding et al., 2025; Braun et al., 2024).

B.5 Preconditioned Diffusion Models

Computing the exact inverse Hessian of a neural Q-function at every sampling step is computationally intractable for online RL (Shen et al., 2019). To mitigate the overhead of exact Riemannian sampling, Preconditioned Diffusion Sampling (PDS) (Ma et al., 2022) was introduced. These methods apply a preconditioning operator to the Euclidean score field, scaling step sizes to match the covariance structure of the target distribution and preventing trajectory collapse. PDS uses a fixed, data-derived covariance. Theoretical investigations, such as Analytic-DPM (Bao et al., 2022), further validate that optimally bounding and estimating the reverse variance is critical for robust generation.

C Invariance of the Target Distribution under Covariance Preconditioning

In this section, we clarify the distributional interpretation of the covariance-preconditioned reverse diffusion used in our method. Our analysis follows the standard reverse-time diffusion framework and is closely related to the linear preconditioning argument of Ma et al. (2025b).

Preconditioned forward process. Let $X_t \in \mathbb{R}^d$ denote the diffusion variable. For each diffusion level t, define an invertible preconditioning matrix

$$M_t := \widehat{\Sigma}_t^{-1/4}, \qquad M_t M_t^\top = \widehat{\Sigma}_t^{-1/2}, \tag{24}$$

where $\widehat{\Sigma}_t$ denotes the regularized covariance estimate defined in Equation 19. For the theoretical argument below, we first regard $\{M_t\}_{t\in[0,1]}$ as a prescribed, state-deterministic matrix schedule.

Consider the preconditioned forward SDE

$$dX_t = b_\Sigma(t, X_t)\, dt + g(t) M_t\, dW_t, \tag{25}$$

where b_Σ denotes the corresponding forward drift. The diffusion matrix of Equation 25 is

$$D_t = g(t)^2 M_t M_t^\top = g(t)^2 \widehat{\Sigma}_t^{-1/2}. \tag{26}$$

Since M_t depends on the diffusion level but not on the sample position, D_t is spatially constant.

Reverse-time dynamics. Let $p_t(x)$ denote the marginal density of the forward process in Equation 25. For a diffusion with spatially independent diffusion matrix D_t, the corresponding reverse-time SDE is

$$dX_t = [b_\Sigma(t, X_t) - D_t \nabla_x \log p_t(X_t)]\, dt + g(t) M_t\, d\bar{W}_t. \tag{27}$$

Substituting Equation 26 gives

$$dX_t = \Big[b_\Sigma(t, X_t) - g(t)^2 \widehat{\Sigma}_t^{-1/2} \nabla_x \log p_t(X_t)\Big]\, dt + g(t) \widehat{\Sigma}_t^{-1/4} d\bar{W}_t. \tag{28}$$

This is the covariance-preconditioned reverse SDE used in the main text. Thus, the same preconditioner that rescales the score field also rescales the reverse diffusion noise in a consistent manner.

Recovery of the target distribution. Equation 28 is the exact reverse-time dynamics of the forward process in Equation 25 when the exact marginal score $\nabla_x \log p_t(x)$ is available. Therefore, integrating the reverse process from the terminal marginal of the forward diffusion recovers its initial distribution. In our implementation, M_t is not specified a priori but is estimated from the gradients of the K parallel sampling chains at each diffusion level. The same estimated matrix is shared by all chains at a given state and diffusion level. The exact invariance statement above therefore applies to a prescribed preconditioning schedule, whereas the finite-chain adaptive estimator used in practice should be viewed as an empirical approximation to this idealized process with an exact score.

D Derivation of equation 17 and Related Assumptions

We derive Eq. equation 17 from the Boltzmann target via the Riemannian reverse dynamics equation 7, making explicit the assumption at each step.

The target is the Boltzmann density on the continuous action space $\mathcal{A}$,

$$p_0(a|s) = \frac{\exp\big(Q(s,a)/\alpha\big)}{Z(s)}, \qquad Z(s) = \int_{\mathcal{A}} \exp\big(Q(s,a)/\alpha\big)\, da, \tag{29}$$

which is well defined whenever $Z(s) < \infty$. The normalizing constant vanishes under differentiation and we have

$$\nabla_a \log p_0(a|s) = \frac{1}{\alpha} \nabla_a Q(s,a). \tag{30}$$

The reverse dynamics, however, require the scores of the diffused marginals $p_t(\cdot|s)$, which are not available in closed form. We replace them by the clean score evaluated at the noisy action,

$$\nabla_{a_t} \log p_t(a_t|s) \;\approx\; \frac{1}{\alpha} \nabla_{a_t} Q(s, a_t). \tag{31}$$

The mean-squared error is bounded by ε_t^2 at level t in Assumption 4.1.

For the geometry, we have $\mathcal{A}$ with the Hessian metric generated by the potential $\phi_s = -Q(s,\cdot)$,

$$G_s(a) = \nabla_a^2 \phi_s(a) = -\nabla_a^2 Q(s,a). \tag{32}$$

In Assumption 4.2, $Q(s,\cdot) \in C^3$ and $mI \preceq G_s \preceq LI$ on $\mathcal{U}$, ensures that G_s is positive definite, invertible, and sufficiently smooth for the metric correction drift equation 9 to be defined. By definition of the Riemannian gradient,

$$\mathrm{grad}_G \log p_t(a_t|s) = G_s(a_t)^{-1} \nabla_{a_t} \log p_t(a_t|s) \;\approx\; \frac{1}{\alpha}\, G_s(a_t)^{-1} \nabla_{a_t} Q(s, a_t), \tag{33}$$

where the equality is exact and the approximation is derived from equation 31.

Substituting equation 33 into the Riemannian reverse-time SDE equation 7 yields

$$da_t \approx \Big[b_G(t,a_t) - g(t)^2 \frac{1}{\alpha} G_s(a_t)^{-1} \nabla_{a_t} Q(s,a_t)\Big]\, dt + g(t)\, G_s(a_t)^{-1/2}\, dB_t, \tag{34}$$

which is equation 17. Here b_G comprises the forward drift together with the divergence correction equation 9 induced by the position-dependent diffusion matrix $g(t)^2 G_s^{-1}$.

Local Gaussian structure and multimodality. Suppose that near a local maximizer μ_j of $Q(s,\cdot)$,

$$Q(s,a) = Q(s,\mu_j) - \tfrac{1}{2}(a-\mu_j)^\top A_j (a-\mu_j) + o\big(\|a-\mu_j\|^2\big), \qquad A_j = -\nabla_a^2 Q(s,\mu_j) \succ 0. \tag{35}$$

Then $\exp(Q(s,a)/\alpha) \propto \mathcal{N}(a;\mu_j, \alpha A_j^{-1})$ up to higher-order terms, and when $Q(s,\cdot)$ has several well-separated maximizers a Laplace approximation gives

$$p_0(a|s) \;\approx\; \sum_j w_j\, \mathcal{N}\big(a;\, \mu_j,\, \alpha A_j^{-1}\big), \tag{36}$$

a mixture that is globally multimodal yet Gaussian in a neighbourhood of each mode. The quadratic argument of Sec. 4.3 therefore justifies curvature-aware refinement conditionally on a chain occupying a given basin; transitions between basins are driven by the annealing of the diffusion and are not explained by this local analysis.

E Diagnostics for the Approximations

Covariance versus Hessian preconditioner. On trained critics we computed, at each reverse step, the exact Hessian preconditioner $|-\nabla_a^2 Q|^{-1}$ and the covariance preconditioner $\widehat{\Sigma}_t^{-1/2}$ on the same chains, and report their cosine similarity (as flattened matrices) and the cosine similarity between the two resulting preconditioned score directions.

Both measures are high throughout and increase from noise toward sample, consistent with the within-basin argument: as chains localize around a peak, the local approximation improves.

Table 2: Alignment between covariance and Hessian preconditioners along the reverse process ($T = 10$; step 9 is pure noise, step 0 the final sample).

Reverse step	0	1	2	3	4	5	6	7	8	9
Preconditioner cosine	0.97	0.96	0.97	0.96	0.95	0.94	0.94	0.93	0.92	0.92
Preconditioned-score cosine	0.98	0.98	0.98	0.97	0.97	0.95	0.95	0.94	0.93	0.93

Standard versus noise-conditioned critic gradient. Following (Ki et al., 2026) we trained an auxiliary noise-conditioned critic $Q_t(s,a_t,t)$ alongside the standard critic and compared $\nabla_{a_t} Q_\phi(s,a_t)$ with $\nabla_{a_t} Q_t(s,a_t,t)$ at identical inputs during training (Table 3). We report the centred normalized RMSE between the two critics at the low-noise diffusion stage. The noise-conditioned critic is itself learned and serves as an empirical reference, not ground truth.

Table 3: Agreement between the standard critic and a noise-conditioned reference critic during training.

Env. steps	100k	200k	300k	400k	500k	600k	700k	800k	900k	1000k
Q NRMSE	0.260	0.222	0.151	0.142	0.128	0.148	0.154	0.130	0.110	0.099

F Exploration metrics evaluation

State-visitation entropy is the differential entropy of the empirical discounted state-occupancy marginal $d^\pi(s)$, estimated per state dimension with a histogram estimator on standardized states over

the final evaluation window, averaged over dimensions, environments and seeds; action entropy is computed analogously on executed actions. Entropy measures breadth of visitation and is sensitive to the estimator and to state scaling; it does not measure whether visited states are reward-relevant.

Table 4: Per-dimension entropy of executed actions and visited states.

	SDAC	DIPO	QSM	DACER	NCLQL	DPMD	DIME	Proposed
Action entropy	0.884	0.818	0.943	0.647	0.763	0.508	0.806	0.847
State entropy	1.050	0.551	1.078	0.612	1.017	0.626	1.068	**1.435**

G Runtime of Target-Action Sampling

Since the TD target action a'_0 is produced by running the diffusion sampler at s' for every transition in the critic minibatch, the per-update cost depends on the minibatch size. Let B be the critic minibatch size, N the number of parallel rollout states, T the number of reverse steps, K the number of chains, d the action dimension, and F the cost of one action-gradient evaluation. With the diagonal covariance estimator, sampling actions for a batch of B states costs

$$C_{\text{sample}}(B) = O\big(BT(KF + Kd)\big), \tag{37}$$

so that rollout sampling costs $C_{\text{rollout}} = O\big(NT(KF + Kd)\big)$ per environment step, while one critic update requires $C_{\text{target}} = O\big(BT(KF + Kd)\big)$ for target-action generation in addition to the ordinary critic forward and backward pass. Target-action sampling therefore grows linearly with B, whereas rollout sampling does not.

We isolated these components empirically on `HalfCheetah-v5` for $B \in \{64, 128, 256\}$ with the default $T = 20$ and $K = 32$ (Table 5). Times are milliseconds per update; the last column is total wall-clock time relative to the default $B = 256$. Rollout sampling is essentially constant in B,

Table 5: Per-update runtime breakdown on `HalfCheetah-v5` as a function of the critic minibatch size B ($T = 20$, $K = 32$).

Batch size B	Rollout (ms)	Target-action (ms)	Target sampler share (%)	Rel. total wall-clock
64	1.225	3.422	85.7	0.38
128	1.229	6.416	92.2	0.58
256	1.266	12.409	95.6	1.00

whereas target-action sampling scales almost linearly with B and accounts for 85.7% to 95.6% of the per-update sampling time. Target-action generation is therefore the primary computational bottleneck of training, and end-to-end wall-clock time grows substantially with larger minibatches.

H Experimental Settings

In this section, we provide the comprehensive experimental setups, network architectures, and hyperparameter configurations used to evaluate our proposed Geometry-Aware Diffusion Policy and the baselines.

H.1 Environments and Evaluation

We evaluate our method on six continuous control tasks from the Gymnasium MuJoCo: `HalfCheetah`, `Walker2d`, `Hopper`, `Ant`, `Humanoid`, and `Swimmer`.

For all environments, we collect samples using parallel environments to accelerate data collection. Consistent with standard online reinforcement learning evaluation protocols, all algorithms are trained for 1 million environment steps. All main results are averaged across 10 independent random seeds; we report normalized mean, IQM, optimality gap and stratified-bootstrap 95% confidence intervals (Agarwal et al., 2021) in Table 1, and mean $\pm$ standard error (shaded regions) in the learning curves. All main experiments were implemented on NVIDIA Tesla V100 GPUs.

Table 6: Hyperparameter settings for Geometry-Aware Diffusion Policy and corresponding shared parameters for the baselines.

Hyperparameter	**Value**
Algorithm & Training	
Optimizer	Adam
Critic learning rate	3×10^{-4}
Replay buffer capacity $\mathcal{D}$	10^6 transitions
Batch size	256
Discount factor γ	0.99
Target network update rate ρ	0.005
Reward scaling factor	0.2
Entropy temperature α	Automatically tuned (target entropy $= -0.9 \times \dim(\mathcal{A})$)
Architecture	
Number of hidden layers	3
Number of hidden units	256
Activation function	`Mish`
Diffusion & Geometry Sampling	
Diffusion Steps T	20
Diffusion noise schedule	Cosine
Parallel sample chains K	32

H.2 Network Architectures

We adopt standard Multilayer Perceptron (MLP) architectures to parameterize both the Q-function (critic) and the score function networks for the diffusion policies. In the proposed methods, the action-value function is parameterized as an MLP with 3 hidden layers, each containing 256 units. We use the `Mish` activation function, as it has been empirically shown to preserve smoother gradients, which is crucial for accurately computing the action-gradient $\nabla_a Q_\phi(s, a)$ required in our sampling process. The target critic network shares the exact same architecture as the primary critic network and is updated via moving average with a factor of $\rho = 0.005$.

For baselines that require explicit score matching networks or parametric actors, we employ equivalent 3-layer MLPs with 256 hidden units and `Mish` activations to ensure a fair comparison regarding model expressivity and capacity.

H.3 Diffusion Process and Hyperparameters

Our diffusion-based sampling relies on a predefined noise schedule. We adopt the Cosine Schedule from standard Denoising Diffusion Probabilistic Models (DDPM). The forward noise variances β_t are defined based on a cumulative product $\bar{\alpha}_t = \prod_{s=1}^{t}(1 - \beta_s)$, which is scheduled as: $\bar{\alpha}_t = \frac{f(t)}{f(0)}, \quad f(t) = \cos^2\left(\frac{t/T+s}{1+s} \cdot \frac{\pi}{2}\right)$ where T is the total number of diffusion steps. We fix the number of diffusion steps $T = 20$ for both our method and all applicable diffusion baselines.

A comprehensive list of hyperparameters used across our experiments is detailed in Table 6. These values were largely aligned with recent state-of-the-art diffusion policy training setups to ensure fairness.

Table 7: Algorithm Performance over 1 million environmental steps and 10 random seeds. (Mean ± SE).

Algorithm	Ant	HalfCheetah	Walker2d
QVPO	1717.0 ± 693.8	6906.1 ± 232.7	143.3 ± 70.8
SDAC	1017.8 ± 67.7	10536.8 ± 478.3	3443.9 ± 169.7
DIPO	1925.3 ± 111.1	8016.6 ± 350.5	2959.2 ± 373.7
QSM	545.8 ± 24.6	9825.1 ± 476.8	1693.9 ± 271.6
DACER	2681.6 ± 84.5	9032.4 ± 319.3	2674.1 ± 409.1
NCLQL	1599.4 ± 207.8	10890.2 ± 181.1	$\mathbf{3512.4 \pm 158.5}$
DPMD	4009.1 ± 348.4	9483.4 ± 319.4	3286.1 ± 310.4
DIME	3293.5 ± 170.6	$\mathbf{10929.0 \pm 172.0}$	2160.0 ± 308.4
Proposed	$\mathbf{4036.0 \pm 111.8}$	9825.5 ± 368.7	3334.5 ± 114.6

Algorithm	Hopper	Humanoid	Swimmer
QVPO	907.0 ± 425.6	406.5 ± 69.0	48.8 ± 2.5
SDAC	2183.8 ± 97.1	3976.0 ± 81.3	72.7 ± 7.0
DIPO	2088.3 ± 237.4	$\mathbf{4666.3 \pm 42.6}$	44.4 ± 0.4
QSM	2243.8 ± 139.6	3689.2 ± 77.0	47.1 ± 1.8
DACER	2331.9 ± 143.4	1799.7 ± 507.0	54.1 ± 7.3
NCLQL	2240.9 ± 85.7	3948.8 ± 144.3	72.1 ± 9.3
DPMD	1591.0 ± 240.3	2105.7 ± 319.1	50.9 ± 4.6
DIME	2208.7 ± 185.4	2030.9 ± 163.8	112.1 ± 19.5
Proposed	$\mathbf{2438.2 \pm 219.1}$	3465.4 ± 109.3	$\mathbf{112.3 \pm 9.2}$

I Additional Experimental Results

J Ablation Studies on the number of Particles and Diffusion Steps

We conduct additional ablation studies to analyze the impact of two key hyperparameters in our framework: (i) the number of particles used for covariance estimation, and (ii) the number of diffusion steps used during sampling.

Number of parallel particles. We vary the number of particles used to estimate the gradient covariance (e.g., 8, 16, 32) and report the results in Figure 5. Overall, increasing the number of particles improves performance across most environments, particularly in the Ant and Walker2d. This suggests that a more accurate estimation of the gradient covariance leads to a better approximation of the underlying geometry.

Number of diffusion steps. We further evaluate the effect of different numbers of diffusion steps (e.g., 5, 10, 20), as shown in Figure 6. The optimal number of steps varies across environments; while more steps can improve performance, excessive steps may introduce stochasticity and lead to fluctuations. A small number of steps already achieves competitive results which proves that geometry-aware diffusion policies may contribute to the sampling efficiency.

These results indicate that our method is robust to both hyperparameters; the gains from preconditioning are largest in environments where the critic landscape is most anisotropic (Swimmer, Ant).

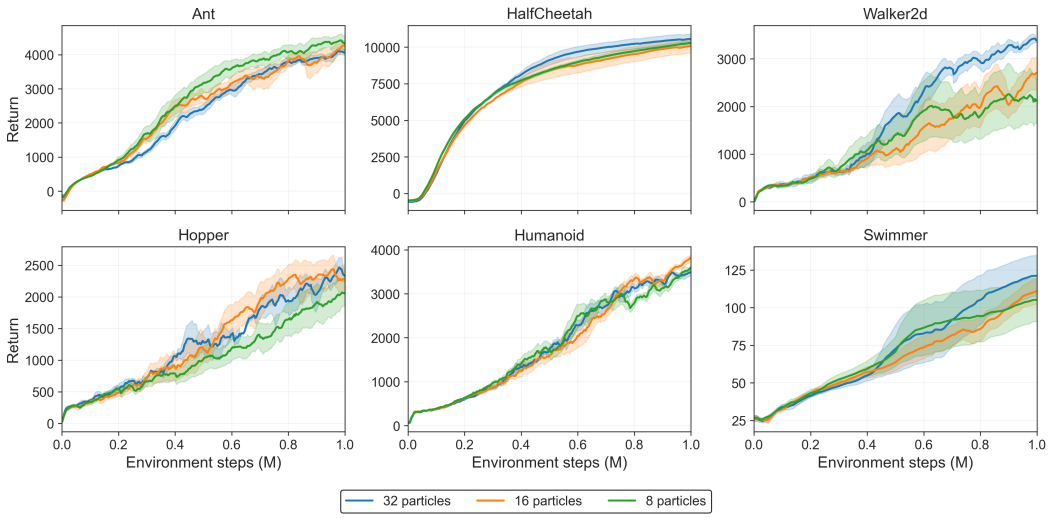


Figure 5: Ablation studies on the number of particles used for parallel sampling and covariance estimation.

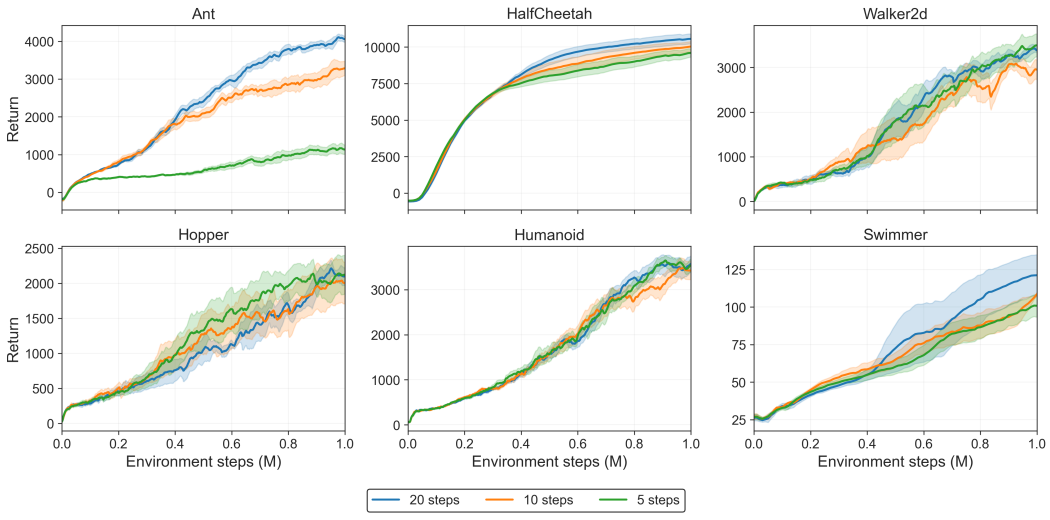


Figure 6: Ablation studies on the number of diffusion steps.