Learning a Diffusion Model Policy from Rewards via Q-Score Matching

Michael Psenka [*1] **Alejandro Escontrela** [*1] **Pieter Abbeel** [1] **Yi Ma** [1]

Abstract

Diffusion models have become a popular choice for representing actor policies in behavior cloning and offline reinforcement learning. This is due to their natural ability to optimize an expressive class of distributions over a continuous space. However, previous works fail to exploit the score-based structure of diffusion models, and instead utilize a simple behavior cloning term to train the actor, limiting their ability in the actor-critic setting. In this paper, we present a theoretical framework linking the structure of diffusion model policies to a learned Q-function, by linking the structure between the score of the policy to the action gradient of the Q-function. We focus on off-policy reinforcement learning and propose a new policy update method from this theory, which we denote *Q-score matching*. Notably, this algorithm only needs to differentiate through the denoising model rather than the entire diffusion model evaluation, and converged policies through Q-score matching are implicitly multi-modal and explorative in continuous domains. We conduct experiments in simulated environments to demonstrate the viability of our proposed method and compare to popular baselines. Source code is available from the project website: `https://michaelpsenka.io/qsm`.

1. Introduction

Reinforcement Learning (RL) has firmly established its importance across a range of complex domains, from discrete game environments like Go, Chess, and Poker (Silver et al., 2016; 2017; Brown & Sandholm, 2019) to continuous environments like goal-oriented robotics (Kober et al., 2013; Sünderhauf et al., 2018; Ibarz et al., 2021; Wu et al., 2022). Robotics RL applications typically need to work in a continuous vector space for both states and actions. This not only makes traditional RL algorithms designed for discrete state and action spaces infeasible, but makes parameterizing the policy (distribution of actions) a difficult challenge, where one must typically choose between ease of sampling (e.g. Gaussians (Agostini & Celaya, 2010)) and expressiveness.

Recently, diffusion models (Hyvärinen & Dayan, 2005; Ho et al., 2020) have emerged as a promising avenue for parameterizing distributions. These models, rooted in the idea of iterative increments of noising and denoising from a distribution, have shown great potential in generative tasks (Rombach et al., 2022; Watson et al., 2022). In the context of RL, diffusion models offer both expressiveness and easy sampling, since normalization constants do not need to be computed for sampling. However, their adoption in RL is relatively nascent, and the nuances of their implementation and performance are still subjects of investigation.

One unexplored approach is through the alignment of the learned policy's score, denoted by $\nabla_a \log(\pi(a|s))$, with the score of an optimal policy, denoted by $\nabla_a \log(\pi^*(a|s))$. However, traditional score matching is ill-posed in this setting, because we not only lack samples from $\nabla_a \log(\pi^*(a|s))$, but also from $\pi^*(a|s)$ itself. Our primary result emphasizes that in the context of RL, one can match the score of π to that of π^* by iteratively matching the score of π to the gradient of the state-action value function with respect to action, $\nabla_a Q^\pi(s,a)$. This offers a new, geometric perspective on policy optimization, where the focus for policy optimization becomes iteratively pushing the vector field $\nabla_a \log(\pi(a|s))$ towards the vector field $\nabla_a Q^\pi(s,a)$. We call this approach *Q-score matching* (QSM).

We then use this novel method on off-policy reinforcement learning scenarios, an important but yet unexplored area for diffusion model policies. Without a fixed distribution to sample from (akin to what is given for behavior cloning or offline RL applications), it is unclear how exactly to train a diffusion model represented policy. We postulate and empirically demonstrate that QSM is a viable algorithm for learning diffusion model policies.

The paper is structured as follows: we begin by establishing a continuous-time formulation of RL via stochastic differential equations to allow score-based analysis while keeping

[*]Equal contribution [1]Department of Electrical Engineering and Computer Science, University of California, Berkeley. Correspondence to: Michael Psenka <psenka@eecs.berkeley.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

the stochastic flexibility of the RL setting. We then introduce the standard policy gradient for diffusion models in this setting, and afterwards give a theoretical introduction to QSM. Finally, we lay out our empirical framework, results, and the broader implications of our method for learning diffusion model policies in reinforcement learning problems.

1.1. Related work

We now introduce the relevant related work for this paper. The most direct relation is the line of work related to diffusion models in the RL setting, which is relatively new but actively explored. We discuss some particular works related to various parts of this paper's setting, including the action gradient of the Q-function and the score of the policy distribution.

1.1.1. Diffusion models in reinforcement learning

Diffusion models, characterized by their incremental noise-driven evolution of data, have found various applications in RL, ranging from imitating expert behavior to optimizing policies using complex action-value functions. The following are some notable examples:

Behavior cloning: "Behavior cloning" is a type of imitation learning where an agent tries to mimic the behavior of an expert without explicit reward feedback. Much of the earlier work in diffusion models for policy learning has been for behavior cloning (Janner et al., 2022; Reuss et al., 2023), as the specifics of the behavior cloning setting (matching a distribution to a given dataset) fit more closely to the original design principle of diffusion models, namely through score matching (Hyvärinen & Dayan, 2005). In a similar vein, work has been done on learning a stochastic state dynamics model using a diffusion model (Li et al., 2022a).

Offline Q-learning: Offline RL techniques leverage existing datasets to learn optimal policies without further interaction with the environment. Similar to the behavior cloning setting, optimizing the policy using a large fixed dataset fits more closely with the common usage of diffusion models for learning distributions, and in combination with the above policy gradient formula for diffusion models has enabled many recent works in this area (Wang et al., 2022; Suh et al., 2023; Kang et al., 2023; Hansen-Estruch et al., 2023; Lu et al., 2023).

Policy gradient: Policy gradient methods seek to directly optimize the policy by computing gradients of the expected reward with respect to the policy parameters (Sutton et al., 1999). Previous work has derived a formula for the policy gradient with respect to a diffusion model's parameters (Black et al., 2023), but such formulas are much more general and do not fully exploit the structure of a diffusion model. For example, the new expectation for the policy gradient becomes dependent on internal action samples, making the estimates less sample efficient (see Section 3).

There are additional works that use the action gradient of the Q-function for learning (Silver et al., 2014; Berseth et al., 2018; D'Oro & Jaśkowski, 2020; Li et al., 2022b; Sarafian et al., 2021), where the standard policy gradient is expanded to include the action gradient of the Q-function through the chain rule, but such methods require an explicit representation for the full policy distribution $\pi(a|s)$, which is not readily available for diffusion models.

Diffusion-QL: Although Wang et al. (Wang et al., 2022) perform experiments in the offline-RL setting with a behavior cloning term, they propose a method for pure Q-learning: training on Q itself as the objective, and backpropagating through the diffusion model evaluation. However, such training still does not utilize the unique structure of diffusion models and presents computational challenges (e.g. exploding/vanishing gradients from differentiating through model applied on itself). In this paper, we explore a method that does not require backpropagating through the entire diffusion evaluation, but rather just the internal denoising model, much like standard diffusion model training.

1.1.2. Stochastic optimal control

At a conceptual level, our work is rooted in the principles of stochastic optimal control (Fleming & Rishel, 2012; Kirk, 2004; Bellman, 1954), which deals with optimizing systems subjected to random disturbances over time. Especially relevant to our context is the continuous-time formulation, where the control strategies are adjusted dynamically in response to evolving system states. However, much of stochastic optimal control literature typically assumes access to some model of the state dynamics. Instead of assuming a state dynamics model, we assume a model for the expected discounted rewards over time (the Q-function), leading to new motivations for the theoretical development of our method. Nonetheless, exploring the link between this paper and stochastic optimal control is an interesting direction for future work, in particular given the surface similarities between Theorems 4.1 and 4.3 and the Hamilton-Jacobi-Bellman equation (Bellman, 1954).

2. Problem formulation

We now define and introduce core mathematical notation and objects used. Of notable difference from a standard reinforcement learning formulation is the use of a continuous time formulation, which is helpful to simplify the theoretical statements of this paper.

2.1. Notation

We first introduce non-standard notation used throughout the main body and proofs in the appendix.

1. Different notations for the time derivative of a path $x(t)$ are used, depending on the setting. In the non-stochastic/deterministic setting, we use dot notation $\dot{x}(t) \coloneqq \frac{d}{dt}x(t)$. In the stochastic setting, we use the standard SDE notation, using $dx(t)$ instead of $\frac{d}{dt}x(t)$.
2. In the main body and proofs, we often refer to the "score" of the action distribution, which is always denoted Ψ and always used as the vector field defining the flow of actions over time. Not all settings have Ψ line up with the classical score of a distribution, but we use the terminology "score" throughout to highlight the analogy to a distribution's true score $\nabla_a \log \pi(a|s)$, as seen most clearly through the Langevin dynamics with respect to a distribution's score (Langevin, 1908; Papanicolaou, 1977; Welling & Teh, 2011).

2.2. Definitions

We denote the *state space* as a Euclidean space $\mathcal{S} = \mathbb{R}^s$, and the *action space* as another Euclidean space $\mathcal{A} = \mathbb{R}^a$. For the theoretical portion of this paper, we consider the following stochastic, continuous-time setting for state and action dynamics:

$$\begin{aligned} ds &= F(s,a)dt + \Sigma_s(s,a)dB_t^s, \\ da &= \Psi(s,a)dt + \Sigma_a(s,a)dB_t^a, \\ s(0) &= s_0, \\ a(0) &= a_0. \end{aligned} \tag{1}$$

$\Psi : \mathcal{S} \times \mathcal{A} \to \mathcal{A}$ corresponds to the "score" of our policy and is the main parameter for policy optimization in this setting. $F : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$ corresponds to the continuous state dynamics, and $\Sigma_s(s,a), \Sigma_a(s,a)$ are functions from $\mathcal{S} \times \mathcal{A}$ to positive semidefinite matrices in $\mathbb{R}^{s\times s}$ and $\mathbb{R}^{a\times a}$ respectively, corresponding to the covariance structure for the uncertainty in the dynamics of $s(t)$ and $a(t)$. The covariances are with respect to the separate Brownian motions B^s, B^a, each embedded in $\mathcal{S}$ and $\mathcal{A}$ respectively.

While many works formulate RL in a continuous state/action/time setting (Bradtke & Duff, 1994; Doya, 2000; Jia & Zhou, 2022; 2023; Zhao et al., 2024), it is less common to formulate the action as a joint dynamical system. This proves important when linking this theoretical structure to a diffusion model, as highlighted in Section 2.3.

Our main objective in this paper is to maximize path integral loss functions of the following form:

$$Q^\Psi(s_0,a_0) = \mathbb{E}\int_0^\infty \gamma^t r(s(t,s_0,a_0))dt, \tag{2}$$

where $r : \mathcal{S} \to [0,1]$ is the *reward function*, the expectation is taken over the stochastic dynamics given in (1), $s(t,s_0,a_0)$ is a sample of the path $s(\cdot)$ at time t from initial conditions (s_0,a_0), and $\gamma \in (0,1)$ is a fixed constant corresponding to a "discount factor". Discretizing by time gives a more familiar formula for the Q-function:

$$Q^\Psi(s_0,a_0) = \mathbb{E}\sum_{t=0}^\infty \gamma^t r(s(t,s_0,a_0)), \tag{3}$$

and furthermore if we cut off at some horizon where $\sum_{T+1}^\infty \gamma^i \approx 0$:

$$Q^\Psi(s_0,a_0) = \mathbb{E}\sum_{t=0}^T \gamma^t r(s(t,s_0,a_0)). \tag{4}$$

We write superscript Ψ because we want to consider Q as a function not of the initial conditions, but of the score Ψ, and try to find a score Ψ^* that maximizes $Q^\Psi(s_0,a_0)$ for a fixed initial condition (or an expectation over a distribution of initial conditions).

There is of course motivation for optimizing the noise covariance structure for the actions Σ_a, but we save this optimization for a future work and here purely focus on the score Ψ.

Our objective in this paper is then to maximize the following function with respect to the vector field $\Psi : \mathcal{S} \times \mathcal{A} \to \mathcal{A}$ for some distribution over initial state/action pairs $\mathbb{P} \times \pi$:

$$J(\Psi) = \mathop{\mathbb{E}}_{\mathbb{P}\times\pi} Q^\Psi(s,a). \tag{5}$$

2.3. Time discretization

One may observe that the action model in (1) is not standard, since actions $a(t)$ are modeled as a smooth flow over time, rather than an explicit function of the state $s(t)$. The motivation for this model comes when we discretize in time via the Euler-Maruyama method, and further when we discretize at different time scales with respect to the state and action dynamics in the following way:

$$\begin{aligned} s_{t+1} &= s_t + F(s_t,a_t) + z, z \sim \mathcal{N}(\mathbf{0}, \Sigma_s(s_t,a_t)), \\ a_t^i &= a_t^{i-1} + \frac{1}{K}\Psi(s_t,a_t^{i-1}) + z_t^i, \\ z_t^i &\sim \mathcal{N}(\mathbf{0}, \frac{1}{K}\Sigma_a(s_t,a_t^{i-1})), \\ a_{t+1} &= a_t^K, \end{aligned} \tag{6}$$

where $a_t^i \coloneqq a_{t+i/K}$. By discretizing the action dynamics at K times the fidelity of the state dynamics discretization, we recover a time-invariant diffusion model of depth K, with denoising mean represented by Ψ and variance represented by Σ_a. As a result, we expect the theory on the model in (1)

to also approximately hold for systems of the form given in (6) where actions are represented by a diffusion model, up to the error incurred by time discretization. This time scaling also allows actions to depend on the state s_t more than the previous action a_{t-1}; as the number of sampling steps K increases, the initial condition a_{t-1} becomes less influential.

3. Policy gradient for diffusion policies

Recall that the policy gradient, the gradient of the global objective $J(\theta) = \mathbb{E}_{s,a} Q^\pi(s,a)$ is given by the following (Sutton et al., 1999):

$$\nabla_\theta J(\theta) = \mathop{\mathbb{E}}_{(s_0,a_0,s_1,a_1,\ldots)} \sum_{t=1}^{\infty} Q(s_t,a_t) \nabla_\theta \log \pi_\theta(a_t|s_t), \tag{7}$$

where the distribution of the expectation is with respect to the policy π and the environment dynamics. For a time-discretization of a diffusion model policy, we do not have access to global probability $\pi(a|s)$, but rather just the incremental steps $\pi(a^\tau|a^{\tau-1},s)$ (superscript here indicates internal diffusion model steps), and would need to integrate over all possible paths to get the following:

$$\pi(a^K|s) = \tag{8}$$
$$\int_{a^{K-1}} \cdots \int_{a^1} \pi(a^1|s) \prod_{\tau=2}^{K} \pi(a^\tau|a^{\tau-1},s) da^1 \cdots da^{K-1},$$

which is quickly infeasible to compute with respect to K. However, after substituting (8) into (7), we can simplify and obtain the following form:

$$\nabla_\theta J(\theta) = \mathbb{E} \sum_{t=1}^{\infty} Q(s_t,a_t^K) \left(\sum_{\tau=1}^{K} \nabla_\theta \log \pi_\theta(a_t^\tau|a_t^{\tau-1},s_t) \right), \tag{9}$$

where the expectation is taken over the all states $\{s_t\}_{t=1}^\infty$ and all internal actions $\{\{a_t^\tau\}_{\tau=1}^K\}_{t=1}^\infty$. Proof of this can be found in Appendix B.1. Importantly, the expectation now explicitly depends on the internal action states $\{a_t^\tau\}_{\tau=1}^K$ rather than just the executed action a_t, and uses no explicit structure of diffusion models; this formula likewise holds for any model with internal actions $\{a_t^\tau\}_{\tau=1}^K$. It is thus reasonable to expect that applying this formula directly will lead to sample inefficiency; we present comparative results in Section 5. We are then motivated to pursue alternative methods for updating the diffusion model policy, that use some of the score-based structure of the diffusion model.

4. Policy optimization via matching the score to the Q-function

We now introduce the main theoretical contribution of this work: an alternative method for updating the policy given an estimate of the Q-function that avoids the aforementioned problems from policy gradients.

The theory of this section is relevant for continuous-time dynamical systems. Thus, we consider the continuous-time Q-function given in eq. (2) and the corresponding total energy function $J(\Psi)$:

$$J(\Psi) = \mathop{\mathbb{E}}_{(s,a)} Q^\Psi(s,a), \tag{10}$$

where the expectation for Q^Ψ is taken over sampled paths of $(s(t),a(t))$ with respect to the stochastic dynamics from (1) starting from (s_0,a_0). In computation, we can also consider a finite-dimensional parameterization Ψ_θ for parameters $\theta \in \mathbb{R}^d$ and the parameterized loss:

$$J(\theta) = \mathop{\mathbb{E}}_{(s,a)} Q^{\Psi_\theta}(s,a), \tag{11}$$

where the parameters θ parameterize the vector field Ψ for the actions given in (1).

Our goal is to match the score Ψ with that of an optimal distribution with respect to J, denoted Ψ^*. However, unlike the standard score matching literature, we not only do not have access to any samples of Ψ^*, but not even from the action distribution π^* it generates. Thus, our matching approach will require access to a surrogate Ψ^* approximator.

One hypothesis from dimensional analysis is to compare Ψ to $\nabla_a Q^\Psi$, which is also a vector field from $\mathcal{S} \times \mathcal{A}$ to $\mathcal{A}$. Intuitively, $\nabla_a Q^\Psi$ provides which ways actions should be modified locally to maximize expected reward. As such, we define $\Psi^* \approx \nabla_a Q$ and define our actor update as an iterative optimization of the score Ψ against the $\nabla_a Q$ target.

One can optimize Ψ by iteratively matching it to the action gradient of its Q-function, $\nabla_a Q^\Psi(s,a)$. Our theoretical statements for this are captured in Theorems 4.1 and 4.3, and their proofs in appendix B.2 and B.3 respectively. Gradients with respect to this optimization would only depend on the denoising model Ψ itself, rather than the full diffusion model evaluation.

4.1. Non-stochastic setting

We begin by proving the above result in the non-stochastic setting. That is, the state and action both evolve as paired ordinary differential equations. To further simplify, we will begin by trying to optimize just the Q-function at a fixed point $Q(s_0,a_0)$, or without loss of generality of the theory, the Q-function at the origin $Q(\mathbf{0},\mathbf{0})$. This is still quite a rich setting, since even for a fixed point $Q(s,a)$ is still a path integral equation. The following theorem formalizes the setting, and gives a concrete statement relating the optimal policy's score to the action gradient of the Q-function.

Theorem 4.1 (Optimality condition, deterministic setting)**.**

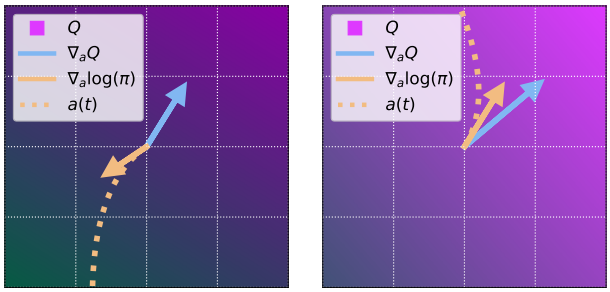

Figure 1. A visual description of Theorem 4.1 and Theorem 4.3, and the implied update rule for a policy π parameterized by a diffusion model. The left image depicts a randomly initialized score Ψ^0, and the right the result after one step of QSM Ψ^1. If there is any discrepancy between the score $\nabla_a \log(\pi(a|s))$ (orange vector, denoted Ψ in the paper and optimized directly) and the action gradient $\nabla_a Q(s,a)$ (blue vector), we can forcefully align the score to the Q action gradient to strictly increase the Q value at (s,a).

Consider the following joint deterministic dynamics governing the state $s(t) \in \mathbb{R}^s$ and action $a(t) \in \mathbb{R}^a$:

$$\begin{aligned} &\dot{s}(t) = F(s(t), a(t)), \quad \dot{a}(t) = \Psi(s(t), a(t)), \\ &s(0) = \mathbf{0}, \quad a(0) = \mathbf{0}. \end{aligned} \tag{12}$$

where $s(t) \in \mathbb{R}^s, a(t) \in \mathbb{R}^a$, $\|\Psi(s,a)\|_2 \le C$ for all (s,a), and Ψ is Lipschitz with respect to $\|\cdot\|_2$. Denote $s(t, s_0, a_0)$ the resulting state $s(t)$ from initial conditions s_0, a_0. Let $r : \mathbb{R}^s \to [0,1]$ be a smooth function. Finally, let $\Psi^* : \mathbb{R}^s \times \mathbb{R}^a \to \mathbb{R}^a$ be a vector field that maximizes the following path norm for fixed $\gamma \in (0,1)$ over smooth flows Ψ with norm bound C and bounded Lipschitz constant:

$$J(\Psi) = Q^\Psi(\mathbf{0}, \mathbf{0}), \tag{13}$$

$$Q^\Psi(s_0, a_0) = \int_0^\infty \gamma^t r(s(t, s_0, a_0))dt, \tag{14}$$

it follows that $\Psi^*(s,a) = \alpha_{s,a} \nabla_a Q^{\Psi^*}(s,a)$ for some $\alpha_{s,a} > 0$ for all (s,a) along the trajectory $(s(t), a(t))$ where $\nabla_a Q^{\Psi^*}(s,a) \neq \mathbf{0}$.

Proof can be found in Section B.2. *At their core, Theorems 4.1 and 4.3 state the following*: if there is anywhere that $\Psi(s,a)$ and $\nabla_a Q^\Psi(s,a)$ are not aligned, then a new vector field Ψ' resulting from aligning $\Psi(s,a)$ with $\nabla_a Q^\Psi(s,a)$ will strictly increase $Q^{\Psi'}(\mathbf{0}, \mathbf{0})$ towards $Q^*(\mathbf{0}, \mathbf{0})$. As one can imagine, once we extend the above to losses of integrals over Q, the collinear condition extends to the entire integrated space.

Corollary 4.2. *Consider the setting of Theorem 4.1, but with the following modified loss:*

$$J(\Psi) = \mathop{\mathbb{E}}_{\mathbb{P} \times \pi} Q^\Psi(s,a), \tag{15}$$

where $\mathbb{P} \times \pi$ is a probability measure over $\mathbb{R}^s \times \mathbb{R}^a$ absolutely continuous with respect to the Lebesgue measure. For any maximizer Ψ^*, it follows that $\Psi^*(s,a) = \alpha_{s,a} \nabla_a Q^{\Psi^*}(s,a)$ for all $(s,a) \in \mathbb{R}^s \times \mathbb{R}^a$ where $\nabla_a Q^{\Psi^*}(s,a) \neq \mathbf{0}$.

4.2. Stochastic setting

We now extend to stochastic differential equations over the actions, as modeled by (1). Fortunately, the theory for the deterministic case in Theorem 4.1 extends rather identically to the fully stochastic case.

Theorem 4.3 (Optimality condition, stochastic setting)**.** *Consider the following joint stochastic dynamics governing the state $s(t) \in \mathbb{R}^s$ and action $a(t) \in \mathbb{R}^a$:*

$$\begin{aligned} ds(t) &= F(s(t), a(t))dt + \Sigma_s(s(t), a(t))dB_t^s, \\ da(t) &= \Psi(s(t), a(t))dt + \Sigma_a(s(t), a(t))dB_t^a, \\ s(0) &= s_0, \\ a(0) &= a_0, \end{aligned} \tag{16}$$

where Ψ, F are globally Lipschitz functions defined from $\mathbb{R}^s \times \mathbb{R}^a$ to $\mathbb{R}^a$, $\Sigma_s(s,a)$ is a globally Lipschitz function from $\mathbb{R}^s \times \mathbb{R}^a$ to positive semidefinite matrices in $\mathbb{R}^{s \times s}$ (and similarly for $\Sigma_a(s,a)$), and B_t^s, B_t^a are separate standard Brownian motions in $\mathbb{R}^s$ and $\mathbb{R}^a$. Further consider the following loss function with respect to the vector field Ψ:

$$J(\Psi) = \mathop{\mathbb{E}}_{\mathbb{P} \times \pi} Q^\Psi(s,a), \tag{17}$$

$$Q^\Psi(s_0, a_0) = \mathbb{E} \int_0^\infty \gamma^t r(s(t, s_0, a_0))dt, \tag{18}$$

where $\mathbb{P} \times \pi$ is a probability measure over $\mathbb{R}^s \times \mathbb{R}^a$, and the expectation for Q^Ψ is taken over sampled paths of $(s(t), a(t))$ starting from (s_0, a_0). For any optimal vector field Ψ^* with respect to J, it follows that $\Psi^*(s,a) = \alpha_{s,a} \nabla_a Q^{\Psi^*}(s,a)$ for some $\alpha_{s,a} > 0$ for all $(s,a) \in \mathbb{R}^s \times \mathbb{R}^a$ where $\nabla_a Q^{\Psi^*}(s,a) \neq \mathbf{0}$ in the support of the distribution generated by $\mathbb{P}, \pi$, and the stochastic dynamics of $(s(t), a(t))$.

Proof can be found in Appendix B.3. While this theorem does not imply a specific algorithm, it does provide theoretical justification for a class of policy update methods: iteratively matching $\Psi(s,a)$ to $\nabla_a Q(s,a)$ will provide strict increases to the resulting Q-function globally.

4.3. Pedagogical reduction in gridworld

To provide intuition for Theorem 4.3, we illustrate a reduction of the theorem in gridworld, where the state space $\mathcal{S} = \{1, \dots, M\} \times \{1, \dots, N\}$ for some fixed $M, N \in \mathbb{N}$, and $\mathcal{A} = \{\text{LEFT}, \text{RIGHT}, \text{UP}, \text{DOWN}\}$. While diffusion models are not well-defined in discrete space, there is a sensible reduction to gridworld through the Langevin dynamics link.

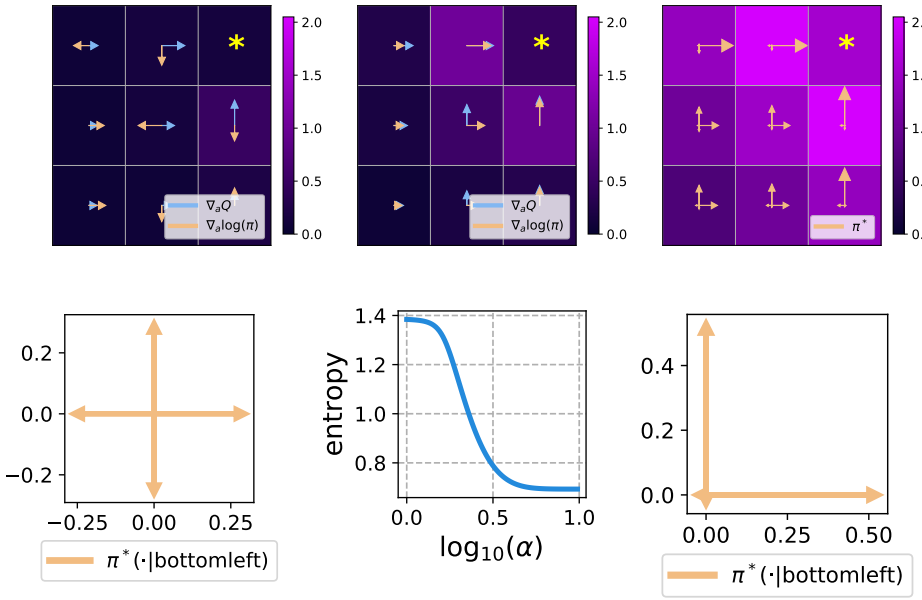

Figure 2. Pedagogical simulation of our algorithm's reduction to a simple single-goal gridworld setting. **The top row** is a visualization of two iterates of $\pi(a|s) \leftarrow e^{\alpha Q^\pi(s,a)} / \sum_{a'} e^{\alpha a Q^\pi(s,a')}$, for $\alpha = 2$. The color of each square is the expected reward starting from that square, and we use the local maximizing direction to define discrete gradients: $\nabla_a Q(s,a) := a^* Q(s,a^*)$, where $a^* := \text{argmax}_{a'} Q(s,a')$, and similarly for $\nabla_a \log(\pi(a|s))$. **The bottom row** shows the effect of the parameter α on the entropy of the converged distribution $\pi*(s|a)$. To the left is the learned policy with $\alpha = 1$, and to the right the learned policy with $\alpha = 10$.

Suppose (for Euclidean $\mathcal{S}$ and $\mathcal{A}$) we have completed the matching described in Theorem 4.3 and found a score Ψ^* such that $\Psi^*(s,a) = \alpha \nabla_a Q(s,a)$ for some fixed $\alpha > 0$. Consider the dynamics of (1) for a fixed state ($F(s,a) = 0, \Sigma_s(s,a) = 0$), and setting $\Sigma_a(s,a) = \sqrt{2} I_a$; given certain conditions (Bakry et al., 2014), we have the following result for the stationary distribution of actions $a(t)$ as $t \to \infty$ for any fixed $s \in \mathcal{S}$, denoted $\pi(a|s)$:

$$\pi(a|s) \sim e^{\alpha Q(s,a)}. \tag{19}$$

Thus, we can view QSM as matching the full action distribution $\pi(a|s)$ the Boltzmann distribution of the Q-function, $\frac{1}{Z} e^{\alpha Q(s,a)}$, where $Z = \int_{\mathcal{A}} e^{\alpha Q(s,a)}$. While directly setting $\pi(a|s) \sim e^{\alpha Q(s,a)}$ is infeasible in continuous state/action spaces, we can represent the probabilities $\pi(a|s)$ directly as a matrix of shape $|\mathcal{S}| \times |\mathcal{A}|$ in the finite case and use the following update rule:

$$\pi'(s|a) = \frac{e^{\alpha Q^\pi(s,a)}}{\sum_{\mathcal{A}} e^{\alpha Q^\pi(s,a)}}. \tag{20}$$

This update rule is the same as *soft policy iteration* (Haarnoja et al., 2018), but using the standard Q-function rather than the soft Q-function. Nonetheless, we still see in simulated gridworld environments that α acts as an inverse entropy regularization parameter: the lower α is, the higher the entropy of the converged distribution $\pi^*(a|s)$. We visualize this update rule in a simple gridworld environment, depicted in Figure 2.

Algorithm 1: Q-Score Matching (QSM)

Initialize critic networks Q_{θ_1}, Q_{θ_2}, and score network Ψ_ϕ with random parameters θ_1, θ_2, ϕ
Initialize target networks $\theta_1' \leftarrow \theta_1$, $\theta_2' \leftarrow \theta_2$
Initialize replay buffer $\mathcal{B}$
while *not converged* **do**
  Choose action by iteratively denoising $a^T \sim \mathcal{N}(0, I) \to a^0$ using Ψ_ϕ: $a_t \sim \pi_\phi(x_t)$;
  Step environment: $x_{t+1}, r_{t+1} \leftarrow \text{env}(a_t)$;
  Store $(x_t, a_t, r_{t+1}, x_{t+1})$ in $\mathcal{B}$;
  Sample minibatch of N transitions $(x_t, a_t, r_{t+1}, x_{t+1})$ from $\mathcal{B}$;
  Sample actions via iterative denoising using Ψ_ϕ: $\tilde{a}_{t+1} \sim \pi_\phi(x_{t+1})$;
  Compute critic target: $y_t = r_{t+1} + \gamma \min_{i=1,2} Q_{\theta_i'}(x_{t+1}, \tilde{a}_{t+1})$;
  Update critics: $\theta_i = \text{argmin}_{\theta_i} N^{-1} \sum (y_t - Q_{\theta_i}(s_t, a_t))^2$;
  Update score model: $\phi = \text{argmin}_\phi N^{-1} \sum (\Psi_\phi(x_t, a_t) - \nabla_a Q(x_t, a_t))^2$;
  Update target networks: $\theta_i' \leftarrow \tau \theta_i + (1-\tau)\theta_i'$;

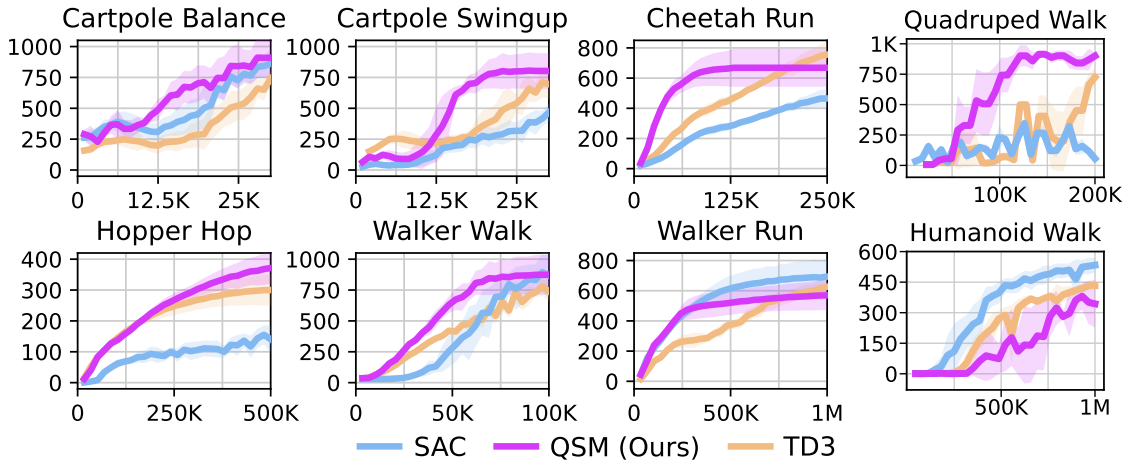

Figure 3. Experimental results across a suite of eight continuous control tasks. QSM matches and sometimes outperforms TD3 and SAC performance on the tasks evaluated, particularly in samples needed to reach high rewards. Even though QSM trains on expressive diffusion models, it matches the sample efficiency of explicit Gaussian and tanh-parameterized models.

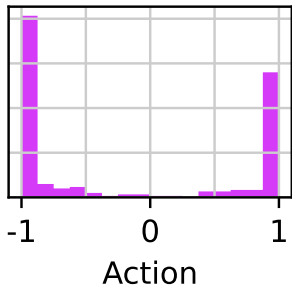

Figure 4. QSM can learn multi-modal policies. Samples from policy shown for the first state of a toy cartpole swingup task, where -1 and 1 represent the initial action for each of two optimal trajectories.

5. Experiments

In this section, we describe a practical implementation of QSM and evaluate our algorithm on various environments from the Deepmind control suite (Tunyasuvunakool et al., 2020). We seek to answer the following questions:

1. Can QSM learn meaningful policies provided limited interaction with the environment?
2. Does QSM learn complex, multi-modal policies?
3. How does QSM compare to popular baselines?

In particular, the first point allows us to verify that using the score-based structure of diffusion models allows us to train diffusion model policies in a sample-efficient manner. We implement QSM using N-step returns (Sutton & Barto, 2018) and the Q-function optimization routine outlined in DDPG (Lillicrap et al., 2019). For each update step, actions from the replay buffer are noised according to a variance-preserving Markov chain, similar to (Hansen-Estruch et al., 2023). We then evaluate $\nabla_a Q$ and provide this as the target to our score model Ψ. This update is computed for every environment step with batches of size 256 sampled from the replay buffer. Both the critic and score model are parameterized as 2-layer MLPs with 256 units per layer. Pseudocode for the algorithm is provided in algorithm 1. In this work, we add a small amount of Gaussian noise to the final sampled actions. We note that other, interesting exploration strategies exist, such as limiting the amount of denoising steps applied to the action. However, in this work we focus mainly on QSM, and leave additional study of exploration to future work.

5.1. Continuous Control Evaluation

In Figure 3, we evaluate QSM on six tasks from the Deepmind control suite of tasks. These tasks range from easy to medium difficulty. We also compare QSM to SAC (Haarnoja et al., 2018) and TD3 (Fujimoto et al., 2018), and find that QSM manages to achieve similar or better performance than the SAC and TD3 baselines. Conventional actor-critic methods parameterize the actor as a Gaussian distribution or a transformed Gaussian distribution, which can lead to sub-optimal exploration strategies, as certain action modes may get dropped or low Q-value actions may get sampled frequently. QSM instead learns policies which are not constrained to a fixed distribution class and are able to cover many different action modes. As such, QSM learn policies which are multi-modal and complex, thereby enabling policies to better approximate the true optimal policy. In Figure 4, we visualize actions sampled from a QSM policy for the initial state of a toy cartpole swingup task.

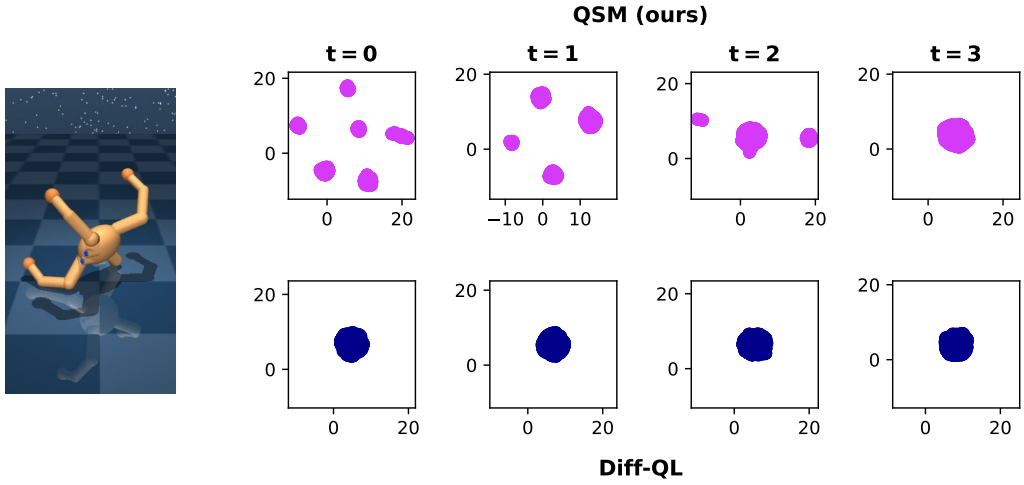

Figure 5. Demonstration of multimodal action distributions due to QSM. Displayed are sampled actions from a successfully QSM-trained model and a successfully Diffusion-QL-trained model. Each figure displays 1,000 sampled actions at the given time step from the displayed initial condition of quadruped_walk, projected down to $\mathbb{R}^2$ using UMAP (McInnes et al., 2018). Both compared models are exactly the same (including sampling procedure), except for the method the denoising submodel was trained. This demonstrates that the diversity in sampling from QSM comes not from the diffusion model architecture, but the training methodology itself.

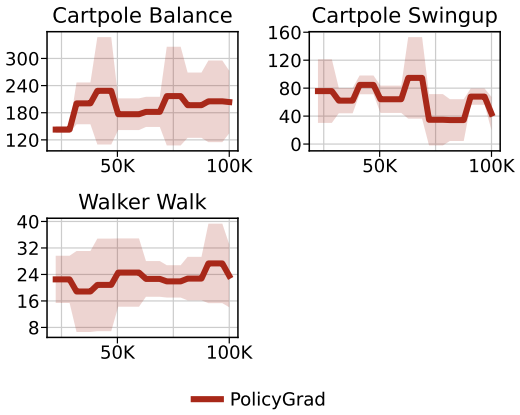

Figure 6. Performance when updating a diffusion model policy via policy gradients on select continuous control tasks, which is notably worse than QSM (ours), matching discussion in Section 3. Results on further environments can be found in Figure 8.

Note the high mass around the extreme actions, both of which represent the initial action for the two unique optimal trajectories.

6. Conclusion

Diffusion models offer a promising class of models to represent policy distributions, not only because of their expressibility and ease of sampling, but the ability to model the distribution of a policy through its score. To train such diffusion model policies in the reinforcement learning setting, we introduce the Q-score matching (QSM) algorithm, which iteratively matches the parameterized score of the policy to the action gradient of its Q-function. This gives a more geometric viewpoint on how to optimize such policies, namely through iteratively matching vector fields to each other. We additionally provide a practical implementation of this algorithm, and find favorable results when compared to popular RL algorithms.

There are still plenty of avenues for future work. This work focused purely on the score Ψ, or the "drift" term of the diffusion model, and assumed a fixed noise model Σ_a. However, optimization of Σ_a is an especially interesting direction for future work; delving deeper into this optimization might reveal intriguing connections to maximum entropy reinforcement learning, providing a richer understanding of the balance between exploration and exploitation. Additionally the theory for this paper builds a baseline for QSM and related algorithms, but there still remain many important theoretical questions to answer. In particular, convergence rate analysis could highlight further improvements to the QSM algorithm.

Acknowledgements

Michael Psenka acknowledges support from ONR grant N00014-22-1-2102. Alejandro Escontrela acknowledges support from an NSF Fellowship, NSF NRI #2024675. Yi Ma acknowledges support from ONR grant N00014-22-1-2102 and the joint Simons Foundation-NSF DMS grant #2031899. This work was partially supported by NSF

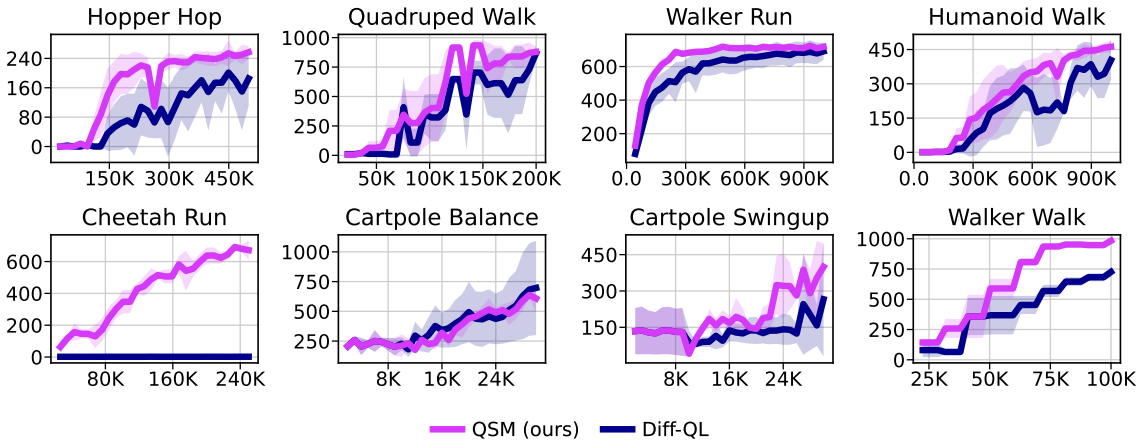

Figure 7. Comparison between QSM and Diffusion-QL (Wang et al., 2022) over the same hyperparameters – QSM performance here differs from Figure 3 due to the enforcement of consistent hyperparameters, in order to emphasize distinctions between QSM and Diffusion-QL. While performance is mostly comparable, note that QSM takes up significantly less computation time for the policy gradient, as it only differentiates through the denoising model rather than the full diffusion evaluation, and generally yields more consistent results, while Diffusion-QL can sometimes get "stuck" at suboptimal solutions. This idea is further supported through Figure 5, which demonstrates QSM's distinct promotion of exploration.

1704458, the Northrop Grumman Mission Systems Research in Applications for Learning Machines (REALM) initiative, NIH NIA 1R01AG067396, and ARO MURI W911NF-17-1-0304.

Impact statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Agostini, A. and Celaya, E. Reinforcement learning with a gaussian mixture model. In *The 2010 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE, 2010.

Bakry, D., Gentil, I., Ledoux, M., et al. *Analysis and geometry of Markov diffusion operators*, volume 103. Springer, 2014.

Bellman, R. Dynamic programming and a new formalism in the calculus of variations. *Proceedings of the national academy of sciences*, 40(4):231–235, 1954.

Berseth, G., Kyriazis, A., Zinin, I., Choi, W., and van de Panne, M. Model-based action exploration for learning dynamic motion skills. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1540–1546. IEEE, 2018.

Black, K., Janner, M., Du, Y., Kostrikov, I., and Levine, S. Training diffusion models with reinforcement learning. *arXiv preprint arXiv:2305.13301*, 2023.

Bradtke, S. and Duff, M. Reinforcement learning methods for continuous-time markov decision problems. *Advances in neural information processing systems*, 7, 1994.

Brown, N. and Sandholm, T. Superhuman ai for multiplayer poker. *Science*, 365(6456):885–890, 2019.

D'Oro, P. and Jaśkowski, W. How to learn a useful critic? model-based action-gradient-estimator policy optimization. *Advances in Neural Information Processing Systems*, 33:313–324, 2020.

Doya, K. Reinforcement learning in continuous time and space. *Neural computation*, 12(1):219–245, 2000.

Fleming, W. H. and Rishel, R. W. *Deterministic and stochastic optimal control*, volume 1. Springer Science & Business Media, 2012.

Fujimoto, S., van Hoof, H., and Meger, D. Addressing function approximation error in actor-critic methods, 2018.

Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor, 2018.

Hansen-Estruch, P., Kostrikov, I., Janner, M., Kuba, J. G., and Levine, S. Idql: Implicit q-learning as an actor-critic method with diffusion policies. *arXiv preprint arXiv:2304.10573*, 2023.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Hyvärinen, A. and Dayan, P. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(4), 2005.

Ibarz, J., Tan, J., Finn, C., Kalakrishnan, M., Pastor, P., and Levine, S. How to train your robot with deep reinforcement learning: lessons we have learned. *The International Journal of Robotics Research*, 40(4-5):698–721, 2021.

Janner, M., Du, Y., Tenenbaum, J., and Levine, S. Planning with diffusion for flexible behavior synthesis. In *International Conference on Machine Learning*, pp. 9902–9915. PMLR, 2022.

Jia, Y. and Zhou, X. Y. Policy gradient and actor-critic learning in continuous time and space: Theory and algorithms. *Journal of Machine Learning Research*, 23(275):1–50, 2022.

Jia, Y. and Zhou, X. Y. q-learning in continuous time. *Journal of Machine Learning Research*, 24(161):1–61, 2023.

Kang, B., Ma, X., Du, C., Pang, T., and Yan, S. Efficient diffusion policies for offline reinforcement learning. *arXiv preprint arXiv:2305.20081*, 2023.

Kirk, D. E. *Optimal control theory: an introduction*. Courier Corporation, 2004.

Kober, J., Bagnell, J. A., and Peters, J. Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*, 32(11):1238–1274, 2013.

Langevin, P. Sur la théorie du mouvement brownien. *C. R. Acad. Sci.*, 146:530–533, 1908.

Li, G., Li, J., Kabra, A., Srebro, N., Wang, Z., and Yang, Z. Exponential family model-based reinforcement learning via score matching. *Advances in Neural Information Processing Systems*, 35:28474–28487, 2022a.

Li, X., Qiao, Z., Gong, A., Lyu, J., Yu, C., Yan, J., and Li, X. Prag: Periodic regularized action gradient for efficient continuous control. In *Pacific Rim International Conference on Artificial Intelligence*, pp. 106–119. Springer, 2022b.

Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., and Wierstra, D. Continuous control with deep reinforcement learning, 2019.

Lu, C., Chen, H., Chen, J., Su, H., Li, C., and Zhu, J. Contrastive energy prediction for exact energy-guided diffusion sampling in offline reinforcement learning. *arXiv preprint arXiv:2304.12824*, 2023.

McInnes, L., Healy, J., and Melville, J. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

Papanicolaou, G. Martingale approach to some limit theorems. In *Papers from the Duke Turbulence Conference, Duke Univ., Durham, NC, 1977*, 1977.

Reuss, M., Li, M., Jia, X., and Lioutikov, R. Goal-conditioned imitation learning using score-based diffusion policies. *arXiv preprint arXiv:2304.02532*, 2023.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Sarafian, E., Keynan, S., and Kraus, S. Recomposing the reinforcement learning building blocks with hypernetworks. In *International Conference on Machine Learning*, pp. 9301–9312. PMLR, 2021.

Silver, D., Lever, G., Heess, N., Degris, T., Wierstra, D., and Riedmiller, M. Deterministic policy gradient algorithms. In *International conference on machine learning*, pp. 387–395. Pmlr, 2014.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016.

Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T., et al. Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *arXiv preprint arXiv:1712.01815*, 2017.

Suh, H., Chou, G., Dai, H., Yang, L., Gupta, A., and Tedrake, R. Fighting uncertainty with gradients: Offline reinforcement learning via diffusion score matching. *arXiv preprint arXiv:2306.14079*, 2023.

Sünderhauf, N., Brock, O., Scheirer, W., Hadsell, R., Fox, D., Leitner, J., Upcroft, B., Abbeel, P., Burgard, W., Milford, M., et al. The limits and potentials of deep learning for robotics. *The International journal of robotics research*, 37(4-5):405–420, 2018.

Sutton, R. S. and Barto, A. G. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018. URL http://incompleteideas.net/book/the-book-2nd.html.

Sutton, R. S., McAllester, D., Singh, S., and Mansour, Y. Policy gradient methods for reinforcement learning with

function approximation. *Advances in neural information processing systems*, 12, 1999.

Tunyasuvunakool, S., Muldal, A., Doron, Y., Liu, S., Bohez, S., Merel, J., Erez, T., Lillicrap, T., Heess, N., and Tassa, Y. dm control: Software and tasks for continuous control. *Software Impacts*, 6:100022, 2020. ISSN 2665-9638. doi: https://doi.org/10.1016/j.simpa.2020.100022. URL https://www.sciencedirect.com/science/article/pii/S2665963820300099.

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

Watson, D., Chan, W., Martin-Brualla, R., Ho, J., Tagliasacchi, A., and Norouzi, M. Novel view synthesis with diffusion models. *arXiv preprint arXiv:2210.04628*, 2022.

Welling, M. and Teh, Y. W. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pp. 681–688, 2011.

Wu, P., Escontrela, A., Hafner, D., Goldberg, K., and Abbeel, P. Daydreamer: World models for physical robot learning. *Proceedings of Machine Learning Research*, 2022.

Zhao, H., Tang, W., and Yao, D. Policy optimization for continuous reinforcement learning. *Advances in Neural Information Processing Systems*, 36, 2024.

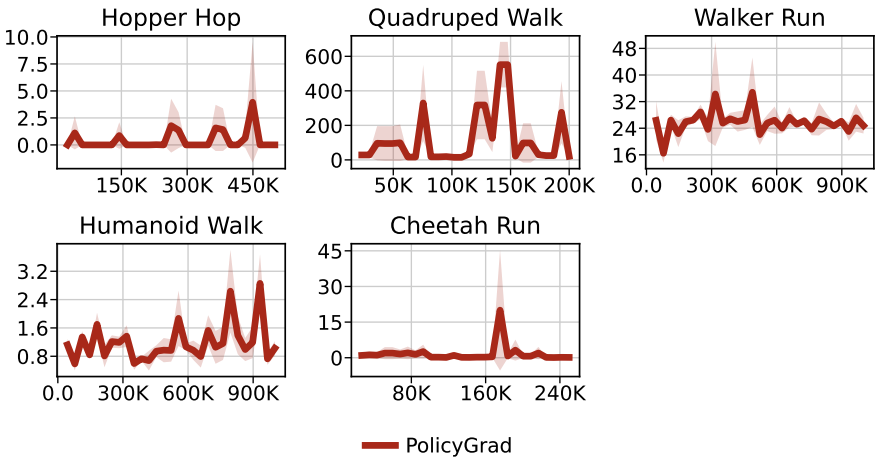

Figure 8. Results on a policy gradient update for the diffusion model policy on the remaining tested environments. As further confirmed, policy gradients fails to learn optimal policies for diffusion models in the off-policy setting. Note that some initializations of `quadruped_walk` have an incredibly wide space of policies that yield high rewards; e.g. when dropped completely right-side up.

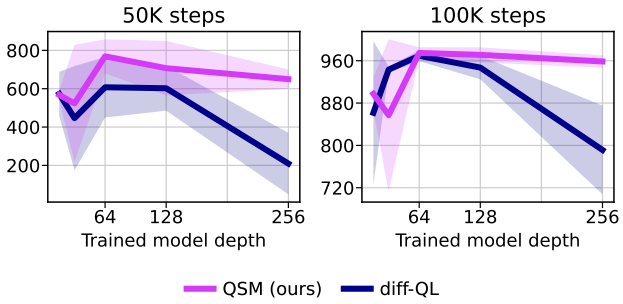

Figure 9. Comparison of QSM and Diffusion-QL (Wang et al., 2022) when increasing number of diffusion sampling steps, for the `walker_walk` environment. Plotted are the validation returns at 50K and 100K training steps respectively. Since QSM uses diffusion model structure and trains the noise model directly, it scales with diffusion depth, while Diffusion-QL becomes unstable.

A. Additional experiments

We provide here additional experiments that give further context for our method and algorithm. Some results, like Figure 9, demonstrate a future potential for QSM as we apply this method to more complex tasks that require deeper diffusion policies, and some results like Figure 8 give further context for the results presented in the main body.

B. Proofs

The following proofs build upon each other; for readability, please read all proofs in sequence. Note the following formalities:

1. Important notation and definitions used for this paper (and likewise for the following proofs) are provided in Section 2.

2. The below proofs consider optimizations over smooth collections of vector fields $\Psi : \mathbb{R}^s \times \mathbb{R}^a \to \mathbb{R}^a$, that are bounded in highest norm ($\mathcal{C}^0$ norm of $(s, a) \to \|\Psi(s, a)\|_2$) and are Lipschitz with some Lipschitz constant. This choice was made to give some control over the vector field space (bounded but not closed), but remain flexible enough to allow our local perturbations of vector fields towards optima. It remains an interesting line for future work to tighten the optimization space for the vector fields (e.g. fixed Lipschitz constant as well), alongside a sharper ascent rule (akin to (37)). For this paper, we wanted to simplify the optimization setting in order to focus theoretical insight on the relation of Ψ to its Q-function action gradient $\nabla_a Q^\Psi$.

3. All norms $\|\cdot\|$, unless specified, are the L^2 norm.

A note for readability. By far the longest and densest segment is the first part of the proof for Theorem 4.1, namely the section "aligning at the origin". This is where the majority of our analysis of the Q-function is built, and the rest of the theory will pretty straightforwardly follow from what is built up there. If you understand this portion, the rest of this paper's theory will follow.

B.1. Proof of policy gradient formula in (9)

We introduce here a formal theorem and proof pair to justify (9).

Theorem B.1. *Suppose we parameterize a policy $\pi_\theta(a|s)$ through an incremental sequence of steps, such that we only have access to marginal probabilities $\pi_\theta(a^\tau|a^{\tau-1}, s)$ that satisfy the following identity:*

$$\pi(a^K|s) = \int_{a^{K-1}} \cdots \int_{a^1} \pi(a^1) \prod_{\tau=2}^{K} \pi(a^\tau|a^{\tau-1}, s) da^1 \cdots da^{K-1}. \tag{21}$$

The following expectations for the policy gradient are then equal:

$$\begin{aligned} &\mathop{\mathbb{E}}_{(s_0,a_0,s_1,\ldots)} \sum_{t=1}^{\infty} Q(s_t, a_t) \nabla_\theta \log \pi_\theta(a_t|s_t) \\ &= \mathop{\mathbb{E}}_{(s_0,\{a_0^\tau\}_{\tau=1}^K,s_1,\ldots)} \sum_{t=1}^{\infty} Q(s_t, a_t^K) \left(\sum_{\tau=1}^{K} \nabla_\theta \log \pi_\theta(a_t^\tau|a_t^{\tau-1}, s_t) \right), \end{aligned} \tag{22}$$

where we denote $\pi_\theta(a_t^1|a_t^0, s) = \pi_\theta(a_t^1|s)$.

Proof. We directly plug in (8) in the simplified terms to the LHS to get the following:

$$\mathop{\mathbb{E}}_{(s_0,a_0^K,s_1,\ldots)} \sum_{t=1}^{\infty} Q(s_t, a_t) \nabla_\theta \log \left(\int_{a_t^{K-1}} \cdots \int_{a_t^1} \prod_{\tau=1}^{K} \pi(a_t^\tau|a_t^{\tau-1}, s_t) da^1 \cdots da^{K-1} \right).$$

While logs of sums/integrals do not expand, we can utilize the fact that this is the gradient of a log to re-expand and simplify to the following:

$$(LHS) = \mathop{\mathbb{E}}_{(s_0,a_0^K,s_1,\dots)} \sum_{t=1}^{\infty} Q(s_t,a_t^K) \frac{1}{\int_{a_t^{K-1}} \cdots \int_{a_t^1} \prod_{\tau=1}^K \pi(a_t^\tau|a_t^{\tau-1},s_t)} \tag{23}$$

$$\cdot \nabla_\theta \int_{a_t^{K-1}} \cdots \int_{a_t^1} \prod_{\tau=1}^K \pi(a_t^\tau|a_t^{\tau-1},s_t) da^1 \cdots da^{K-1},$$

$$= \mathop{\mathbb{E}}_{(s_0,a_0^K,s_1,\dots)} \sum_{t=1}^{\infty} Q(s_t,a_t^K) \frac{1}{\pi_\theta(a_t^K|s_t)} \int_{a_t^{K-1}} \cdots \int_{a_t^1} \nabla_\theta \prod_{\tau=1}^K \pi(a_t^\tau|a_t^{\tau-1},s_t) da^1 \cdots da^{K-1}, \tag{24}$$

$$= \mathop{\mathbb{E}}_{(s_0,a_0^K,s_1,\dots)} \sum_{t=1}^{\infty} \int_{a_t^{K-1}} \cdots \int_{a_t^1} Q(s_t,a_t^K) \frac{1}{\pi_\theta(a_t|s_t)} \left(\prod_{\tau=1}^K \pi(a_t^\tau|a_t^{\tau-1},s_t) \right) \tag{25}$$

$$\cdot \nabla_\theta \log(\prod_{\tau=1}^K \pi(a_t^\tau|a_t^{\tau-1},s_t)) da^1 \cdots da^{K-1},$$

$$= \mathop{\mathbb{E}}_{(s_0,s_1,s_2,\dots)} \sum_{t=1}^{\infty} \int_{a_t^K} \int_{a_t^{K-1}} \cdots \int_{a_t^1} \left(\prod_{\tau=1}^K \pi(a_t^\tau|a_t^{\tau-1},s_t) \right) \tag{26}$$

$$\cdot Q(s_t,a_t^K) \nabla_\theta \sum_{\tau=1}^K \log(\pi(a_t^\tau|a_t^{\tau-1},s_t)) da^1 \cdots da^K,$$

$$= \mathop{\mathbb{E}}_{(s_0,\{a_0^\tau\}_{\tau=1}^K,s_1,\dots)} \sum_{t=1}^{\infty} Q(s_t,a_t^K) \left(\sum_{\tau=1}^K \nabla_\theta \log(\pi(a_t^\tau|a_t^{\tau-1},s_t)) \right). \tag{27}$$

□

B.2. Proof of Theorem 4.1

Proof. We will use the shorthand $Q \coloneqq Q^\Psi$. Note that for any $\tau > 0$, we can decompose $Q(\mathbf{0},\mathbf{0})$ recursively into the following:

$$Q(\mathbf{0},\mathbf{0}) = \int_0^\infty \gamma^t r(s(t)) dt, \tag{28}$$

$$= \int_0^\tau \gamma^t r(s(t)) dt + \int_\tau^\infty \gamma^t r(s(t)) dt, \tag{29}$$

$$= \int_0^\tau \gamma^t r(s(t)) dt + \int_0^\infty \gamma^{t+\tau} r(s(t+\tau)) dt, \tag{30}$$

$$= \int_0^\tau \gamma^t r(s(t)) dt + \gamma^\tau \int_0^\infty \gamma^t r(s(t,s(\tau),a(\tau)) dt, \tag{31}$$

$$= \int_0^\tau \gamma^t r(s(t)) dt + \gamma^\tau Q(s(\tau),a(\tau)). \tag{32}$$

This will form the baseline of all of our local analysis, namely the fact that the Q-function can be broken down into two components:

1. a purely state-dependent integral that can be made arbitrarily small, and
2. a boundary term of the same Q-function.

We proceed with this proof with the contrapositive statement: suppose that, for some $t \in [0,\infty)$, it follows that $\Psi(s(t),a(t))$ is not aligned with $\nabla_a Q(s(t),a(t))$. We show that there exists a "bump" of Ψ towards $\nabla_a Q$ that strictly increases $Q(\mathbf{0},\mathbf{0})$.

Aligning at the origin. Suppose the above contrapositive statement yields $t = 0$, and assume that $\nabla_a Q(\mathbf{0},\mathbf{0}) \neq \mathbf{0}$, and that $\Psi(\mathbf{0},\mathbf{0})$ is not collinear with $\nabla_a Q(\mathbf{0},\mathbf{0})$. We will use this to construct a modified vector field Ψ' such that $Q^{\Psi'}(\mathbf{0},\mathbf{0}) > Q^\Psi(\mathbf{0},\mathbf{0})$. Suppose we have chosen τ and ϵ small enough such that the following all hold:

1. $s(\tau') = \tau' F(\mathbf{0}, \mathbf{0}) + o(\tau')$ for all $0 < \tau' \leq \tau$,
2. $a(\tau') = \tau' \Psi(\mathbf{0}, \mathbf{0}) + o(\tau')$ for all $0 < \tau' \leq \tau$,
3. $r(s) = r(\mathbf{0}) + \langle \nabla_s r(\mathbf{0}), s \rangle + o(\|s\|)$ for all $s, \|s\| \leq \epsilon$,
4. $Q(s,a) = Q(\mathbf{0},\mathbf{0}) + \langle \nabla_s Q(\mathbf{0},\mathbf{0}), s\rangle + \langle \nabla_a Q(\mathbf{0},\mathbf{0}), a\rangle + o(\|s\colon a\|)$ for all s, a such that $\|s\|, \|a\| \leq \epsilon$, where $(s\colon a) \in \mathbb{R}^{s\times a}$ is the stacked vector of s and a,
5. $\|s(\tau')\| \leq \epsilon, \|a(\tau')\| \leq \epsilon$ for all $0 < \tau' \leq \tau$.

We can then make the following approximations for $Q(\mathbf{0}, \mathbf{0})$:

$$
\begin{aligned}
Q(\mathbf{0},\mathbf{0}) &= \int_0^\tau \gamma^t r(s(t))dt + \gamma^\tau Q(s(\tau), a(\tau)), && (33)\\
&= \int_0^\tau \gamma^t \left(r(\mathbf{0}) + \langle \nabla_s r(\mathbf{0}), s(t)\rangle + o(\|s(t)\|)\right) dt && (34)\\
&\quad + \gamma^\tau \left(Q(\mathbf{0},\mathbf{0}) + \langle \nabla_s Q(\mathbf{0},\mathbf{0}), s(\tau)\rangle + \langle \nabla_a Q(\mathbf{0},\mathbf{0}), a(\tau)\rangle + o(\|s(\tau)\colon a(\tau)\|)\right),\\
&= \int_0^\tau \gamma^t \left(r(\mathbf{0}) + t\,\langle \nabla_s r(\mathbf{0}), F(\mathbf{0},\mathbf{0})\rangle + o(t)(\|F(\mathbf{0},\mathbf{0})\| + \|\nabla_s r(\mathbf{0})\|)\right) dt && (35)\\
&\quad + \gamma^\tau (Q(\mathbf{0},\mathbf{0}) + \tau\,\langle \nabla_s Q(\mathbf{0},\mathbf{0}), F(\mathbf{0},\mathbf{0})\rangle + \tau\,\langle \nabla_a Q(\mathbf{0},\mathbf{0}), \Psi(\mathbf{0},\mathbf{0})\rangle\\
&\quad + o(\tau)(\|\nabla_s Q(\mathbf{0},\mathbf{0})\| + \|\nabla_a Q(\mathbf{0},\mathbf{0})\| + \|F(\mathbf{0},\mathbf{0})\colon \Psi(\mathbf{0},\mathbf{0})\|)),\\
&= \int_0^\tau \gamma^t \left(r(\mathbf{0}) + t\,\langle \nabla_s r(\mathbf{0}), F(\mathbf{0},\mathbf{0})\rangle\right) dt && (36)\\
&\quad + \gamma^\tau (Q(\mathbf{0},\mathbf{0}) + \tau\,\langle \nabla_s Q(\mathbf{0},\mathbf{0}), F(\mathbf{0},\mathbf{0})\rangle + \tau\,\langle \nabla_a Q(\mathbf{0},\mathbf{0}), \Psi(\mathbf{0},\mathbf{0})\rangle)\\
&\quad + o(\tau).
\end{aligned}
$$

Note the three summands in (36) that $Q(\mathbf{0},\mathbf{0})$ now splits into:

1. An integral term purely dependent on the origin state dynamics $F(\mathbf{0},\mathbf{0})$,
2. A term with linearly separable dependence on the origin's state dynamics $\Psi(\mathbf{0},\mathbf{0})$, and
3. A term decaying strictly faster than our discretization term τ.

We then focus our attention on the second summand, as it at least appears to contain the separated dependence of the score Ψ. Note, however, that the Q-function itself still has dependence on Ψ, so we need to be careful. To finalize the separation of the score, we split into two cases:

1. Suppose there exists some $\epsilon > 0$ such that $a(t)$ will not re-cross the ϵ-ball around the origin, denoted B_ϵ, after its first escape, from any initial condition within B_ϵ. Thus, for this further reduced neighborhood ϵ (and likewise reduced τ), $Q(\mathbf{0},\mathbf{0})$ is completely determined by the local integral $\int_0^\tau \gamma^t r(s(t))dt$ while $(s(t), a(t))$ traverses through B_ϵ, and the boundary condition $\gamma^\tau Q(s(\tau), a(\tau))$ once it hits the ϵ-ball boundary. This gives us a neighborhood to manipulate Ψ: if the changes to Ψ only happen within B_ϵ, then the value of $Q(s,a)$ on the border of B_ϵ remains the same.

   Let $\phi : \mathbb{R}^a \to \mathbb{R}$ be a partition of unity with respect to B_ϵ: ϕ is a function purely of the norm $\|a\|$, $\phi(a) = 1$ for all $\|a\| \leq \frac{\epsilon}{2}$, and $\phi(a) = 0$ for all $\|a\| \geq \epsilon$. Consider the following modified score function:

$$
\Psi'(s,a) \coloneqq (1 - \phi(a))\Psi(s,a) + \phi(a)\frac{\|\Psi(\mathbf{0},\mathbf{0})\|}{\|\nabla_a Q(\mathbf{0},\mathbf{0})\|}\nabla_a Q(\mathbf{0},\mathbf{0}). \tag{37}
$$

   By construction, $\|\Psi\|_{\mathcal{C}^0} \coloneqq \sup_{s,a}\|\Psi(s,a)\| \leq C \implies \|\Psi'\|_{\mathcal{C}^0} \leq C$, and Ψ' is Lipschitz (of a likely higher but finite constant) as it is the sum and product of Lipschitz functions. Note that if $\Psi(\mathbf{0},\mathbf{0}) = \mathbf{0}$, then $\Psi' = (1 - \phi(a))\Psi$, in which case we replace $\|\Psi(\mathbf{0},\mathbf{0})\|$ in (37) with an appropriate constant such that $\|\Psi'\|_{\mathcal{C}^0} \leq C$.

We then get the following improvement of $Q(\mathbf{0},\mathbf{0})$, working around the fact that the Taylor approximation bounds with respect to ϵ, τ do not necessarily apply now for the action component of Ψ':

$$
\begin{aligned}
Q^{\Psi'}(\mathbf{0},\mathbf{0}) - Q^{\Psi}(\mathbf{0},\mathbf{0}) &= \gamma^\tau \left\langle \nabla_a Q(\mathbf{0},\mathbf{0}), a_{\Psi'}(\tau) - a_\Psi(\tau) \right\rangle + o(\tau), && (38)\\
&= \gamma^\tau \left\langle \nabla_a Q(\mathbf{0},\mathbf{0}), \int_0^\tau \Psi'(\mathbf{0}, a(\tau))dt - \tau\Psi(\mathbf{0},\mathbf{0}) \right\rangle \\
&\quad + o(\tau), && (39)\\
&= \gamma^\tau \left\langle \nabla_a Q(\mathbf{0},\mathbf{0}), \int_0^\tau \Psi'(\mathbf{0}, a(t))dt - \int_0^\tau \Psi(\mathbf{0},\mathbf{0})dt \right\rangle \\
&\quad + o(\tau), && (40)\\
&= \gamma^\tau \int_0^\tau \left\langle \nabla_a Q(\mathbf{0},\mathbf{0}), \Psi'(\mathbf{0}, a(t)) - \Psi(\mathbf{0},\mathbf{0}) \right\rangle dt \\
&\quad + o(\tau), && (41)\\
&= \gamma^\tau \ell(\tau) + o(\tau), && (42)
\end{aligned}
$$

where $\ell(\tau) \coloneqq \int_0^\tau \left\langle \nabla_a Q(\mathbf{0},\mathbf{0}), \Psi'(\mathbf{0}, a(\tau)) - \Psi(\mathbf{0},\mathbf{0}) \right\rangle dt$. Note that the desired inequality follows if $\ell(\tau) \geq M\tau$ for some $M > 0$, since this would imply $\lim_{\tau\to 0} \frac{o(\tau)}{\gamma^\tau \ell(\tau)} = 0$ and further imply there exists some τ such that $|\gamma^\tau \ell(\tau)| > |o(\tau)|$, and thus further that $Q^{\Psi'}(\mathbf{0},\mathbf{0}) > Q^{\Psi}(\mathbf{0},\mathbf{0})$.

It then remains to find a constant M such that $\ell(\tau) \geq M\tau$. Split the integral at hand as follows:

$$
\begin{aligned}
&\int_0^\tau \left\langle \nabla_a Q(\mathbf{0},\mathbf{0}), \Psi'(\mathbf{0}, a(\tau)) - \Psi(\mathbf{0},\mathbf{0}) \right\rangle dt && (43)\\
&= \int_0^{\tau_{\epsilon/2}} \left\langle \nabla_a Q(\mathbf{0},\mathbf{0}), \Psi'(\mathbf{0}, a(\tau)) - \Psi(\mathbf{0},\mathbf{0}) \right\rangle dt \\
&\quad + \int_{\tau_{\epsilon/2}}^\tau \left\langle \nabla_a Q(\mathbf{0},\mathbf{0}), \Psi'(\mathbf{0}, a(\tau)) - \Psi(\mathbf{0},\mathbf{0}) \right\rangle dt, \\
&= \tau_{\epsilon/2} \left\langle \nabla_a Q(\mathbf{0},\mathbf{0}), \frac{\|\Psi(\mathbf{0},\mathbf{0})\|}{\|\nabla_a Q(\mathbf{0},\mathbf{0})\|} \nabla_a Q(\mathbf{0},\mathbf{0}) \right\rangle && (44)\\
&\quad + \int_{\tau_{\epsilon/2}}^\tau \left\langle \nabla_a Q(\mathbf{0},\mathbf{0}), \Psi'(\mathbf{0}, a(\tau)) - \Psi(\mathbf{0},\mathbf{0}) \right\rangle dt, \\
&\geq \tau_{\epsilon/2} M' && (45)
\end{aligned}
$$

where $\tau_{\epsilon/2}$ is the hitting time of $a(t)$ at $\|a(t)\| = \epsilon/2$, $M' \coloneqq \left\langle \nabla_a Q(\mathbf{0},\mathbf{0}), \frac{\|\Psi(\mathbf{0},\mathbf{0})\|}{\|\nabla_a Q(\mathbf{0},\mathbf{0})\|} \nabla_a Q(\mathbf{0},\mathbf{0}) \right\rangle > 0$, and the second integral is non-negative by non-collinearity of $\nabla_a Q(\mathbf{0},\mathbf{0})$ and $\Psi(\mathbf{0},\mathbf{0})$, along with the construction of Ψ' as convex combinations of these two vectors. Finally, we can further find some $M > 0$ such that $\tau_{\epsilon/2} M' \geq M\tau$, since the flow of the original $a(t)$ is governed by a constant vector up to error $o(\tau)$, and thus the constructed ϵ scales at fastest linearly towards $\epsilon \to 0$ (and likewise for the modified $a_{\Psi'}(t)$ within $\epsilon/2$), noting that ϵ can be chosen separately and arbitrarily for the action dynamics if $\Psi(\mathbf{0},\mathbf{0}) = \mathbf{0}$. This then concludes that $\ell(\tau) \geq M\tau$ for some $M > 0$, and concludes this part of the proof.

2. Suppose however we cannot find such a further reduced ϵ (this can happen for example in certain attractor systems). We then leave ϵ as-is up to this point, and similarly construct Ψ' from (37). Let $\mathcal{T} \coloneqq \{T_1, T_1^e, T_2, T_2^e, \ldots\}$ be a monotonically increasing set of positive times with respect to the modified score Ψ' such that $a_{\Psi'}(t)$ crosses into B_ϵ at each time T_i and subsequently back out at time T_i^e (T_1 is the first re-entry of $a(t)$ *after* it first leaves from $t = 0$). Since $\nabla_a Q(\mathbf{0},\mathbf{0}) \neq 0$, the origin is not an equilibrium, and each T_i has a paired T_i^e.

   If $\mathcal{T}$ is empty (note this is technically still possible, as there can still exist *some* initial condition along the boundary of B_ϵ that does not re-enter), we repeat the analysis above from the existence of a non-re-enterable epsilon-ball B_ϵ.

If $\mathcal{T}$ is finite, we can make the following decomposition of $Q^{\Psi'}(\mathbf{0}, \mathbf{0})$ for some $k \in \mathbb{N}$:

$$\begin{aligned} Q^{\Psi'}(\mathbf{0},\mathbf{0}) &= \int_0^{T_1} \gamma^t r(s_{\Psi'}(t))dt + \int_{T_1}^{T_1^e} \gamma^t r(s_{\Psi'}(t))dt \\ &\quad + \int_{T_1^e}^{T_2} \gamma^t r(s_{\Psi'}(t))dt + \int_{T_2}^{T_2^e} \gamma^t r(s_{\Psi'}(t))dt \\ &\quad + \dots \\ &\quad + \int_{T_{k-1}^e}^{T_k} \gamma^t r(s_{\Psi'}(t))dt + \int_{T_k}^{T_k^e} \gamma^t r(s_{\Psi'}(t))dt + \int_{T_k^e}^{\infty} \gamma^t r(s_{\Psi'}(t))dt, \end{aligned} \tag{46}$$

$$\begin{aligned} &= \int_0^{T_1} \gamma^t r(s_{\Psi'}(t))dt + \int_{T_1}^{T_1^e} \gamma^t r(s_{\Psi'}(t))dt \\ &\quad + \dots \\ &\quad + \int_{T_{k-1}^e}^{T_k} \gamma^t r(s_{\Psi'}(t))dt + \int_{T_k}^{T_k^e} \gamma^t r(s_{\Psi'}(t))dt \\ &\quad + \gamma^{T_k^e} Q^{\Psi'}(s_{\Psi'}(T_k^e), a_{\Psi'}(T_k^e)), \end{aligned} \tag{47}$$

Focusing now on the last two summands of (47), we can make the following simplifications, with explanations below:

$$\int_{T_{k-1}^e}^{T_k} \gamma^t r(s_{\Psi'}(t))dt + \int_{T_k}^{T_k^e} \gamma^t r(s_{\Psi'}(t))dt + \gamma^{T_k^e} Q^{\Psi'}(s_{\Psi'}(T_k^e), a_{\Psi'}(T_k^e)), \tag{48}$$

$$= \int_{T_{k-1}^e}^{T_k} \gamma^t r(s_{\Psi'}(t))dt + \int_{T_k}^{T_k^e} \gamma^t r(s_{\Psi'}(t))dt + \gamma^{T_k^e} Q^{\Psi}(s_{\Psi'}(T_k^e), a_{\Psi'}(T_k^e)), \tag{49}$$

$$> \int_{T_{k-1}^e}^{T_k} \gamma^t r(s_{\Psi'}(t))dt + \int_{T_k}^{T_k^e} \gamma^t r(s_{\Psi'}(t))dt + \gamma^{T_k^e} Q^{\Psi}(s_{\Psi'_{T_k}}(T_k^e), a_{\Psi'_{T_k}}(T_k^e)), \tag{50}$$

$$\begin{aligned} &= \int_{T_{k-1}^e}^{T_k} \gamma^t r(s_{\Psi'}(t))dt \\ &\quad + \gamma^{T_k} \left(\int_0^{T_k^e - T_k} \gamma^t r(s_{\Psi'_{T_k}}(t))dt + o(\tau) + \gamma^{T_k^e - T_k} Q^{\Psi}(s_{\Psi'_{T_k}}(T_k^e), a_{\Psi'_{T_k}}(T_k^e)) \right), \end{aligned} \tag{51}$$

$$= \int_{T_{k-1}^e}^{T_k} \gamma^t r(s_{\Psi'}(t))dt + \gamma^{T_k} Q^{\Psi}(s_{\Psi'}(T_k), a_{\Psi'}(T_k)) + \gamma^{T_k} o(\tau), \tag{52}$$

$$= \int_{T_{k-1}^e}^{T_k} \gamma^t r(s_{\Psi'_{T_{k-1}^e}}(t))dt + \gamma^{T_k} Q^{\Psi}(s_{\Psi'}(T_k), a_{\Psi'}(T_k)) + \gamma^{T_k} o(\tau), \tag{53}$$

$$= \gamma^{T_{k-1}^e} Q^{\Psi}(s_{\Psi'}(T_{k-1}^e), a_{\Psi'}(T_{k-1}^e)) + \gamma^{T_k} o(\tau), \tag{54}$$

where $s_{\Psi'_T}(t)$ for $T < t$ is the resulting state $s(t)$ after integrating using Ψ' until T, then switching to Ψ.

(a) Equation 49 follows by definition of T_k^e being the last time the system $a(t)$ crosses B_ϵ, so $\Psi = \Psi'$ for the rest of the path and $Q^{\Psi'}(s_{\Psi'}(T_k^e), a_{\Psi'}(T_k^e)) = Q^{\Psi}(s_{\Psi'}(T_k^e), a_{\Psi'}(T_k^e))$.

(b) Equation 50 follows from the same analysis in part 1, since we can now fix our analysis on the originally Taylor-approximated Q^{Ψ}. Note that we now need to integrate Ψ' through the whole neighborhood B_ϵ rather than just the sub-neighborhood $B_{\epsilon/2}$ where Ψ' is constant, but the analysis remains the same, noting the linearity at (34) even before taking an approximation of the dynamical system's flow.

(c) Equation 51 comes from a reparamaterization of integrals, and using the analysis for (36) to get an approximation of the internal reward integral $\int_{T_k}^{T_k^e} \gamma^t r(s_{\Psi'}(t))dt$ that is unchanged with respect to the action dynamics Ψ and Ψ', with error $o(\tau)$. Note τ here is the max difference between T_i and T_i^e; this can be arbitrarily decreased with ϵ since Ψ is Lipschitz.

(d) Equation 52 comes from wrapping the recursive equation for the Q-function (see (32)), noting we have now removed dependence on Ψ' from time T_k onwards.

(e) Equation 53 comes from the fact that, by construction of $\mathcal{T}$, $\Psi' = \Psi$ on the time interval $[T^e_{k-1}, T_k]$.

(f) Finally, (54) is another wrapping of the recursive equation for the Q-function.

The proof strategy now becomes clear: we can iteratively repeat the process of (49) through (54) to eventually write the RHS in terms of Q^Ψ and a rapidly decaying $o(\tau)$ term, achieving the desired inequality.

Repeat the above process inductively, where we conclude with the following:

$$
\begin{aligned}
Q^{\Psi'}(\mathbf{0},\mathbf{0}) &> \int_0^{T_1} \gamma^t r(s_{\Psi'}(t))dt + \int_{T_1}^{T_1^e} \gamma^t r(s_{\Psi'}(t))dt && (55)\\
&\quad + \gamma^{T_1^e} Q^\Psi(s_{\Psi'}(T_1^e), a_{\Psi'}(T_1^e)) + (\sum_{i=2}^{k} \gamma^{T_i}) o(\tau),\\
&> \int_0^{T_1} \gamma^t r(s_{\Psi'}(t))dt + \int_{T_1}^{T_1^e} \gamma^t r(s_{\Psi'}(t))dt && (56)\\
&\quad + \gamma^{T_1^e} Q^\Psi(s_{\Psi'_{T_1}}(T_1^e), a_{\Psi'_{T_1}}(T_1^e)) + (\sum_{i=2}^{k} \gamma^{T_i}) o(\tau),\\
&= \int_0^{\tau} \gamma^t r(s_{\Psi'}(t))dt + \int_{\tau}^{T_1} \gamma^t r(s_{\Psi'}(t))dt && (57)\\
&\quad + \gamma^{T_1} Q^\Psi(s_{\Psi'}(T_1), a_{\Psi'}(T_1)) + (\sum_{i=1}^{k} \gamma^{T_i}) o(\tau),\\
&= \int_0^{\tau} \gamma^t r(s_{\Psi'}(t))dt + \gamma^\tau Q^\Psi(s_{\Psi'}(\tau), a_{\Psi'}(\tau)) + (\sum_{i=1}^{k} \gamma^{T_i}) o(\tau), && (58)\\
&= Q^{\Psi'}(\mathbf{0},\mathbf{0}) + \gamma^\tau \ell(\tau) + (1 + \sum_{i=1}^{k} \gamma^{T_i}) o(\tau). && (59)
\end{aligned}
$$

Since $\lim_{\tau\to 0} \frac{(1+\sum_{i=1}^k \gamma^{T_k}) o(\tau)}{\gamma^\tau \ell(\tau)} = 0$, there exists some τ such that the above equation implies $Q^{\Psi'}(\mathbf{0},\mathbf{0}) > Q^\Psi(\mathbf{0},\mathbf{0})$.

If $\mathcal{T}$ is infinite, we no longer have our "base case" for the above induction. However, by using the decaying structure of the Q-function, we can artificially create a base case. Noting that $0 \le Q^\Psi(s,a), Q^{\Psi'}(s,a) \le \int_0^\infty \gamma^t dt = (\log(\gamma^{-1}))^{-1}$, we can write the following for any fixed k for some error term $q_k \in \mathbb{R}$:

$$
\gamma^{T_k^e} Q^{\Psi'}(s_{\Psi'}(T_k^e), a_{\Psi'}(T_k^e)) = \gamma^{T_k^e} Q^\Psi(s_{\Psi'}(T_k^e), a_{\Psi'}(T_k^e)) + \gamma^{T_k^e} q_k, \tag{60}
$$

where each $|q_k| \le (\log(\gamma^{-1}))^{-1}$. Thus, we can repeat the analysis from the finite $\mathcal{T}$ case to get the following:

$$
Q^{\Psi'}(\mathbf{0},\mathbf{0}) > Q^\Psi(\mathbf{0},\mathbf{0}) + \gamma^\tau \ell(\tau) + (1 + \sum_{i=1}^{k} \gamma^{T_i}) o(\tau) + \gamma^{T_k^e} q_k. \tag{61}
$$

Since $\sum_{i=1}^\infty \gamma^{T_i} < \infty$, it follows that $\lim_{\tau\to 0} \frac{(1+\sum_{i=1}^\infty \gamma^{T_i}) o(\tau)}{\gamma^\tau \ell(\tau)} = 0$, and there exists some τ and some constant κ such that $\gamma^\tau \ell(\tau) + (1 + \sum_{i=1}^k \gamma^{T_i}) o(\tau) > \kappa > 0$, regardless of what value of k is chosen. Since $\mathcal{T}$ is infinite and q_k is bounded, we can choose some k' such that $|\gamma^{T^e_{k'}} q_{k'}| < \kappa$. Under such chosen k', it follows that $Q^{\Psi'}(\mathbf{0},\mathbf{0}) > Q^\Psi(\mathbf{0},\mathbf{0})$.

Aligning beyond the origin. Suppose that Ψ is aligned with $\nabla_a Q$ until some positive $t' > 0$. Split $Q(\mathbf{0},\mathbf{0}) = \int_0^{t'-\tau} r(s(t))dt + \gamma^{t'-\tau}\left(\int_0^\tau \gamma^t r(s(t+t'-\tau))dt + \gamma^\tau Q(s(t),a(t))\right)$, and repeat the analysis above on $Q(s(t),a(t))$, noting both that the first summand is unaffected by changing Ψ to Ψ' and that τ can be chosen small enough for $\int_0^\tau \gamma^t r(s(t+t'-\tau))dt$ to only be dependent on state dynamics, with error $o(\tau)$. □

Extensions. A trivial extension of Theorem 4.1 is to any starting condition $Q(s_0, a_0)$, and one we can cover briefly is Corollary 4.2, where we consider expectations over initial conditions of the form $\mathbb{E}_{\mathbb{P}\times\pi} Q^{\Psi}(s, a)$. Since the analysis for Theorem 4.1 made perturbations in strictly positive measure regions, we can repeat the above analysis to nudge Ψ at any non-aligned point (s, a) to not only conclude a strict increase in Q at (s, a), but in a measure-positive region around (s, a), and since this change does not decrease $Q(s', a')$ at any other pair (s', a') by repeating the "aligning beyond the origin" analysis, such constructions of updated scores Ψ' in the proof of Theorem 4.1 similarly conclude Corollary 4.2.

B.3. Proof of Theorem 4.3

Proof. Firstly, let's formalize our notion of the support of this full joint distribution with the initial condition distribution and the stochastic dynamics of $s(t), a(t)$:

Definition B.2. A point $(s, a) \in \mathbb{R}^s \times \mathbb{R}^a$ is in the support of the fully joint distribution between initial condition distribution $\mathbb{P}, \pi$ and stochastic dynamics of $(s(t), a(t))$ if for every $\epsilon > 0$, the probability that $a(t), s(t) \in B_\epsilon(s, a)$ at some finite time t is strictly greater than 0.

Using this definition, we then pave a clear path forward. We first repeat the first part analysis from the proof of Theorem 4.1; namely, we first focus on purely $Q(\mathbf{0}, \mathbf{0})$ and the case where the misalignment happens at the origin, then build up to the theorem with straightforward corollaries.

We then first focus on $Q(\mathbf{0}, \mathbf{0})$ and assume that $\nabla_a Q(\mathbf{0}, \mathbf{0})$ is not collinear with $\Psi(\mathbf{0}, \mathbf{0})$, using this to construct a score Ψ' such that $Q^{\Psi'}(\mathbf{0}, \mathbf{0}) > Q^{\Psi}(\mathbf{0}, \mathbf{0})$.

We start with our new recursive equation for the Q-function for any $\tau > 0$. Note that any expectations $\mathbb{E}$ are taken with respects to the stochastic dynamics of $s(t), a(t)$.

$$
\begin{aligned}
Q(\mathbf{0}, \mathbf{0}) &= \mathbb{E} \int_0^\infty \gamma^t r(s(t)) dt, && (62)\\
&= \mathop{\mathbb{E}}_{s,a}[\mathbb{E}[\int_0^\infty \gamma^t r(s(t)) dt \mid (s(\tau), a(\tau)) = (s, a)]], && (63)\\
&= \mathop{\mathbb{E}}_{s,a}[\mathbb{E}[\int_0^\tau \gamma^t r(s(t)) dt + \gamma^\tau Q(s, a) \mid (s(\tau), a(\tau)) = (s, a)]], && (64)\\
&= \mathop{\mathbb{E}}_{s,a}[\mathbb{E}[\int_0^\tau \gamma^t r(s(t)) dt \mid (s(\tau), a(\tau)) = (s, a)]] \\
&\quad + \gamma^\tau \mathop{\mathbb{E}}_{s,a}[\mathbb{E}[Q(s, a) \mid (s(\tau), a(\tau)) = (s, a)]], && (65)\\
&= \mathbb{E}[\int_0^\tau \gamma^t r(s(t)) dt] + \gamma^\tau \mathop{\mathbb{E}}_{s,a}[Q(s, a) \mid (s(\tau), a(\tau)) = (s, a)]. && (66)
\end{aligned}
$$

Our new conditions for ϵ, τ will be the following:

1. For all $\tau' \leq \tau$, the distribution of $a(\tau')$ is close to $\mathcal{N}(\tau'\Psi(\mathbf{0}, \mathbf{0}), \tau'\Sigma_a(\mathbf{0}, \mathbf{0}))$ in the following sense: for any fixed smooth $f : \mathbb{R}^a \to \mathbb{R}$, the difference between $\mathbb{E}[f(a(\tau'))]$ and $\mathbb{E}_{N(\tau'\Psi(\mathbf{0},\mathbf{0}),\tau')} f(a(\tau'))$ is controlled by some little-o function $o_f(\tau')$: $\mathbb{E}[f(a(\tau'))] - \mathbb{E}_{N(\tau'\Psi(\mathbf{0},\mathbf{0}),\tau'\Sigma_a(\mathbf{0},\mathbf{0}))}[f(a(\tau))] = o_f(\tau')$,

2. $s(\tau')$ is close in distribution to $\mathcal{N}(\tau F(\mathbf{0}, \mathbf{0}), \tau\Sigma_s(\mathbf{0}, \mathbf{0}))$ with high probability in a similar sense to the above: for any fixed smooth $f : \mathbb{R}^s \to \mathbb{R}$, $\mathbb{E}[f(s(\tau'))] - \mathbb{E}_{N(\tau F(\mathbf{0},\mathbf{0}),\tau\Sigma_s(\mathbf{0},\mathbf{0}))}[f(s(\tau))] = o_f(\tau')$,

3. $r(s) = r(\mathbf{0}) + \langle \nabla_s r(\mathbf{0}), s\rangle + o(\|s\|)$ for all s, $\|s\| \leq \epsilon$.

4. $Q(s, a) = Q(\mathbf{0}, \mathbf{0}) + \langle \nabla_s Q(\mathbf{0}, \mathbf{0}), s\rangle + \langle \nabla_a Q(\mathbf{0}, \mathbf{0}), a\rangle + o(\|s\colon a\|)$ for all s, a such that $\|s\|, \|a\| \leq \epsilon$.

5. τ is chosen small enough so that the distribution of paths is essentially similar to restricting the norm of each of $s(\tau), a(\tau)$ to be at most ϵ in the following sense: for any fixed smooth $f : \mathbb{R}^s \times \mathbb{R}^a \to \mathbb{R}$, it follows that $\mathbb{E}[f(s(\tau), a(\tau))] - \mathbb{E}_{\|s(\tau)\|,\|a(\tau)\|\leq\epsilon}[f(s(\tau), a(\tau))] = o_f(\tau)$.

We now repeat the analysis in (34) to (36) to get the following:

$$
\begin{aligned}
Q(\mathbf{0},\mathbf{0}) &= \mathbb{E}[\int_0^\tau \gamma^t r(s(t))dt] + \gamma^\tau \mathop{\mathbb{E}}_{s,a}[Q(s,a) \mid (s(\tau),a(\tau)) = (s,a)] && (67)\\
&= \mathop{\mathbb{E}}_{\|s(t)\|,\|a(t)\|\le\epsilon}[\int_0^\tau \gamma^t r(s(t))dt] + \gamma^\tau \mathop{\mathbb{E}}_{\|s\|,\|a\|\le\epsilon}[Q(s,a) \mid (s(\tau),a(\tau)) = (s,a)] \\
&\quad + (\log(\gamma^{-1}))^{-1} o(\tau), && (68)\\
&= \mathop{\mathbb{E}}_{\|s(t)\|,\|a(t)\|\le\epsilon} \int_0^\tau \gamma^t \left(r(\mathbf{0}) + t\left\langle \nabla_s r(\mathbf{0}), F(\mathbf{0},\mathbf{0})\right\rangle\right) dt && (69)\\
&\quad + \gamma^\tau \mathop{\mathbb{E}}_{\|s(\tau)\|,\|a(\tau)\|\le\epsilon}[Q(\mathbf{0},\mathbf{0}) + \langle \nabla_s Q(\mathbf{0},\mathbf{0}), s(\tau)\rangle + \langle \nabla_a Q(\mathbf{0},\mathbf{0}), a(\tau)\rangle] \\
&\quad + o(\tau), \\
&= \int_0^\tau \gamma^t \left(r(\mathbf{0}) + t\left\langle \nabla_s r(\mathbf{0}), F(\mathbf{0},\mathbf{0})\right\rangle\right) dt && (70)\\
&\quad + \gamma^\tau \mathbb{E}\left[Q(\mathbf{0},\mathbf{0}) + \langle \nabla_s Q(\mathbf{0},\mathbf{0}), s(\tau)\rangle + \langle \nabla_a Q(\mathbf{0},\mathbf{0}), a(\tau)\rangle\right] \\
&\quad + o(\tau), \\
&= \int_0^\tau \gamma^t \left(r(\mathbf{0}) + t\left\langle \nabla_s r(\mathbf{0}), F(\mathbf{0},\mathbf{0})\right\rangle\right) dt && (71)\\
&\quad + \gamma^\tau (Q(\mathbf{0},\mathbf{0}) + \tau\left\langle \nabla_s Q(\mathbf{0},\mathbf{0}), F(\mathbf{0},\mathbf{0})\right\rangle + \tau\left\langle \nabla_a Q(\mathbf{0},\mathbf{0}), \Psi(\mathbf{0},\mathbf{0})\right\rangle) \\
&\quad + \gamma^\tau \sqrt{\tau}(\left\langle \nabla_s Q(\mathbf{0},\mathbf{0}), \mathop{\mathbb{E}}_{z\sim\mathcal{N}(\mathbf{0},\Sigma_s(\mathbf{0},\mathbf{0}))} z\right\rangle + \left\langle \nabla_a Q(\mathbf{0},\mathbf{0}), \mathop{\mathbb{E}}_{z\sim\mathcal{N}(\mathbf{0},\Sigma_a(\mathbf{0},\mathbf{0})} z\right\rangle) + o(\tau), \\
&= \int_0^\tau \gamma^t \left(r(\mathbf{0}) + t\left\langle \nabla_s r(\mathbf{0}), F(\mathbf{0},\mathbf{0})\right\rangle\right) dt && (72)\\
&\quad + \gamma^\tau (Q(\mathbf{0},\mathbf{0}) + \tau\left\langle \nabla_s Q(\mathbf{0},\mathbf{0}), F(\mathbf{0},\mathbf{0})\right\rangle + \tau\left\langle \nabla_a Q(\mathbf{0},\mathbf{0}), \Psi(\mathbf{0},\mathbf{0})\right\rangle \\
&\quad + o(\tau).
\end{aligned}
$$

Since (72) is exactly the same as (36), we can proceed exactly as we did in the non-stochastic case (proof of Theorem 4.1), conditioning on events and using the tower rule to exploit specific path structure as we did in the non-stochastic case, i.e. by conditioning the expectation for Q on the number of times each sampled path $(s(t), a(t))$ re-crosses the ϵ-ball at the origin. Definition B.2, along with previous analysis, ensure that any such local perturbations Ψ' result in a strict increase of the Q-function, both at a fixed starting condition and a distribution over starting conditions. □