Maximum-Entropy Exploration with Future State-Action Visitation Measures

Adrien Bolland, Gaspard Lambrechts, Damien Ernst

Keywords: Maximum-Entropy RL, Exploration, Policy Gradients, Occupancy Measures.

Summary

Maximum entropy reinforcement learning motivates agents to explore states and actions to maximize the entropy of some distribution, typically by providing additional intrinsic rewards proportional to that entropy function. In this paper, we study intrinsic rewards proportional to the entropy of the discounted distribution of state-action features visited during future time steps. This approach is motivated by two results. First, we show that the expected sum of these intrinsic rewards is a lower bound on the entropy of the discounted distribution of state-action features visited in trajectories starting from the initial states, which we relate to an alternative maximum entropy objective. Second, we show that the distribution used in the intrinsic reward definition is the fixed point of a contraction operator and can therefore be estimated off-policy. Experiments highlight that the new objective often leads to improved visitation of features within individual trajectories, in exchange for slightly reduced visitation of features in expectation over different trajectories, as suggested by the lower bound. It also leads to improved convergence speed for learning exploration-only agents. Control performance remains similar across most methods on the considered benchmarks.

Contribution(s)

1. This paper introduces a new MaxEntRL objective in which the intrinsic rewards that regularize the objective are proportional to the entropy of the discounted distribution of state-action features visited during future time steps; it motivates agents to explore these future features.
   Context: Prior MaxEntRL objectives typically regularize policy entropy or the entropy of the state visitation/occupancy instead.
2. The paper shows that the expected sum of these intrinsic rewards is a lower bound on the entropy of the discounted distribution of state-action features visited in trajectories starting from the initial states (equivalently, an alternative MaxEntRL objective).
   Context: This result situates the proposed objective as a surrogate for this alternative MaxEntRL objective and highlights a trade-off between the two.
3. The paper proves that the conditional feature distribution used in the intrinsic reward is the unique fixed point of a contractive operator, which justifies learning an estimate of this distribution via TD-style bootstrapping with N-step transitions and leads to an off-policy estimation scheme for the intrinsic rewards.
   Context: The fixed point and its derivation closely resemble those of successor-style representations of future state visitation (Janner et al., 2020); this structure can be leveraged for estimating MaxEntRL intrinsic rewards off-policy, which is otherwise often difficult.
4. The paper provides an off-policy algorithm for maximizing the new MaxEntRL objective; the reported results highlight a trade-off between within-trajectory and marginal feature visitation, depending on the MaxEntRL objective, and emphasize a step toward improving sample efficient intrinsic reward estimation.
   Context: The best exploration objective to pursue is still an open question despite the vast literature, and sample efficient intrinsic reward estimation is mostly ignored in the literature.

Maximum-Entropy Exploration with Future State-Action Visitation Measures

Adrien Bolland[1], **Gaspard Lambrechts**[2,3], **Damien Ernst**[1]
adrien.bolland@uliege.be, gaspard.lambrechts@mcgill.ca, dernst@uliege.be

[1]**University of Liège, Belgium**
[2]**McGill University, Canada**
[3]**Mila Québec Artificial Intelligence Institute, Canada**

Abstract

Maximum entropy reinforcement learning motivates agents to explore states and actions to maximize the entropy of some distribution, typically by providing additional intrinsic rewards proportional to that entropy function. In this paper, we study intrinsic rewards proportional to the entropy of the discounted distribution of state-action features visited during future time steps. This approach is motivated by two results. First, we show that the expected sum of these intrinsic rewards is a lower bound on the entropy of the discounted distribution of state-action features visited in trajectories starting from the initial states, which we relate to an alternative maximum entropy objective. Second, we show that the distribution used in the intrinsic reward definition is the fixed point of a contraction operator and can therefore be estimated off-policy. Experiments highlight that the new objective often leads to improved visitation of features within individual trajectories, in exchange for slightly reduced visitation of features in expectation over different trajectories, as suggested by the lower bound. It also leads to improved convergence speed for learning exploration-only agents. Control performance remains similar across most methods on the considered benchmarks.

1 Introduction

Many challenging tasks where an agent makes sequential decisions have been solved with reinforcement learning (RL). Examples range from playing games (Mnih et al., 2015; Silver et al., 2017) to managing energy systems and markets (Boukas et al., 2021; Aittahar et al., 2024). In practice, many RL algorithms are applied in combination with an exploration strategy to achieve high-performance control. These exploration strategies usually consist of providing intrinsic reward bonuses to the agent for achieving certain behaviors. Typically, the bonus enforces taking actions that reduce the uncertainty about the environment (Pathak et al., 2017; Burda et al., 2018; Zhang et al., 2021b), or actions that enhance the variety of states and actions in trajectories (Bellemare et al., 2016; Lee et al., 2019; Guo et al., 2021; Williams & Peng, 1991; Haarnoja et al., 2019). In many of the latter methods, the intrinsic reward function is the entropy of some distribution over the state-action space. Optimizing jointly the reward function of the MDP and the intrinsic reward function is called Maximum Entropy RL (MaxEntRL) and was shown to be effective in many problems.

The reward of the MDP was already extended with the entropy of the policy in early algorithms (Williams & Peng, 1991) and was only later called MaxEntRL (Ziebart et al., 2008; Toussaint, 2009). This particular reward regularization provides substantial improvements in the robustness of the resulting policy (Ziebart, 2010; Husain et al., 2021; Brekelmans et al., 2022) and provides a learning objective function with good smoothness and concavity properties (Ahmed et al., 2019; Bolland et al., 2023). Well-known algorithms include soft Q-learning (Haarnoja et al., 2017; Schulman et al., 2017a) and soft actor-critic (Haarnoja et al., 2018; 2019). This MaxEntRL objective nevertheless only rewards the randomness of actions and neglects the influences of the policy on the visited states.

In order to enhance exploration, Hazan et al. (2019) were the first to intrinsically motivate agents to visit different states. It led to maximizing the entropy of the discounted state visitation measure and the stationary state visitation measure. For discrete state-action spaces, optimal exploration policies, which maximize the entropy of these visitation measures, can be computed to near optimality with off-policy tabular model-based RL algorithms (Hazan et al., 2019; Mutti & Restelli, 2020; Tiapkin et al., 2023). For continuous state-action spaces, alternative methods rely on k-nearest neighbors to estimate the density of the visitation measure of states (or features built from the states) and compute the intrinsic rewards, which can afterward be optimized with any RL algorithm (Liu & Abbeel, 2021; Yarats et al., 2021; Seo et al., 2021; Mutti et al., 2021). These methods require sampling new trajectories at each iteration; they are on-policy, and estimating the intrinsic reward function is computationally expensive. Some other methods rely on parametric density estimators to reduce the computational complexity and share information across learning steps (Lee et al., 2019; Guo et al., 2021; Islam et al., 2019; Zhang et al., 2021a). This additional function is typically learned by maximum likelihood estimation using batches of truncated trajectories. Alternative methods have adapted this MaxEntRL objective to maximize the entropy of states visited in single trajectories (Mutti et al., 2022; Jain et al., 2024). When large and/or continuous state and action spaces are involved, relying on parametric function approximators is likely the best choice. Nevertheless, existing algorithms are again on-policy. They require sampling new trajectories from the environment at (nearly) every update of the policy, and cannot be applied using a buffer of arbitrary transitions, in batch-mode RL, or in continuing tasks. Furthermore, the discounted visitation measure is more desirable to learn than the stationary one, but may be challenging due to the exponentially decreasing influence of the time step at which states are visited (Islam et al., 2019).

The main contribution of this paper is to introduce a MaxEntRL objective relying on a new intrinsic reward function for exploring effectively the state and action spaces, which also alleviates the previous limitations. This intrinsic reward function is the relative entropy of the discounted distribution of state-action features visited during future time steps. We prove two results motivating the MaxEntRL objective. First, this new objective is a lower bound on the MaxEntRL objective previously described, which integrates the marginal visitation distribution of states and actions. Second, the visitation distribution used in the new intrinsic reward function is the fixed point of a contraction operator. The intrinsic reward can therefore efficiently be computed off-policy using N-step state-action transitions and bootstrapping the operator. It is then possible to approximate the intrinsic reward function and learn a policy maximizing the extended rewards with existing RL algorithms. We demonstrate in our experiments how to adapt soft actor-critic (Haarnoja et al., 2018) to optimize the new objective, and we compare the effectiveness of exploration with different maximum entropy objectives. Experiments highlight improved visitation of features within individual trajectories, in exchange for slightly reduced visitation of features in expectation over different trajectories, as suggested by the lower bound. It also leads to improved convergence speed for learning exploration-only agents. Control performance remains similar across most methods on the considered benchmarks.

The visitation measure of future states and actions, which we use to extend the reward function in this article, has a well-established history in the development of RL algorithms. It was popularized by Janner et al. (2020), who learned the distribution of future states as a generalization of the successor features (Barreto et al., 2017). They demonstrated that this distribution allows expressing the state-action value function by separating the influence of the dynamics and the reward function, and that it could be learned off-policy by exploiting its recursive expression. Several algorithms have been proposed to learn this distribution, using maximum likelihood estimation (Janner et al., 2020), contrastive learning (Mazoure et al., 2023b), or diffusion models (Mazoure et al., 2023c). These distributions of future states and actions have found applications in goal-based RL (Eysenbach et al., 2020; 2022), in offline pre-training with expert examples (Mazoure et al., 2023a), in model-based RL (Ma et al., 2023), or in planning (Eysenbach et al., 2023). We are the first to integrate them into the MaxEntRL framework for enhancing exploration. In parallel with our work, Mohamed et al. (2025) developed a similar intrinsic reward dependent on the distribution of future states.

The manuscript is organized as follows. In Section 2, the RL problem and the MaxEntRL framework are formulated. In Section 3, we introduce and discuss a new MaxEntRL objective. Section 4 details how to learn a model of the conditional state visitation probability that allows estimating this new objective. We finally present experimental results in Section 5 and conclude in Section 6.

2 Background and Preliminaries

2.1 Markov Decision Processes

This paper focuses on problems in which an agent makes sequential decisions in a stochastic environment (Sutton & Barto, 2018). The environment is modeled with an infinite-time Markov decision process (MDP) composed of a state space $\mathcal{S}$, an action space $\mathcal{A}$, an initial state distribution p_0, a transition distribution p_1, a bounded reward function R, and a discount factor $\gamma \in [0, 1)$. Agents interact in this MDP by providing actions sampled from a policy π. During this interaction, an initial state $s_0 \sim p_0(\cdot)$ is first sampled before the agent provides, at each time step t, an action $a_t \sim \pi(\cdot|s_t)$ leading to a new state $s_{t+1} \sim p_1(\cdot|s_t, a_t)$. In addition, after each action a_t is executed, a reward $r_t = R(s_t, a_t) \in \mathbb{R}$ is observed. We denote the expected return of the policy π by

$$J(\pi) = \underset{\substack{s_0 \sim p_0(\cdot) \\ a_t \sim \pi(\cdot|s_t) \\ s_{t+1} \sim p_1(\cdot|s_t, a_t)}}{\mathbb{E}} \left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t) \right] . \tag{1}$$

An optimal policy π^* is one with maximum expected return

$$\pi^* \in \arg\max_{\pi} J(\pi) . \tag{2}$$

2.2 Maximum Entropy Reinforcement Learning

In maximum entropy reinforcement learning (MaxEntRL), an optimal policy π^* is approximated by maximizing a surrogate objective function $L(\pi)$, where the reward function from the MDP is extended by an intrinsic reward function. The latter is the (negated relative) entropy of some particular distribution. A general definition of the MaxEntRL objective function is

$$L(\pi) = \underset{\substack{s_0 \sim p_0(\cdot) \\ a_t \sim \pi(\cdot|s_t) \\ s_{t+1} \sim p_1(\cdot|s_t, a_t)}}{\mathbb{E}} \left[\sum_{t=0}^{\infty} \gamma^t \left(R(s_t, a_t) + \lambda R^{int}(s_t, a_t) \right) \right] , \tag{3}$$

where this objective depends on the intrinsic reward function R^{int}. We propose a generic formulation that, to the best of our knowledge, encompasses most existing MaxEntRL intrinsic rewards. Given a feature space $\mathcal{Z}$, a conditional feature distribution $q^\pi : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{Z})$, depending on the policy π, and a relative measure $q^* \in \Delta(\mathcal{Z})$, the intrinsic reward function is

$$R^{int}(s, a) = -KL_z \left[q^\pi(z|s, a) \| q^*(z) \right] = \underset{z \sim q^\pi(\cdot|s,a)}{\mathbb{E}} \left[\log q^*(z) - \log q^\pi(z|s, a) \right] . \tag{4}$$

Importantly, the intrinsic reward function is (implicitly) dependent on the policy π through the distribution q^π. We define an optimal exploration policy as a policy that maximizes the expected sum of discounted intrinsic rewards only. Note that a policy maximizing $L(\pi)$ is generally not optimal, due to the potential gap between the optimum of the return $J(\pi)$ and the optimum of the learning objective $L(\pi)$. This subject is inherent to exploration with intrinsic rewards (Bolland et al., 2024).

MaxEntRL algorithms optimize objective functions as defined in equation (3) depending on some intrinsic reward function that can be expressed as in equation (4). Each algorithm provides a particular estimate of the intrinsic reward and introduces a particular approach for maximizing the learning objective. Often, pseudo rewards are computed to extend the MDP rewards, which are jointly optimized, typically with (biased) policy gradient algorithms.

Many of the existing MaxEntRL algorithms optimize an objective that depends on the entropy of the policy for exploring the action space (Haarnoja et al., 2018; Toussaint, 2009). Features then correspond to actions such that the feature space is $\mathcal{Z} = \mathcal{A}$ and the conditional feature distribution is the policy, $q^\pi(z|s,a) = \pi(z|s)$ for all a. Other algorithms optimize objectives enhancing state space exploration (Hazan et al., 2019; Lee et al., 2019; Islam et al., 2019; Guo et al., 2021). The feature space is the state space $\mathcal{Z} = \mathcal{S}$. The conditional feature distribution $q^\pi(z|s,a)$ is either the marginal probability of states in trajectories of T time steps, or the discounted state visitation measure, for all s and a. In the literature, the relative measure $q^*(z)$ is usually a uniform distribution.

3 MaxEntRL with Visitation Distributions

3.1 Definition of the MaxEntRL Objective

In the following, we introduce a new MaxEntRL intrinsic reward based on the conditional state-action visitation probability $d^{\pi,\gamma}(\bar{s},\bar{a}|s,a)$ and the conditional state visitation probability $d^{\pi,\gamma}(\bar{s}|s,a)$

$$d^{\pi,\gamma}(\bar{s},\bar{a}|s,a) = (1-\gamma)\pi(\bar{a}|\bar{s}) \sum_{\Delta=1}^{\infty} \gamma^{\Delta-1} p_\Delta^\pi(\bar{s}|s,a) \tag{5}$$

$$d^{\pi,\gamma}(\bar{s}|s,a) = (1-\gamma) \sum_{\Delta=1}^{\infty} \gamma^{\Delta-1} p_\Delta^\pi(\bar{s}|s,a)\,, \tag{6}$$

where p_Δ^π is the transition probability in Δ time steps with the policy π. The distribution from equation (5) can be factorized as a function of the distribution from equation (6) such that $d^{\pi,\gamma}(\bar{s},\bar{a}|s,a) = \pi(\bar{a}|\bar{s}) d^{\pi,\gamma}(\bar{s}|s,a)$. The conditional state (respectively, state-action) visitation probability distribution measures the states (respectively, states and actions) that are visited in expectation over infinite trajectories starting from a state and an action. Both distributions extend the (marginal discounted) state visitation/occupancy probability (Manne, 1960), defined as $d^{\pi,\gamma}(\bar{s}) = (1-\gamma)\sum_{t=0}^{\infty} \gamma^t p_t^\pi(\bar{s})$, where p_t^π is the state probability at time t under p_0 and π.

Definition 3.1. Let us consider the feature space $\mathcal{Z}$ and the feature distribution $h : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{Z})$. The intrinsic reward is defined by equation (4), for any relative measure q^*, with the distribution

$$q^\pi(z|s,a) = \int h(z|\bar{s},\bar{a}) d^{\pi,\gamma}(\bar{s},\bar{a}|s,a)\, d\bar{s}\, d\bar{a}\,. \tag{7}$$

Optimal exploration policies are here intrinsically motivated to take actions so that the discounted visitation measure of future features is distributed according to q^* in each state and for each action. It allows selecting features that must be visited during trajectories according to prior knowledge about the problem if any. Alternatively, it allows exploring lower-dimensional feature spaces, or exploring some statistics from the state-action pairs.

Existing RL algorithms can be used to compute MaxEntRL policies following Definition 3.1 by computing for each state s and action a the additional (pseudo) reward

$$R^{int}(s,a) = \log q^*(z) - \log q^\pi(z|s,a)\,, \tag{8}$$

where $z \sim q^\pi(\cdot|s,a)$. This reward is a single-sample Monte-Carlo estimate of equation (4), unbiased for fixed q^π. This computation requires sampling features z from the conditional distribution q^π and estimating the probability of these samples $q^\pi(z|s,a)$. It can be achieved by solving the integral equation (7), e.g., numerically by sampling state-action pairs $(\bar{s},\bar{a}) \sim d^{\pi,\gamma}(\cdot,\cdot|s,a)$ and features $z \sim h(\cdot|\bar{s},\bar{a})$, or learning a model of the conditional feature distribution. Section 4 provides a method for learning such a model off-policy.

3.2 Relationship with Alternative MaxEntRL Objectives

Let us consider an MDP without rewards and study exploration policies. We assume for simplicity that the feature distribution is the identity mapping, so that $z = (\bar{s}, \bar{a})$ and that $q^\pi(z|s,a) = d^{\pi,\gamma}(\bar{s}, \bar{a}|s,a) = d^{\pi,\gamma}(\bar{s}|s,a)\pi(\bar{a}|\bar{s})$, according to Definition 3.1. This MaxEntRL intrinsic reward function can be compared to one where the conditional distribution is $q^\pi(z|s,a) = d^{\pi,\gamma}(\bar{s},\bar{a}) = d^{\pi,\gamma}(\bar{s})\pi(\bar{a}|\bar{s})$, for all s and a, and where the MaxEntRL objective function simplifies to the entropy of that distribution. Theorem 3.2, shown in Appendix A, relates the two objectives.

Theorem 3.2. *For any policy π and any relative measure q^*, the marginal and conditional visitation measures satisfy*

$$\begin{aligned}\mathop{\mathbb{E}}_{s,a\sim d^{\pi,\gamma}(\cdot,\cdot)} &\left[-KL_{\bar{s},\bar{a}}\left[d^{\pi,\gamma}(\bar{s},\bar{a}|s,a)||q^*(\bar{s},\bar{a})\right]\right] \\ &\leq -KL_{\bar{s},\bar{a}}(d^{\pi,\gamma}(\bar{s},\bar{a})||q^*(\bar{s},\bar{a})) + L\sqrt{2\,KL_{\bar{s},\bar{a}}(d^{\pi,\gamma}(\bar{s},\bar{a})||\tilde{d}^{\pi,\gamma}(\bar{s},\bar{a}))}\,,\end{aligned}$$

where L is a finite constant and $\tilde{d}^{\pi,\gamma}(\bar{s},\bar{a}) = \mathbb{E}_{s,a\sim d^{\pi,\gamma}(\cdot,\cdot)}[d^{\pi,\gamma}(\bar{s},\bar{a}|s,a)]$.

Let us first assume that $\tilde{d}^{\pi,\gamma}(\bar{s},\bar{a}) = d^{\pi,\gamma}(\bar{s},\bar{a})$. Then, the left-hand side corresponds to our new objective with $R(s,a) = 0$, which is a lower bound on the entropy of the marginal state-action visitation measure. If $d^{\pi,\gamma}(\bar{s},\bar{a}|s,a) = q^*(\bar{s},\bar{a})$ almost everywhere, then the lower bound is zero, and $d^{\pi,\gamma}(\bar{s},\bar{a}) = q^*(\bar{s},\bar{a})$ almost everywhere as well. It implies that our MaxEntRL objective is a lower bound on the alternative MaxEntRL objective with marginal visitation. Additionally, a policy maximizing our objective also maximizes the alternative objective, when achieving an intrinsic reward of zero. In the limit when the discount factor tends to one $\gamma \to 1$, this equivalence also holds as the effect of the initial state vanishes and both distributions converge to the stationary visitation distribution, provided it exists. In practice, the bound holds only when $\tilde{d}^{\pi,\gamma}(\bar{s},\bar{a})$ is close to $d^{\pi,\gamma}(\bar{s},\bar{a})$; it intuitively corresponds to a stationary assumption on the initial states.

This result connects MaxEntRL optimizing the entropy of the conditional visitation with MaxEntRL optimizing the entropy of the same distribution marginalized over initial states and actions. It can be straightforwardly extended to relate the conditional and marginal visitation of features, typically such that MaxEntRL algorithms optimizing intrinsic rewards from Definition 3.1 with $z = \bar{s}$ maximize lower bounds of MaxEntRL algorithms using the state visitation (Hazan et al., 2019).

4 Off-policy Learning of Conditional Visitation Models

4.1 Fixed-Point Properties of Conditional Visitation

As explained in Section 3.1, the MaxEntRL intrinsic reward function in Definition 3.1 can be computed by sampling from the conditional feature distribution and evaluating the probability of these samples. In this section, we establish useful properties of this conditional distribution.

Let us first recall that the conditional state-action visitation distribution accepts a recursive definition (Janner et al., 2020) that is a trivial fixed point of the operator $\mathcal{T}^\pi$ from Definition 4.1.

Definition 4.1. The operator $\mathcal{T}^\pi$ is defined over the space of conditional state-action distributions as

$$\mathcal{T}^\pi q(\bar{s},\bar{a}|s,a) = (1-\gamma)\pi(\bar{a}|\bar{s})p_1(\bar{s}|s,a) + \gamma \mathop{\mathbb{E}}_{\substack{s'\sim p_1(\cdot|s,a)\\ a'\sim\pi(\cdot|s')}}\left[q(\bar{s},\bar{a}|s',a')\right]\,.$$

Theorem 4.2 establishes that the operator $\mathcal{T}^\pi$ is a contraction mapping, which implies the uniqueness of its fixed point. Assuming the result of the operator could be computed (or estimated), the fixed point could also be computed by successive application of this operator. It would allow computing the conditional state-action visitation distribution, and later estimating the intrinsic reward function.

Theorem 4.2. *The operator $\mathcal{T}^\pi$ is γ-contractive in $\bar{L}_n$-norm, where $\bar{L}_n(f)^n = \sup_y \int |f(x|y)|^n dx$.*

Definition 4.3 introduces the operator $\mathcal{P}^\pi$ that implicitly depends on the feature distribution h. If this distribution is the identity map, then both operators $\mathcal{P}^\pi$ and $\mathcal{T}^\pi$ are equal.

Definition 4.3. The operator $\mathcal{P}^\pi$ is defined over the space of conditional feature distributions as

$$\mathcal{P}^\pi q(z|s,a) = \mathop{\mathbb{E}}_{\substack{s'\sim p_1(\cdot|s,a)\\ a'\sim\pi(\cdot|s')}} \left[(1-\gamma)h(z|s',a') + \gamma q(z|s',a')\right] .$$

Next, we establish in Theorem 4.4 that this operator is also a contraction mapping, such that Theorem 4.2 can be considered a corollary of the present result. The fixed point could again theoretically be computed by successive application of this operator.

Theorem 4.4. *The operator $\mathcal{P}^\pi$ is γ-contractive in $\bar{L}_n$-norm, where $\bar{L}_n(f)^n = \sup_y \int |f(x|y)|^n dx$.*

Finally, we establish in Theorem 4.5 that the unique fixed point of $\mathcal{P}^\pi$ is the conditional distribution used in the MaxEntRL intrinsic reward from Definition 3.1. Assuming we could approximate this fixed point, we would get a model to compute the reward as in equation (8).

Theorem 4.5. *The unique fixed point of the operator $\mathcal{P}^\pi$ is*

$$q^\pi(z|s,a) = \int h(z|\bar{s},\bar{a}) d^{\pi,\gamma}(\bar{s},\bar{a}|s,a)\, d\bar{s}\, d\bar{a} \,.$$

The theorems are shown in Appendix B.

4.2 TD Learning of Conditional Visitation Models

In practice, computing the result of the operator $\mathcal{P}^\pi$ (and $(\mathcal{P}^\pi)^N$ after N applications) may be intractable when large state and action spaces are at hand or when these spaces are continuous. It furthermore requires having a model of the MDP. A common approach is then to rely on a function approximator q_ψ to approximate the fixed point. Furthermore, similarly to TD-learning methods (Sutton & Barto, 2018), Theorem 4.4 suggests optimizing the parameters of this model q_ψ to minimize the residual of the operator, measured with an $\bar{L}_n$-norm for which the operator is contractive. With this metric, estimating the residual would require estimating the probability of future features; it cannot be trivially minimized by stochastic gradient descent using transitions from the environment. Indeed, this residual involves the transition density, whose score would be required for gradient-based minimization but is not accessible in the model-free setting. We therefore propose to solve a surrogate minimum cross-entropy problem, in which stochastic gradient descent can be applied afterwards. For any policy π, the fixed point q^π is approximated with a function approximator q_ψ with parameter ψ optimized to solve

$$\arg\min_\psi \mathop{\mathbb{E}}_{\substack{s,a\sim\mu(\cdot,\cdot)\\ \bar{z}\sim(\mathcal{P}^\pi)^N q_\psi(\cdot|s,a)}} \left[-\log q_\psi(\bar{z}|s,a)\right] , \tag{9}$$

where μ is an arbitrary distribution over the state and action space, and where N is any positive integer. This optimization problem is related to minimizing the KL-divergence instead of an $\bar{L}_n$-norm (Bishop & Nasrabadi, 2006).

Let us make explicit how samples from the distribution $(\mathcal{P}^\pi)^N q_\psi(\bar{z}|s,a)$ can be generated from the MDP. By definition of the operator $\mathcal{P}^\pi$, the distribution $(\mathcal{P}^\pi)^N q_\psi(\bar{z}|s,a)$ is the mixture

$$\begin{aligned}(\mathcal{P}^\pi)^N q_\psi(\bar{z}|s,a) &= \left(\sum_{\Delta=1}^{N} (1-\gamma)\gamma^{\Delta-1} \mathop{\mathbb{E}}_{\substack{s'\sim p^\pi_\Delta(\cdot|s,a)\\ a'\sim\pi(\cdot|s')}} [h(\bar{z}|s',a')]\right) + \gamma^N \mathop{\mathbb{E}}_{\substack{s'\sim p^\pi_N(\cdot|s,a)\\ a'\sim\pi(\cdot|s')}} [q_\psi(\bar{z}|s',a')] \\ &= \sum_{\Delta=1}^{\infty} \mathcal{G}_{1-\gamma}(\Delta) \mathop{\mathbb{E}}_{\substack{s'\sim p^\pi_{\Delta'}(\cdot|s,a)\\ a'\sim\pi(\cdot|s')}} [b_\psi(\bar{z}|s',a',\Delta)] \Bigg|_{\Delta'=\min(\Delta,N)} ,\end{aligned} \tag{10}$$

where $\mathcal{G}_{1-\gamma}(\Delta)$ is the probability of Δ from a geometric distribution of parameter $1-\gamma$, and

$$b_\psi(\bar{z}|s,a,\Delta) = \begin{cases} h(\bar{z}|s,a) & \Delta \leq N \\ q_\psi(\bar{z}|s,a) & \Delta > N \end{cases} \tag{11}$$

Sampling from $(\mathcal{P}^\pi)^N q_\psi(\bar{z}|s,a)$ consists of sampling from the mixture.

Let us reformulate the problem in equation (9) highlighting the elements from the mixture and applying importance weighting

$$\arg\min_\psi \underset{\substack{s,a\sim\mu(\cdot,\cdot)\\ \Delta\sim\mathcal{G}_{1-\gamma'}(\cdot)\\ s'\sim p^\beta_{\Delta'}(\cdot|s,a)\\ a'\sim\pi(\cdot|s')\\ \bar{z}\sim b_\psi(\cdot|s',a',\Delta)}}{\mathbb{E}} \left[-\frac{\mathcal{G}_{1-\gamma}(\Delta)}{\mathcal{G}_{1-\gamma'}(\Delta)} \frac{p^\pi_{\Delta'}(s'|s,a)}{p^\beta_{\Delta'}(s'|s,a)} \log q_\psi(\bar{z}|s,a) \right] \Bigg|_{\Delta'=\min(\Delta,N)} . \tag{12}$$

Importance weighting is applied to the transition probability $p^\pi_{\Delta'}$ and to the geometric distribution $\mathcal{G}_{1-\gamma}$. It introduces the behavior policy β and a pseudo discount factor γ'. The first allows off-policy learning, and the second helps prevent some elements of the mixture from having negligible probabilities, improving results in practice. When $\beta = \pi$ or when $N = 1$, the second importance ratio simplifies to one; otherwise, it simplifies to a (finite) product of ratios of policies.

Learning q_ψ from samples can be achieved by solving the problem in equation (12) as an intermediate step to existing RL algorithms. First, the objective function is estimated using transitions. Second, this estimate is differentiated, and the parameter ψ is updated by gradient descent steps. In practice, the gradients generated by differentiating this loss function are biased. The influence of the parameter ψ on the probability of the sample $\bar{z}$ is neglected when bootstrapping, i.e., the partial derivative of $(\mathcal{P}^\pi)^N q_\psi(\bar{z}|s,a)$ with respect to ψ is neglected, and a target network is used. This is analogous to SARSA and TD-learning strategies (Sutton & Barto, 2018). Furthermore, we suggest neglecting the ratio of transition probabilities, which introduces a dependency of the distribution q_ψ on the policy β. Finally, the model q_ψ is used to compute the intrinsic rewards and update the policy.

5 Experiments

5.1 Experimental Setting

In this section, we detail the methodology applied to compare the MaxEntRL objectives discussed above. We use a suite of adapted environments for illustration, describe algorithms from the literature adapted to isolate the effect of the choice of objective from algorithmic considerations, and introduce measures to quantify the quality of exploration.

Environments. Experiments are performed on environments from the Minigrid suite (Chevalier-Boisvert et al., 2023). In the latter, an agent must travel across a grid containing walls and passages in order to reach a goal. The state space is a full observation of the maze, represented with an image, and the agent's orientation. The agent can take four different actions: turn left, turn right, move forward, or stand still. The need for exploration comes from the sparsity of the reward function, which is zero everywhere and equals one in the state to be reached.

Exploration strategies. We compare three exploration strategies, i.e., three intrinsic reward functions and their corresponding MaxEntRL algorithm. The first exploration strategy motivates agents to perform actions uniformly. The feature space is the action space $\mathcal{Z} = \mathcal{A}$, the conditional feature distribution is the policy $q^\pi(z|s,a) = \pi(z|s)$ for all a, and the relative measure q^* is uniform. The second exploration strategy motivates agents to visit uniformly the grid positions when starting from initial states. Here, the features $z \in \mathcal{Z}$ are the positions of the agent in the environment, the conditional distribution $q^\pi(z|s,a)$ is the discounted visitation measure of features, for all state s and action a, and the relative measure q^* is uniform. It corresponds to the alternative objective discussed in Section 3.2. The last exploration strategy is the one presented in Section 3.1. Again, the features $z \in \mathcal{Z}$ are the positions of the agent in the environment, and the relative measure q^* is uniform.

Algorithms. Existing algorithms can be adapted to optimize the previous MaxEntRL objective by adding the intrinsic reward to the reward from the MDP during policy optimization. In some algorithms, as in ours, it requires learning an additional model of some visitation measure. In this paper we adapted soft actor-critic to incorporate the different intrinsic reward functions. We consider three variants, one for each exploration strategy. First, without additional intrinsic reward, it is already a MaxEntRL algorithm that enforces the entropy of the policy. Second, we adapt the algorithm from Zhang et al. (2021a), using SAC (Haarnoja et al., 2018) instead of PPO (Schulman et al., 2017b) to improve sample efficiency and using a categorical distribution rather than a variational auto-encoder to approximate the visitation measure, which is made possible as the state-action space is discrete. It allows optimizing the approximator without relying on the evidence lower bound. Third, we adapt SAC to incorporate our reward function as discussed previously and detailed in Appendix C.

Exploration metrics. The last step is to quantify the quality of the exploration policies, which most often consists in measuring the diversity of states (or features). Here we report two such metrics during learning. First, the entropy of features visited by policies

$$-KL_z(d^{\pi,\gamma}(z)||q^*(z)) \, , \tag{13}$$

where $d^{\pi,\gamma}(z) = \mathbb{E}_{s,a\sim d^{\pi,\gamma}(\cdot,\cdot)}[h(z|s,a)]$ for all a and s. Second, the conditional entropy of features visited by policies given the initial state, in expectation,

$$-\mathbb{E}_{s_0\sim p_0(\cdot)}\left[KL_z(d^{\pi,\gamma}(z|s_0)||q^*(z))\right] \, , \tag{14}$$

where $d^{\pi,\gamma}(z|s_0) = \mathbb{E}_{s,a\sim d^{\pi,\gamma}(\cdot,\cdot|s_0)}[h(z|s,a)]$. The first metric measures the diversity in expectation over episodes, whereas the second measures the diversity within individual episodes. The first metric also corresponds to the exploration objectives maximized by the second algorithm.

5.2 Experimental Results

The methodology explained above is applied to various environments; further implementation details are given in Appendix D. Following the recommendations of Agarwal et al. (2021), all metrics are reported as interquartile mean (IQM) estimates with 95% confidence intervals. The evolution of all metrics is shown in Appendix E. We summarize the observations in this section.

Marginal exploration. Figure 2 in Appendix E illustrates the evolution of the entropy of features visited by policies $-KL_z(d^{\pi,\gamma}(z)||q^*(z))$ as a function of the learning iterations, when only optimizing the intrinsic rewards. We distinguish two situations. First, for some environments, the entropy does not evolve much during learning, and the three exploration strategies perform similarly. This is mostly due to the influence of the initial state distribution. In several environments, the initial position is drawn uniformly at random, such that the entropy of a random policy leads to high feature entropy due to symmetries. Finding optimal policies in such environments is arguably easy, as a large diversity of transitions will be observed without requiring complex MaxEntRL objectives. Second, for other environments, the entropy increases rapidly for the second algorithm, with marginal visitation measures (MV), and for the third algorithm, with conditional visitation measures (CV), and a high-entropy policy results from both optimizations. In these environments, MV achieves the highest entropy, followed closely by CV, while SAC performs poorly. It is worth noting that CV converges much faster than MV and eventually approaches the final entropy of MV despite optimizing a different objective, the remaining gap being small relative to their common gain over SAC. The first observation is likely a consequence of the off-policy estimation of the intrinsic reward function, which leads to improved sample efficiency, and the second may be justified by the bound from Theorem 3.2. Here, the environments are apparently more complex to explore, and both advanced strategies allow observing a wide diversity of features in expectation. Figure 1 reports these results for two representative environments.

Episode exploration. Figure 3 in Appendix E illustrates the evolution of the second metric $-\mathbb{E}_{s_0 \sim p_0(\cdot)}\left[KL_z(d^{\pi,\gamma}(z|s_0)||q^*(z))\right]$ as a function of the learning iterations, when only optimizing the intrinsic rewards. The metric improves as a function of the learning iterations in most cases. The best performance is most often achieved with the algorithm using CV, with SAC as the worst performer. The algorithm using MV nevertheless catches up with, and sometimes slightly exceeds, CV at the end of learning, notably in environments where the initial position is constant in each episode. Even in environments with uniform initial positions, motivating agents to explore leads to a larger variety of features visited during individual trajectories. Again, the intrinsic reward we present in this paper converges faster, which we attribute to the improved sample efficiency of the off-policy algorithm. The slightly lower final entropy of CV in some environments may result from the pseudo discount factor $\gamma' < \gamma$, which stabilizes learning but shifts the intrinsic reward toward features visited in the near future, as discussed in Appendix D. The evolution of the entropy is reported for the same representative environments as before in Figure 1.

Control policies. The literature sometimes focuses on exploration only, but MaxEntRL usually aims to explore in order to eventually compute a high-performance policy. Figure 4 in Appendix E reports the evolution of the expected return of policies when combining rewards from the environment with intrinsic rewards. In most environments, a sufficiently large weight to environment rewards combined with a large buffer leads to improving return during learning. In some environments, the complex exploration strategies allow learning faster, but no significant improvement was observed for the most favorable hyperparameters. The evolution of the expected return is reported for the same environments as before in Figure 1.

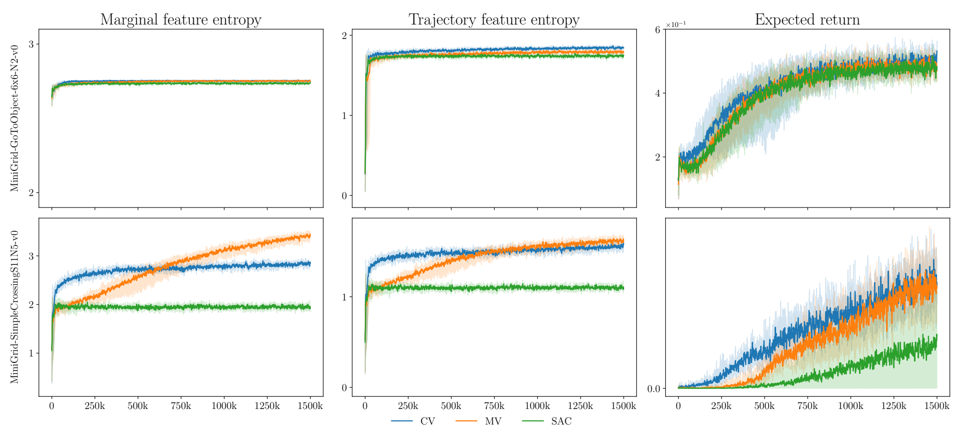

Figure 1: Learning results for two representative environments, selected from Appendix E. The first column represents the evolution of $-KL_z(d^{\pi,\gamma}(z)||q^*(z))$, and the second column the evolution of $-\mathbb{E}_{s_0 \sim p_0(\cdot)}\left[KL_z(d^{\pi,\gamma}(z|s_0)||q^*(z))\right]$, when learning exploration policies. The third column represents the evolution of the expected return when learning MaxEntRL control policies.

5.3 Discussion of Experiments

Experiments highlight that complex MaxEntRL methods are sometimes necessary to achieve better feature space coverage. In particular, our method often allows better exploration of features within individual trajectories compared to more standard objectives. A likely justification is that our objective, relying on the conditional entropy, motivates agents to explore in the future, i.e., for the remainder of the trajectory, where alternative objectives are more influenced by the initial states and actions. To the best of our knowledge, it is unclear if there is a best metric to compare exploration strategies, and experiments seem to highlight that exploration is highly environment-dependent.

Several phenomena influence the learning of the visitation models. First, when γ is close to one, the learning becomes unstable in practice. We hypothesize that it results from the increased importance of future states. Increasing the parameter N helps mitigate the issue as there is less bootstrapping, reducing the risk of learning a biased target. Second, we neglect some importance weights in practice to reduce variance, which makes q_ψ partially dependent on the behavior policy β. Bootstrapping still propagates long-term effects of the policy. In practice, regularizing the policy entropy even when using the two complex exploration objectives appeared to be critical to stabilize learning.

Finally, we relied on off-policy actor-critic for concreteness, yet the MaxEntRL objective is agnostic to the control backbone, and similar results should hold with other RL methods. We decided to share the same backbone to center the discussion around the MaxEntRL objective but acknowledge that the choice of algorithm may influence practical performance. Also, we observe that our method offers a practical alternative to directly maximizing marginal visitation, but we did not focus on potential theoretical advantages of different exploration objectives.

6 Conclusion

In this paper, we presented a new MaxEntRL objective that provides intrinsic reward bonuses proportional to the entropy of the distribution of features built from the states and actions visited by the agent in future time steps. We show that the expected sum of these intrinsic rewards is a lower bound on an alternative maximum entropy objective that uses the marginal distribution of features starting from the initial state. The reward bonus can furthermore be estimated efficiently by sampling from the conditional distribution of visited features, which we proved to be the fixed point of a contraction mapping and which can be learned for any policy using batches of arbitrary transitions. This estimation procedure is easy to implement and can be integrated into existing RL algorithms by augmenting the reward with our intrinsic bonus. It was integrated into an end-to-end off-policy algorithm and benchmarked on several control problems. Experiments show improved feature visitation within individual trajectories in most environments and faster convergence for exploration-only agents, while control performance remains similar on the considered benchmarks.

This study reminds us that the exploration objective to pursue remains unclear and, in practice, depends on the environment. Future work includes clarifying this question theoretically and empirically. This includes studying the algorithm we presented and benchmarking it in more challenging environments. A key limitation of our algorithm is that the feature space is fixed a priori, and performance depends on this choice. A potential avenue is to learn it and explore maximum-information state-action feature spaces or reward-predictive feature spaces. Another minor practical limitation is that we estimate the conditional probabilities of future features with a parametric density model trained by minimizing cross-entropy, which restricts the model class, though it still covers mixtures of parametric distributions and normalizing flows. Using other models, such as latent space models or diffusion models, would require adapting the estimation and learning procedure.

Acknowledgment

Adrien Bolland gratefully acknowledges the financial support of the *Fonds de la Recherche Scientifique – FNRS*. Gaspard Lambrechts is a Marie Skłodowska-Curie Actions (MSCA) postdoctoral fellow at McGill University, funded by Horizon Europe under grant 101284668. Part of this work was carried out while he was a postdoctoral fellow of the FNRS. He also gratefully acknowledges Wallonia-Brussels International for its support through a World Excellence Fellowship.

References

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron C Courville, and Marc G Bellemare. Deep reinforcement learning at the edge of the statistical precipice. *Advances in Neural Information Processing Systems*, 34:29304–29320, 2021.

Zafarali Ahmed, Nicolas Le Roux, Mohammad Norouzi, and Dale Schuurmans. Understanding the impact of entropy on policy optimization. In *International Conference on Machine Learning*, pp. 151–160. PMLR, 2019.

Samy Aittahar, Adrien Bolland, Guillaume Derval, and Damien Ernst. Optimal control of renewable energy communities subject to network peak fees with model predictive control and reinforcement learning algorithms. *arXiv preprint arXiv:2401.16321*, 2024.

André Barreto, Will Dabney, Rémi Munos, Jonathan J Hunt, Tom Schaul, Hado P van Hasselt, and David Silver. Successor features for transfer in reinforcement learning. *Advances in neural information processing systems*, 30, 2017.

Marc Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Remi Munos. Unifying count-based exploration and intrinsic motivation. *Advances in Neural Information Processing Systems*, 29, 2016.

Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.

Adrien Bolland, Gilles Louppe, and Damien Ernst. Policy gradient algorithms implicitly optimize by continuation. *Transactions on Machine Learning Research*, 2023.

Adrien Bolland, Gaspard Lambrechts, and Damien Ernst. Behind the myth of exploration in policy gradients. *arXiv preprint arXiv:2402.00162*, 2024.

Ioannis Boukas, Damien Ernst, Thibaut Théate, Adrien Bolland, Alexandre Huynen, Martin Buchwald, Christelle Wynants, and Bertrand Cornélusse. A deep reinforcement learning framework for continuous intraday market bidding. *Machine Learning*, 110:2335–2387, 2021.

Rob Brekelmans, Tim Genewein, Jordi Grau-Moya, Grégoire Delétang, Markus Kunesch, Shane Legg, and Pedro Ortega. Your policy regularizer is secretly an adversary. *arXiv preprint arXiv:2203.12592*, 2022.

Yuri Burda, Harri Edwards, Deepak Pathak, Amos Storkey, Trevor Darrell, and Alexei A Efros. Large-scale study of curiosity-driven learning. *arXiv preprint arXiv:1808.04355*, 2018.

Maxime Chevalier-Boisvert, Bolun Dai, Mark Towers, Rodrigo de Lazcano, Lucas Willems, Salem Lahlou, Suman Pal, Pablo Samuel Castro, and Jordan Terry. Minigrid & miniworld: Modular & customizable reinforcement learning environments for goal-oriented tasks. *CoRR*, abs/2306.13831, 2023.

Thomas Degris, Martha White, and Richard S Sutton. Off-policy actor-critic. *arXiv preprint arXiv:1205.4839*, 2012.

Benjamin Eysenbach, Ruslan Salakhutdinov, and Sergey Levine. C-learning: Learning to achieve goals via recursive classification. *arXiv preprint arXiv:2011.08909*, 2020.

Benjamin Eysenbach, Tianjun Zhang, Sergey Levine, and Russ R Salakhutdinov. Contrastive learning as goal-conditioned reinforcement learning. *Advances in Neural Information Processing Systems*, 35:35603–35620, 2022.

Benjamin Eysenbach, Vivek Myers, Sergey Levine, and Ruslan Salakhutdinov. Contrastive representations make planning easy. In *NeurIPS 2023 Workshop on Generalization in Planning*, 2023.

Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Alaa Saade, Shantanu Thakoor, Bilal Piot, Bernardo Avila Pires, Michal Valko, Thomas Mesnard, Tor Lattimore, and Rémi Munos. Geometric entropic exploration. *arXiv preprint arXiv:2101.02055*, 2021.

Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. In *International Conference on Machine Learning*, pp. 1352–1361. PMLR, 2017.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, pp. 1861–1870. PMLR, 2018.

Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, Pieter Abbeel, et al. Soft actor-critic algorithms and applications. *arXiv preprint arXiv:1812.05905*, 2019.

Elad Hazan, Sham Kakade, Karan Singh, and Abby Van Soest. Provably efficient maximum entropy exploration. In *International Conference on Machine Learning*, pp. 2681–2691. PMLR, 2019.

Shengyi Huang, Rousslan Fernand Julien Dossa, Chang Ye, Jeff Braga, Dipam Chakraborty, Kinal Mehta, and João G.M. Araújo. Cleanrl: High-quality single-file implementations of deep reinforcement learning algorithms. *Journal of Machine Learning Research*, 23(274):1–18, 2022.

Hisham Husain, Kamil Ciosek, and Ryota Tomioka. Regularized policies are reward robust. In *International Conference on Artificial Intelligence and Statistics*, pp. 64–72. PMLR, 2021.

Riashat Islam, Zafarali Ahmed, and Doina Precup. Marginalized state distribution entropy regularization in policy optimization. *arXiv preprint arXiv:1912.05128*, 2019.

Arnav Kumar Jain, Lucas Lehnert, Irina Rish, and Glen Berseth. Maximum state entropy exploration using predecessor and successor representations. *Advances in Neural Information Processing Systems*, 36, 2024.

Michael Janner, Igor Mordatch, and Sergey Levine. Generative temporal difference learning for infinite-horizon prediction. *arXiv preprint arXiv:2010.14496*, 2020.

Diederik P Kingma and Jimmy L Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Lisa Lee, Benjamin Eysenbach, Emilio Parisotto, Eric Xing, Sergey Levine, and Ruslan Salakhutdinov. Efficient exploration via state marginal matching. *arXiv preprint arXiv:1906.05274*, 2019.

Hao Liu and Pieter Abbeel. Behavior from the void: Unsupervised active pre-training. *Advances in Neural Information Processing Systems*, 34:18459–18473, 2021.

Yecheng Jason Ma, Kausik Sivakumar, Jason Yan, Osbert Bastani, and Dinesh Jayaraman. Learning policy-aware models for model-based reinforcement learning via transition occupancy matching. In *Learning for Dynamics and Control Conference*, pp. 259–271. PMLR, 2023.

Alan S Manne. Linear programming and sequential decisions. *Management Science*, 6(3):259–267, 1960.

Bogdan Mazoure, Jake Bruce, Doina Precup, Rob Fergus, and Ankit Anand. Accelerating exploration and representation learning with offline pre-training. *arXiv preprint arXiv:2304.00046*, 2023a.

Bogdan Mazoure, Benjamin Eysenbach, Ofir Nachum, and Jonathan Tompson. Contrastive value learning: Implicit models for simple offline rl. In *Conference on Robot Learning*, pp. 1257–1267. PMLR, 2023b.

Bogdan Mazoure, Walter Talbott, Miguel Angel Bautista, Devon Hjelm, Alexander Toshev, and Josh Susskind. Value function estimation using conditional diffusion models for control. *arXiv preprint arXiv:2306.07290*, 2023c.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

Faisal Mohamed, Catherine Ji, Benjamin Eysenbach, and Glen Berseth. Curiosity-driven exploration via temporal contrastive learning. In *Workshop on Reinforcement Learning Beyond Rewards@ Reinforcement Learning Conference 2025*, 2025.

Mirco Mutti and Marcello Restelli. An intrinsically-motivated approach for learning highly exploring and fast mixing policies. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 5232–5239, 2020.

Mirco Mutti, Lorenzo Pratissoli, and Marcello Restelli. Task-agnostic exploration via policy gradient of a non-parametric state entropy estimate. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 9028–9036, 2021.

Mirco Mutti, Riccardo De Santi, and Marcello Restelli. The importance of non-markovianity in maximum state entropy exploration. *arXiv preprint arXiv:2202.03060*, 2022.

Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *International conference on machine learning*, pp. 2778–2787. PMLR, 2017.

John Schulman, Xi Chen, and Pieter Abbeel. Equivalence between policy gradients and soft q-learning. *arXiv preprint arXiv:1704.06440*, 2017a.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017b.

Younggyo Seo, Lili Chen, Jinwoo Shin, Honglak Lee, Pieter Abbeel, and Kimin Lee. State entropy maximization with random encoders for efficient exploration. In *International Conference on Machine Learning*, pp. 9443–9454. PMLR, 2021.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of Go without human knowledge. *Nature*, 550(7676):354–359, 2017.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

Daniil Tiapkin, Denis Belomestny, Daniele Calandriello, Eric Moulines, Remi Munos, Alexey Naumov, Pierre Perrault, Yunhao Tang, Michal Valko, and Pierre Menard. Fast rates for maximum entropy exploration. In *International Conference on Machine Learning*, pp. 34161–34221. PMLR, 2023.

Marc Toussaint. Robot trajectory optimization using approximate inference. In *International Conference on Machine Learning*, volume 26, pp. 1049–1056, 2009.

Ronald J Williams and Jing Peng. Function optimization using connectionist reinforcement learning algorithms. *Connection Science*, 3(3):241–268, 1991.

Denis Yarats, Rob Fergus, Alessandro Lazaric, and Lerrel Pinto. Reinforcement learning with prototypical representations. In *International Conference on Machine Learning*, pp. 11920–11931. PMLR, 2021.

Chuheng Zhang, Yuanying Cai, Longbo Huang, and Jian Li. Exploration by maximizing rényi entropy for reward-free rl framework. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 10859–10867, 2021a.

Tianjun Zhang, Huazhe Xu, Xiaolong Wang, Yi Wu, Kurt Keutzer, Joseph E Gonzalez, and Yuandong Tian. Noveld: A simple yet effective exploration criterion. *Advances in Neural Information Processing Systems*, 34:25217–25230, 2021b.

Brian D Ziebart. *Modeling purposeful adaptive behavior with the principle of maximum causal entropy*. PhD thesis, Carnegie Mellon University, 2010.

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

Supplementary Materials

The following content was not necessarily subject to peer review.

A Proof Lower Bound

Theorem (3.2, restated)**.** *For any policy π and any relative measure q^*, the marginal and conditional visitation measures satisfy*

$$\mathop{\mathbb{E}}_{s,a\sim d^{\pi,\gamma}(\cdot,\cdot)}\left[-KL_{\bar{s},\bar{a}}\left[d^{\pi,\gamma}(\bar{s},\bar{a}|s,a)||q^*(\bar{s},\bar{a})\right]\right]$$
$$\leq -KL_{\bar{s},\bar{a}}(d^{\pi,\gamma}(\bar{s},\bar{a})||q^*(\bar{s},\bar{a})) + L\sqrt{2\,KL_{\bar{s},\bar{a}}(d^{\pi,\gamma}(\bar{s},\bar{a})||\tilde{d}^{\pi,\gamma}(\bar{s},\bar{a}))}\,,$$

where L is a finite constant and $\tilde{d}^{\pi,\gamma}(\bar{s},\bar{a}) = \mathbb{E}_{s,a\sim d^{\pi,\gamma}(\cdot,\cdot)}[d^{\pi,\gamma}(\bar{s},\bar{a}|s,a)]$.

Proof Theorem 3.2 Let us assume that all distributions admit continuous, smooth, and strictly positive densities on their support with bounded log-ratios. Let $\tilde{d}^{\pi,\gamma}(\bar{s},\bar{a}) = \mathbb{E}_{s,a\sim d^{\pi,\gamma}(\cdot,\cdot)}[d^{\pi,\gamma}(\bar{s},\bar{a}|s,a)]$. Let us develop using the convexity of KL divergence, positivity of KL divergence, integral properties and Pinsker's inequality

$$\begin{aligned}
&\mathop{\mathbb{E}}_{s,a\sim d^{\pi,\gamma}(\cdot,\cdot)}\left[KL_{\bar{s},\bar{a}}\left[d^{\pi,\gamma}(\bar{s},\bar{a}|s,a)||q^*(\bar{s},\bar{a})\right]\right]\\
&\quad= \mathop{\mathbb{E}}_{\substack{s,a\sim d^{\pi,\gamma}(\cdot,\cdot)\\ \bar{s},\bar{a}\sim d^{\pi,\gamma}(\cdot,\cdot|s,a)}}\left[\log\frac{d^{\pi,\gamma}(\bar{s},\bar{a}|s,a)}{q^*(\bar{s},\bar{a})}\right]\\
&\quad= \mathop{\mathbb{E}}_{\substack{s,a\sim d^{\pi,\gamma}(\cdot,\cdot)\\ \bar{s},\bar{a}\sim d^{\pi,\gamma}(\cdot,\cdot|s,a)}}\left[\log\frac{d^{\pi,\gamma}(\bar{s},\bar{a}|s,a)}{d^{\pi,\gamma}(\bar{s},\bar{a})} + \log\frac{d^{\pi,\gamma}(\bar{s},\bar{a})}{q^*(\bar{s},\bar{a})}\right]\\
&\quad\geq \mathop{\mathbb{E}}_{\substack{s,a\sim d^{\pi,\gamma}(\cdot,\cdot)\\ \bar{s},\bar{a}\sim d^{\pi,\gamma}(\cdot,\cdot|s,a)}}\left[\log\frac{d^{\pi,\gamma}(\bar{s},\bar{a})}{q^*(\bar{s},\bar{a})}\right]\\
&\quad= \int \tilde{d}^{\pi,\gamma}(\bar{s},\bar{a})\log\frac{d^{\pi,\gamma}(\bar{s},\bar{a})}{q^*(\bar{s},\bar{a})}\,d\bar{s}\,d\bar{a}\\
&\quad= \int d^{\pi,\gamma}\log\frac{d^{\pi,\gamma}}{q^*}\,d\bar{s}\,d\bar{a} - \int (d^{\pi,\gamma}-\tilde{d}^{\pi,\gamma})\log\frac{d^{\pi,\gamma}}{q^*}\,d\bar{s}\,d\bar{a}\\
&\quad= KL_{\bar{s},\bar{a}}(d^{\pi,\gamma}(\bar{s},\bar{a})||q^*(\bar{s},\bar{a})) - \int (d^{\pi,\gamma}-\tilde{d}^{\pi,\gamma})\log\frac{d^{\pi,\gamma}}{q^*}\,d\bar{s}\,d\bar{a}\\
&\quad\geq KL_{\bar{s},\bar{a}}(d^{\pi,\gamma}(\bar{s},\bar{a})||q^*(\bar{s},\bar{a})) - \left|\int (d^{\pi,\gamma}-\tilde{d}^{\pi,\gamma})\log\frac{d^{\pi,\gamma}}{q^*}\,d\bar{s}\,d\bar{a}\right|\\
&\quad\geq KL_{\bar{s},\bar{a}}(d^{\pi,\gamma}(\bar{s},\bar{a})||q^*(\bar{s},\bar{a})) - L\int\left|d^{\pi,\gamma}-\tilde{d}^{\pi,\gamma}\right|\,d\bar{s}\,d\bar{a}\\
&\quad\geq KL_{\bar{s},\bar{a}}(d^{\pi,\gamma}(\bar{s},\bar{a})||q^*(\bar{s},\bar{a})) - 2\,L\,TV(d^{\pi,\gamma},\tilde{d}^{\pi,\gamma})\\
&\quad\geq KL_{\bar{s},\bar{a}}(d^{\pi,\gamma}(\bar{s},\bar{a})||q^*(\bar{s},\bar{a})) - L\sqrt{2\,KL_{\bar{s},\bar{a}}(d^{\pi,\gamma}(\bar{s},\bar{a})||\tilde{d}^{\pi,\gamma}(\bar{s},\bar{a}))}
\end{aligned}$$

$\square$

B Proofs Theorems on Contractions

Theorem (4.2, restated)**.** *The operator $\mathcal{T}^\pi$ is γ-contractive in $\bar{L}_n$-norm, where $\bar{L}_n(f)^n = \sup_y \int |f(x|y)|^n \, dx$.*

Proof Theorem 4.2. This theorem is a particular case of Theorem 4.4, where the feature space $\mathcal{Z} = \mathcal{S} \times \mathcal{A}$ and the mapping h is a Dirac

$$h(z|s,a) = \delta_{(s,a)}(z)\,.$$

□

Theorem (4.4, restated)**.** *The operator $\mathcal{P}^\pi$ is γ-contractive in $\bar{L}_n$-norm, where $\bar{L}_n(f)^n = \sup_y \int |f(x|y)|^n \, dx$.*

Proof Theorem 4.4. For all conditional distributions f and g

$$\begin{aligned}
\sup_{s,a} L_n(\mathcal{P}^\pi f(\cdot|s,a), \mathcal{P}^\pi g(\cdot|s,a))^n &= \sup_{s,a} \int |\mathcal{P}^\pi f(\bar{z}|s,a) - \mathcal{P}^\pi g(\bar{z}|s,a)|^n \, d\bar{z} \\
&= \gamma^n \sup_{s,a} \int \left| \mathop{\mathbb{E}}_{\substack{s' \sim p_1(\cdot|s,a) \\ a' \sim \pi(\cdot|s')}} [f(\bar{z}|s',a') - g(\bar{z}|s',a')] \right|^n \, d\bar{z} \\
&\leq \gamma^n \sup_{s,a} \int \mathop{\mathbb{E}}_{\substack{s' \sim p_1(\cdot|s,a) \\ a' \sim \pi(\cdot|s')}} [|f(\bar{z}|s',a') - g(\bar{z}|s',a')|^n] \, d\bar{z} \\
&= \gamma^n \sup_{s,a} \mathop{\mathbb{E}}_{\substack{s' \sim p_1(\cdot|s,a) \\ a' \sim \pi(\cdot|s')}} \left[\int |f(\bar{z}|s',a') - g(\bar{z}|s',a')|^n \, d\bar{z} \right] \\
&\leq \gamma^n \sup_{s,a} \sup_{s',a'} \left(\int |f(\bar{z}|s',a') - g(\bar{z}|s',a')|^n \, d\bar{z} \right) \\
&= \gamma^n \sup_{s',a'} \int |f(\bar{z}|s',a') - g(\bar{z}|s',a')|^n \, d\bar{z} \\
&= \gamma^n \sup_{s,a} L_n(f(\cdot|s,a), g(\cdot|s,a))^n
\end{aligned}$$

□

Theorem (4.5, restated)**.** *The unique fixed point of the operator $\mathcal{P}^\pi$ is*

$$q^\pi(z|s,a) = \int h(z|\bar{s},\bar{a}) d^{\pi,\gamma}(\bar{s},\bar{a}|s,a)\, d\bar{s}\, d\bar{a}\,.$$

Proof Theorem 4.5. The operator $\mathcal{P}^\pi$ is contractive, which implies that there exists a unique fixed point. Let us apply the operator $\mathcal{P}^\pi$ to the distribution q^π

$$\begin{aligned}
\mathcal{P}^\pi q^\pi(z|s,a) &= \mathcal{P}^\pi \int h(z|\bar{s},\bar{a}) d^{\pi,\gamma}(\bar{s},\bar{a}|s,a)\, d\bar{s}\, d\bar{a} \\
&= \mathop{\mathbb{E}}_{\substack{s'\sim p_1(\cdot|s,a) \\ a'\sim\pi(\cdot|s')}} \left[(1-\gamma) h(z|s',a') + \gamma \left(\int h(z|\bar{s},\bar{a}) d^{\pi,\gamma}(\bar{s},\bar{a}|s',a')\, d\bar{s}\, d\bar{a} \right) \right] \\
&= \int h(z|\bar{s},\bar{a}) \left(\mathop{\mathbb{E}}_{\substack{s'\sim p_1(\cdot|s,a) \\ a'\sim\pi(\cdot|s')}} \left[(1-\gamma)\delta_{\bar{s},\bar{a}}(s',a') + \gamma d^{\pi,\gamma}(\bar{s},\bar{a}|s',a') \right] \right) d\bar{s}\, d\bar{a} \\
&= \int h(z|\bar{s},\bar{a}) \left((1-\gamma)\pi(\bar{a}|\bar{s}) p_1(\bar{s}|s,a) + \gamma \mathop{\mathbb{E}}_{\substack{s'\sim p_1(\cdot|s,a) \\ a'\sim\pi(\cdot|s')}} \left[d^{\pi,\gamma}(\bar{s},\bar{a}|s',a') \right] \right) d\bar{s}\, d\bar{a} \\
&= \int h(z|\bar{s},\bar{a}) d^{\pi,\gamma}(\bar{s},\bar{a}|s,a)\, d\bar{s}\, d\bar{a} \\
&= q^\pi(z|s,a)\,.
\end{aligned}$$

□

C Soft Actor-Critic with Conditional Visitation Measure

In the following, we adapt soft actor-critic (Haarnoja et al., 2018), itself an adaptation of off-policy actor-critic (Degris et al., 2012), to include additional intrinsic rewards. In essence, soft actor-critic estimates the state-action value function with a parameterized critic Q_ϕ, which is learned using expected SARSA (sometimes called generalized SARSA), and updates the parameterized policy π_θ with approximate policy iteration (i.e., off-policy policy gradient), all based on one-step transitions stored in a replay buffer $\mathcal{D}$. The actor and critic loss functions are furthermore extended with the log-likelihood of actions weighted by the parameter λ_{SAC}; it is therefore called soft and considered a MaxEntRL algorithm regularizing the entropy of policies. In the particular case in which λ_{SAC} equals zero, the algorithm boils down to a slightly revisited implementation of off-policy actor-critic.

Soft actor-critic is adapted to MaxEntRL with the intrinsic reward function defined in Section 3.1, as follows. First, N-step transitions are stored in the buffer $\mathcal{D}$ instead of one-step transitions. Second, the conditional feature distribution is estimated with a function approximator q_ψ and learned with stochastic gradient descent. Third, at each iteration of the critic updates, the reward provided by the MDP is extended with the intrinsic reward. Algorithm 1 summarizes the learning steps.

Formally, the parameterized critic Q_ϕ is iteratively updated by performing stochastic gradient descent steps on the loss function

$$\mathcal{L}(\phi) = \mathop{\mathbb{E}}_{s_t, a_t \sim \mathcal{D}} \left[\left(Q_\phi(s_t, a_t) - y \right)^2 \right] \tag{15}$$

$$y = \lambda_R R(s_t, a_t) + \lambda R^{int}(s_t, a_t) + \gamma \left(Q_{\phi'}(s_{t+1}, a_{t+1'}) - \lambda_{SAC} \log \pi_\theta(a_{t+1'}|s_{t+1}) \right) , \tag{16}$$

where $a_{t+1'} \sim \pi_\theta(\cdot|s_{t+1})$, and where ϕ' is the target network parameter.

Furthermore, the policy π_θ is updated by performing gradient descent steps on the loss function

$$\mathcal{L}(\theta) = - \mathop{\mathbb{E}}_{s_t, a_t \sim \mathcal{D}} \left[\log \pi_\theta(a_{t'}|s_t) A(s_t, a_{t'}) \right] \tag{17}$$

$$A(s_t, a_{t'}) = Q_\phi(s_t, a_{t'}) - \lambda_{SAC} \log \pi_\theta(a_{t'}|s_t) , \tag{18}$$

where $a_{t'} \sim \pi_\theta(\cdot|s_t)$.

Algorithm 1 SAC with conditional visitation measure for exploration

Initialize the policy π_θ, the soft critic Q_ϕ, and the conditional feature model q_ψ
Initialize the critic target $Q_{\phi'}$ and visitation target $q_{\psi'}$
Initialize the replay buffer with N-step transitions
while Learning **do**
  Sample transitions from the policy π_θ and add them to the buffer
  while Update the visitation model **do**
    Sample a batch of N-step transitions from the buffer
    Update the model q_ψ
  end while
  while Update the critic **do**
    Sample a batch of N-step transitions from the buffer (use only the 1-step transitions)
    For each element of the batch sample $z_t \sim q_\psi(\cdot|s_t, a_t)$
    Estimate the intrinsic reward $R^{int}(s_t, a_t) = \log q^*(z_t) - \log q_\psi(z_t|s_t, a_t)$
    Perform a stochastic gradient descent step on $\mathcal{L}(\phi)$
  end while
  Sample a batch of N-step transitions from the buffer (use only the 1-step transitions)
  Perform a stochastic gradient descent step on $\mathcal{L}(\theta)$
  Update the target parameters with Polyak averaging
end while

D Implementation Details

This section aims to clarify the practical implementation and how it differs in practice from the simplified algorithms presented.

The agent observes an image that is processed by a convolutional neural network into a state feature. The policy π_θ is a forward network that processes this feature and that outputs a categorical distribution over the action representation. The critic Q_ϕ is a neural network that averages the state feature with a projection of the action representation, computed by a forward network, and processes the result with a final forward network that outputs a scalar. In CV, the distribution model q_ψ is also a neural network that takes the same input as the critic Q_ϕ and outputs a categorical distribution over a one-hot-encoding representation of positions. In MV, the visitation distribution model q_ψ is a marginal distribution over the same one-hot-encoding representation. This marginal model is learned from transitions of the replay buffer, weighted by $(1-\gamma)\gamma^t$ where t is the time step of the transition within its episode, without importance correction for the behavior policy; the model thus estimates the discounted feature visitation of the buffer distribution rather than that of the current policy, which we found more stable in practice.

The implementation of soft actor-critic differs slightly from that of the original work (Haarnoja et al., 2018). The actor loss equation is based on the log-trick instead of the reparameterization trick, the expected SARSA update of the critic is approximated by sampling, and a single value function is learned, as implemented in CleanRL (Huang et al., 2022).

In CV, the feature z_t used to estimate the intrinsic reward in equation (8) is furthermore not sampled from the model q_ψ directly. It is instead sampled from the bootstrapped N-step mixture serving as target for learning the density in Section 4.2: a horizon is drawn from the geometric distribution with pseudo discount factor γ', the observed feature from the stored N-step transition is used when the horizon is at most N, and a sample from the bootstrapped model is used otherwise. Target networks are used for the density model and for the policy providing the future action, both when bootstrapping and when evaluating the log-probability of the sample, and no importance weights are applied. Relying on observed features rather than model samples reduces the bias introduced by an imperfectly learned density model. In addition, sampling horizons with $\gamma' < \gamma$ shifts the reward toward features visited in the near future; in practice, reducing γ' leads to a slightly lower within-trajectory entropy but to a more greedy exploration overall, as features are diversified earlier in the trajectories.

Some last practical choices were made and should be noted. The replay buffer is first filled with transitions collected from the current policy that are replaced afterwards following a FIFO policy during learning iterations. The number of transitions sampled per learning iteration is a hyperparameter. The actor update uses replay time step weights of the form $\gamma^{t-t_{\min}}$ and relies on an optional centering of the soft Q-values, which we refer to as actor centering. During the critic updates, the intrinsic reward is bounded on both sides: a small probability $\epsilon = 10^{-4}$ is added to the model density before taking its logarithm, which caps the maximum intrinsic reward when densities vanish, and the reward is clipped from below at $\log 10^{-2}$ to avoid negative reward overshoots when densities are too concentrated.

The parameters of all previous models are updated independently using the Adam optimization rule (Kingma & Ba, 2014).

Hyperparameters were selected to maximize the area under the learning curve of the marginal feature entropy in the exploration experiments. A grid search was performed over the learning rates, the Polyak averaging weights were reduced when instabilities were observed, and the remaining parameters were fixed to standard values. Configurations leading to large instabilities were rejected.

Table 1 summarizes the hyperparameters used for the experiments.

Table 1: Hyperparameters

Parameter	Value
Convolutional layer	2
Neurons for each convolutional layer	32
State-feature size	256
Forward layers policy	2
Forward layers critic	2
Neurons forward layers	256
Action-projection size critic	256
Learning rate policy	10^{-6}
Learning rate critic	10^{-6}
Maximum trajectory length	200
Buffer size	100000
Initial buffer fill (random policy)	10000
Batch size	128
Critic target update weight τ	0.005
Discount factor γ	0.98
Extrinsic reward weight (control experiments)	5
SAC λ_{SAC}	0.01
Forward layers visitation model	2
Neurons forward layers	256
Action-projection size visitation model	256
Learning rate visitation model	10^{-6}
MaxEntRL λ	0.005
Density model target update weight τ	0.005
Bootstrapping steps N (CV)	8
Pseudo discount factor γ' (CV)	0.95
Policy target update weight τ (CV)	0.005

E Experimental Results

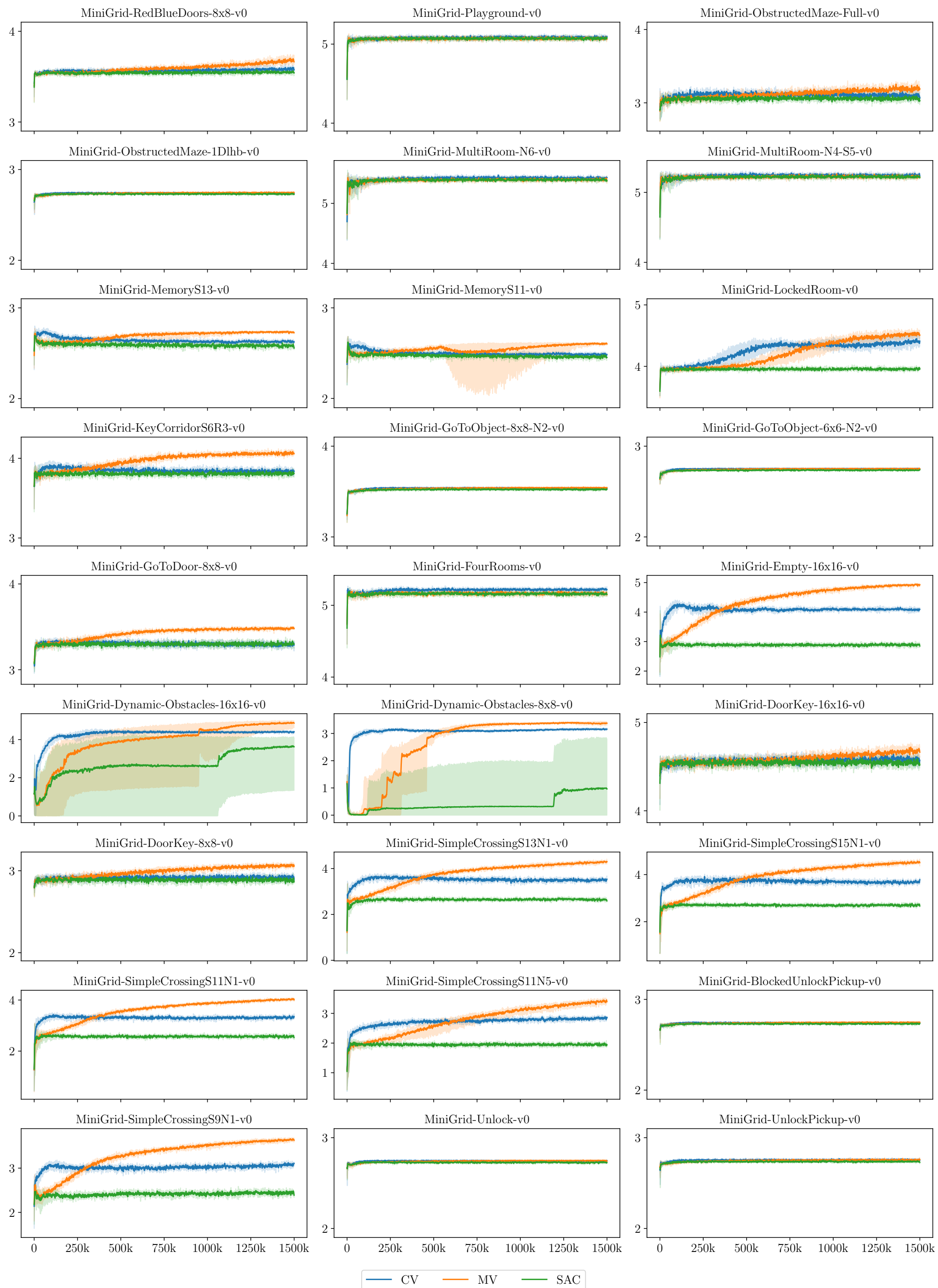

Figure 2: Evolution of the entropy of the discounted visitation probability measure of the position of the agent on the grid when computing exploration policies (i.e., when neglecting the rewards of the MDP). The entropy is computed empirically with Monte Carlo simulations. For each iteration, the interquartile mean over 5 runs is reported, along with its 95% confidence interval.

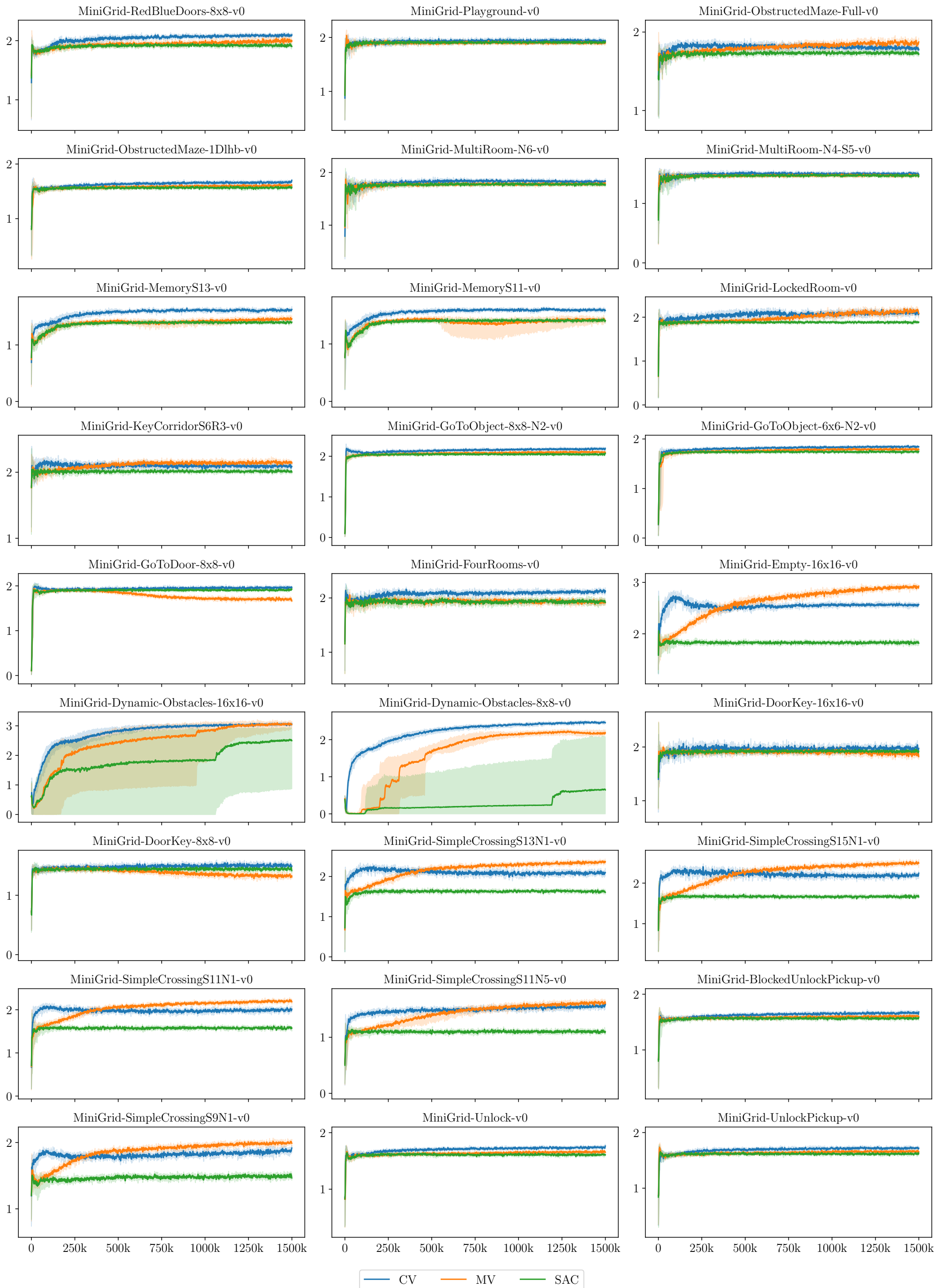

Figure 3: Evolution of the conditional entropy of the discounted visitation probability measure of the position of the agent on the grid when computing exploration policies (i.e., when neglecting the rewards of the MDP). The entropy is computed empirically with Monte Carlo simulations. For each iteration, the interquartile mean over 5 runs is reported, along with its 95% confidence interval.

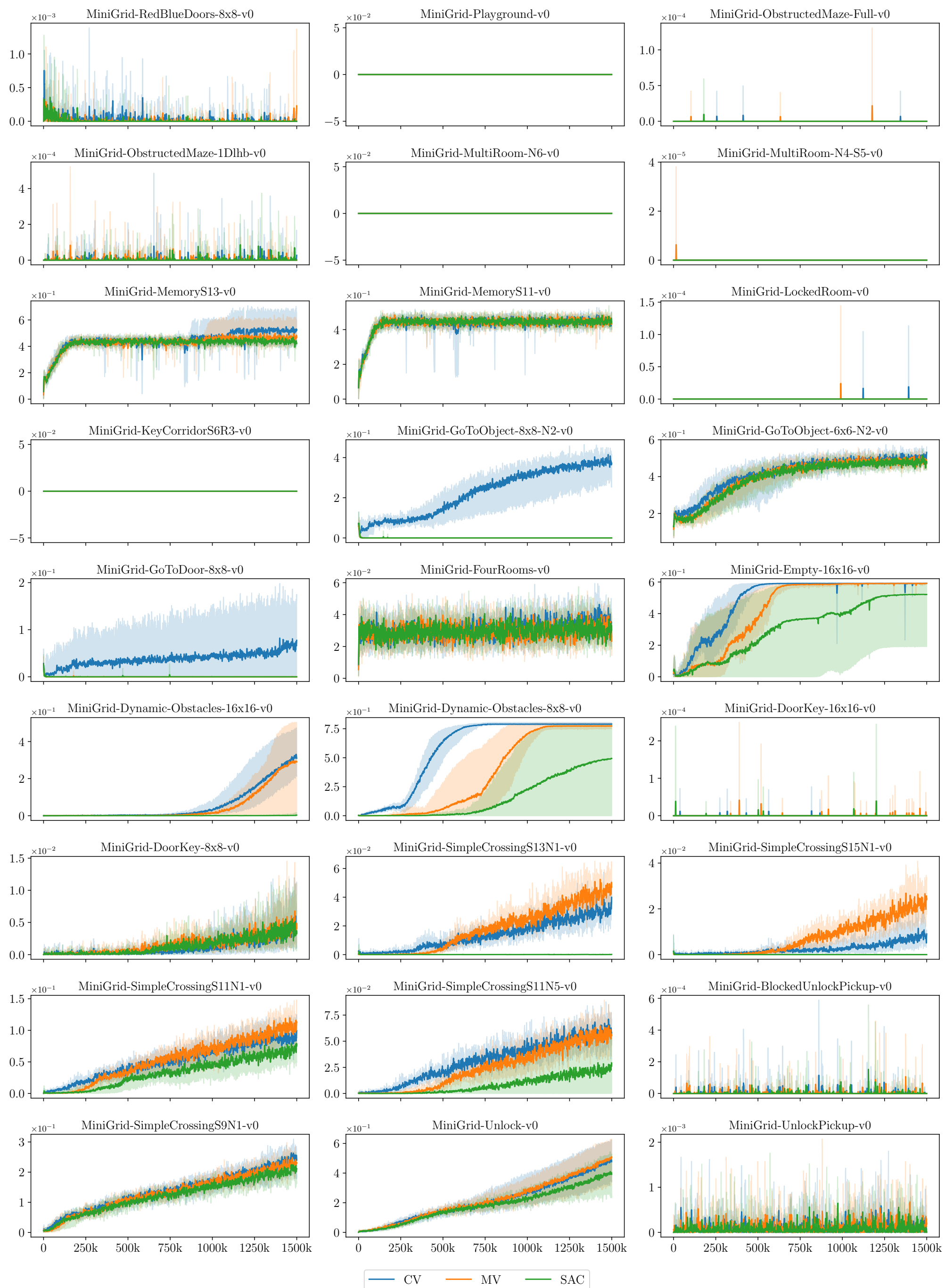

Figure 4: Expected return during the policy optimization. The expectation is computed empirically with Monte Carlo simulations. For each iteration, the interquartile mean over 5 runs is reported, along with its 95% confidence interval.