# Active Exploration for Inverse Reinforcement Learning

**David Lindner** david.lindner@inf.ethz.ch
*Department of Computer Science*
*ETH Zurich*

**Andreas Krause** krausea@ethz.ch
*Department of Computer Science*
*ETH Zurich*

**Giorgia Ramponi** giorgia.ramponi@ai.ethz.ch
*ETH AI Center*

## Abstract

Inverse Reinforcement Learning (IRL) is a powerful paradigm for inferring a reward function from expert demonstrations. Many IRL algorithms require a known transition model and sometimes even a known expert policy, or they at least require access to a generative model. However, these assumptions are too strong for many real-world applications, where the environment can be accessed only through sequential interaction. We propose a novel IRL algorithm: **Ac**tive **e**xploration for **I**nverse **R**einforcement **L**earning (AceIRL), which actively explores an unknown environment and expert policy to quickly learn the expert's reward function and identify a good policy. AceIRL uses previous observations to construct confidence intervals that capture plausible reward functions and find exploration policies that focus on the most informative regions of the environment. AceIRL is the first approach to active IRL with sample-complexity bounds that does not require a generative model of the environment. AceIRL matches the sample complexity of active IRL with a generative model in the worst case. Additionally, we establish a problem-dependent bound that relates the sample complexity of AceIRL to the suboptimality gap of a given IRL problem. We empirically evaluate AceIRL in simulations and find that it significantly outperforms more naive exploration strategies.

**Keywords:** Inverse Reinforcement Learning, Active Learning, Reward-free Exploration

## 1. Introduction

Reinforcement Learning (RL; Sutton and Barto, 2018) has achieved impressive results recently, from playing video games (Mnih et al., 2015) to solving robotic control problems (Haarnoja et al., 2019). However, in many applications, it is challenging to design a reward function that robustly describes the desired task (Amodei et al., 2016; Hendrycks et al., 2021). Instead of using an explicit reward function, Inverse Reinforcement Learning (IRL; Ng et al., 2000) seeks to recover the reward by observing an *expert*, e.g., an human who already knows how to perform a task. However, most existing IRL algorithms assume that the transition model, and in some cases, the expert's policy, are *known*. In many real-world applications, this is not given, and the agent needs to estimate the transition dynamics and the expert policy from samples. Figure 1 shows an illustrative example where the agent can choose between different paths that have different properties, e.g., walking speeds, and lead to different goals. The agent has to explore the environment and query the expert policy in order to infer the expert's reward function.

IRL with sample-based estimation was only recently analyzed formally by Metelli et al. (2021). They decompose the error on the reward into a contribution from estimating the transition model and estimating the expert policy. Based on this, Metelli et al. (2021) propose an efficient sampling strategy to recover a good reward function. However, they assume a *generative model* of the environment, i.e., the agent can query the transition dynamics for arbitrary states and actions. In practice, this assumption is unrealistic. The agent in Figure 1 starts in the middle and cannot learn about the properties of any path without actually *exploring the environment*.

In this work, we consider IRL with unknown transition dynamics and expert policy and focus on exploring the environment in order to recover the expert's reward function efficiently. To the best of our knowledge, we present the first paper providing sample complexity guarantees for the active IRL problem without access to a generative model.

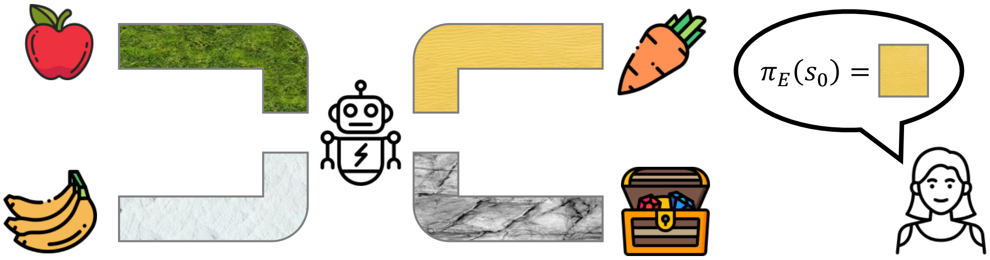

Figure 1: An illustrative example of the Active IRL problem. The agent can choose between four paths that lead to different objects. It can get action recommendations from an expert but does not know about the properties of the different paths (the transition dynamics) or the value of different items (the reward function). Observing the expert actions is not enough to infer a reward function. For example, from observing the expert recommending to take the yellow path, the agent cannot infer that the human prefers to find the carrot. The human might prefer to find the treasure but know it is too hard to reach. Therefore, the agent has to explore the environment and learn about its dynamics to infer a good reward function. AceIRL implements an exploration strategy that aims to infer which reward functions are consistent with the expert's recommendations as quickly as possible. We present experiments on a version of this environment in Section 7.

Our main contributions are:

- We propose the active IRL problem in a finite-horizon, undiscounted Markov Decision Process (MDP) and characterize necessary and sufficient conditions for solving it (Section 4.1).
- We analyze how the estimation errors of the transition model and the expert policy contribute to the estimation error of the reward function, extending prior work to the finite-horizon setting. We provide a novel analysis of how this error affects the performance of the policy, which optimizes the recovered reward function (Section 4.2).
- We propose a novel algorithm, **Ac**tive **e**xploration for **I**nverse **R**einforcement **L**earning (AceIRL), which actively explores the environment and the expert policy to infer a good reward function. In each iteration, AceIRL constructs an exploratory policy based on the estimation error of the recovered reward function (Section 6).
- We consider two different exploration strategies for AceIRL. The first, more straightforward, strategy provides a sample complexity similar to the algorithm proposed by Metelli et al. (2021), which has access to a generative model (Section 6.1). The second strategy takes the expected reduction in uncertainty into account (Section 6.2). This yields a tighter, problem-dependent sample complexity bound at the cost of solving a convex optimization problem in each iteration (Section 6.3).
- We evaluate AceIRL empirically in simulated environments and demonstrate that it achieves significantly better performance than move naive exploration strategies (Section 7).

The proofs of all results presented in the main paper can be found in Appendix B.

## 2. Related Work

Most IRL algorithms assume that the underlying transition model is known (Ratliff et al., 2006; Ziebart et al., 2008; Ramachandran and Amir, 2007; Levine et al., 2011). However, the transition model usually needs to be estimated from samples, which induces an error in the recovered reward function that most papers do not study. Metelli et al. (2021) analyze this error and the sample complexity of IRL in a tabular setting with a generative model. They propose an algorithm focused on transferring the learned reward function to a fully known target environment. Dexter et al. (2021) provides a similar analysis in continuous state spaces and discrete action spaces, but they still require a generative model of the environment. In contrast, we *do not* assume access to a generative model and thus need to tackle the exploration problem in IRL.

Some prior work studies active learning algorithms for IRL in a Bayesian framework but without theoretical guarantees. Lopes et al. (2009) propose an active learning algorithm for IRL that estimates a posterior distribution over reward functions from demonstrations, requiring a prior distribution and full knowledge of the environment dynamics. Relatedly, Cohn et al. (2011) consider a Bayesian IRL setting with a semi-autonomous agent that asks an expert for advice if it is uncertain about the reward. Cohn et al. (2011) consider a semi-autonomous agent who acts autonomously when it is confident and asks a human expert for advice otherwise. Similarly to Lopes et al. (2009), they consider a Bayesian IRL setting to define these acquisition functions, and assume full knowledge of the environment dynamics. Brown et al. (2018) empirically study active IRL in several safety-critical environments, selecting queries using value at risk. Kulick et al. (2013) consider active learning for a robotic manipulation task, asking a human expert for advice in situations with the highest predictive uncertainty. The robot aims to create the most informative situations that are physically feasible to ask a human overseer for advice. The active learning criterion is the predictive uncertainty about the label of a specific situation. Similarly, Losey and O'Malley (2018) propose a method to learn uncertainty estimates from human corrections in a robotics context. All of these papers assume a Bayesian framework and do not provide theoretical guarantees. In contrast, our setup does not require a prior over reward functions, and we provide theoretical sample complexity guarantees for our algorithm.

A separate line of work studies sample complexity in *imitation learning* where the goal is to imitate an expert policy rather than infer a reward function (Rajaraman et al., 2020; Xu et al., 2020). In particular, Abbeel and Ng (2005) also focus on exploration and propose to use the expert policy to explore relevant regions, whereas Shani et al. (2022) use an upper-confidence approach to exploration. Our setting is different because we focus on IRL instead of imitation learning, and we aim to explore to infer a reward function learn most effectively.

## 3. Preliminaries

Let us first introduce necessary background and notation that we use throughout the paper.

**Markov decision process.** A finite-horizon (or episodic) Markov Decision Process without reward function (MDP$\backslash$R) is a tuple $\mathcal{M} := (\mathcal{S}, \mathcal{A}, P, H, s_0)$, where $\mathcal{S}$ is the finite state space of size $S$; $\mathcal{A}$ is the finite action space of size $A$; $P : \mathcal{S} \times \mathcal{A} \to \Delta_{\mathcal{S}}$ is the transition model; $H$ is the horizon and $s_0$ is the initial state. [1] In other words, a finite-horizon MDP$\backslash$R is a finite-horizon MDP (Puterman, 2014) without the reward function. We describe an agent's behaviour with a (possible stochastic) policy $\pi \in \mathcal{S} \times [H] \to \Delta_{\mathcal{A}}$.

**Reward function.** A reward function $r : \mathcal{S} \times \mathcal{A} \times [H] \to [0, R_{\max}]$ maps state-action-time step triplets to a reward. Given an MDP$\backslash$R $\mathcal{M}$ and a reward function $r$, we denote the resulting MDP by $\mathcal{M} \cup r$.

**Value functions and optimality conditions.** We define the *Q-function* $Q^{\pi,h}_{\mathcal{M}\cup r}(s, a)$ and *value-function* $V^{\pi,h}_{\mathcal{M}\cup r}(s)$ of a policy $\pi$ in the MDP $\mathcal{M} \cup r$ at time step $h$, state $s$ and action $a$ as:

$$Q^{\pi,h}_{\mathcal{M}\cup r}(s, a) = r_h(s, a) + \sum_{s',a'} \pi_{h+1}(a'|s')P(s'|s, a)Q^{\pi,h+1}_{\mathcal{M}\cup r}(s', a'); \quad V^{\pi,h}_{\mathcal{M}\cup r}(s) = \sum_{a} \pi_h(a|s)Q^{\pi,h}_{\mathcal{M}\cup r}(s, a)$$

We define the *advantage function* $A^{\pi,h}_{\mathcal{M}\cup r}(s, a)$ as $A^{\pi,h}_{\mathcal{M}\cup r}(s, a) = Q^{\pi,h}_{\mathcal{M}\cup r}(s, a) - V^{\pi,h}_{\mathcal{M}\cup r}(s)$. A policy $\pi$ is optimal if $A^{\pi,h}_{\mathcal{M}\cup r}(s, a)$ is $\leq 0$ for each time step $h \in [H]$, state $s \in \mathcal{S}$, action $a \in \mathcal{A}$. We denote the set of optimal policies for the MDP $\mathcal{M} \cup r$ with $\Pi^*_{\mathcal{M}\cup r}$.

**State-visitation frequencies.** We define $\rho^{h,h'}_{\mathcal{M},\pi}(s|s_0)$ as the probability of being in state $s$ at time $h' \geq h$ following policy $\pi$ in MDP$\backslash$R $\mathcal{M}$ starting in state $s_0$ at time $h$. We can compute it recursively:

$$\rho^{h,h}_{\mathcal{M},\pi}(s'|s) := \mathbb{1}_{\{s'=s\}} \quad \text{and} \quad \rho^{h,h'+1}_{\mathcal{M},\pi}(s'|s) := \sum_{s'',a} P(s'|s'', a)\,\pi_{h'}(a|s'')\,\rho^{h,h'}_{\mathcal{M},\pi}(s''|s).$$

[1] We can model any initial state distribution as a single initial state by modifying the transitions.

# 4. Active Learning for Inverse Reinforcement Learning (Active IRL)

In this section, we first introduce the Active Inverse Reinforcement Learning problem with and without a generative model (Section 4.1). Then, we define the feasible reward set for finite-horizon MDPs (Section 4.2) and characterize the error propagation on the reward function and the value function (Section 4.3), extending results by Metelli et al. (2021) to the finite horizon setting.

## 4.1 Problem Definition

Our goal is to design an exploration strategy to construct a dataset of demonstrations $D$ such that an arbitrary IRL algorithm can recover a *good* reward function from it. To be agnostic to the choice of IRL algorithm, we consider the set of all feasible reward functions for a specific expert policy. Formally, we consider IRL problems $(\mathcal{M}, \pi^E)$ consisting of an MDP$\backslash$R and an expert policy $\pi^E$, and we define the feasible reward set as follows.

**Definition 1 (Feasible Reward Set)** *A reward function $r$ is* feasible *for an IRL problem $(\mathcal{M}, \pi^E)$, if and only if the expert policy $\pi^E$ is optimal in $\mathcal{M} \cup r$. We call the set of all feasible reward functions $\mathcal{R}_{\mathcal{M} \cup \pi^E}$ the* feasible reward set. *If we estimate the transition model and expert policy from samples we refer to the* recovered feasible set $\mathcal{R}_{\hat{\mathfrak{B}}} = \mathcal{R}_{\widehat{\mathcal{M}} \cup \hat{\pi}^E}$ *in contrast to the* exact feasible set $\mathcal{R}_{\mathfrak{B}} = \mathcal{R}_{\mathcal{M} \cup \pi^E}$.

Now, we can formalize the goal of Active IRL as finding a sampling strategy that satisfies the following PAC optimality criterion.

**Definition 2 (Optimality Criterion)** *Let $\mathcal{S}$ be a sampling strategy. Let $\mathcal{R}_{\mathfrak{B}}$ be the exact feasible set and $\mathcal{R}_{\hat{\mathfrak{B}}}$ be the feasible set recovered after observing $n \geq 0$ samples collected from $\mathcal{M}$ and $\pi^E$. We say that $\mathcal{S}$ is $(\epsilon, \delta, n)$-*correct *if with probability at least $1 - \delta$ it holds that:*

$$\inf_{\hat{r} \in \mathcal{R}_{\hat{\mathfrak{B}}}} \sup_{\hat{\pi}^* \in \Pi^*_{\widehat{\mathcal{M}} \cup \hat{r}}} \max_{s,a,h} \left| Q^{\pi^*,h}_{\mathcal{M} \cup r}(s,a) - Q^{\hat{\pi}^*,h}_{\mathcal{M} \cup r}(s,a) \right| \leq \epsilon \qquad \text{for each } r \in \mathcal{R}_{\mathfrak{B}},$$

$$\inf_{r \in \mathcal{R}_{\mathfrak{B}}} \sup_{\pi^* \in \Pi^*_{\mathcal{M} \cup r}} \max_{s,a,h} \left| Q^{\pi^*,h}_{\mathcal{M} \cup r}(s,a) - Q^{\hat{\pi}^*,h}_{\mathcal{M} \cup r}(s,a) \right| \leq \epsilon \qquad \text{for each } \hat{r} \in \mathcal{R}_{\hat{\mathfrak{B}}},$$

*where $\pi^*$ is an optimal policy in $\mathcal{M} \cup r$ and $\hat{\pi}^*$ is an optimal policy in $\widehat{\mathcal{M}} \cup \hat{r}$.*

The first condition states that for each reward in the exact feasible set, the best reward we could estimate in the recovered feasible set has a low error everywhere. This condition is a type of "recall": every possible true reward function needs to be captured by the recovered feasible set. The second condition ensures that there is a possible true reward function with a low error for every possible recovered reward function. This avoids an unnecessarily large recovered feasible set. This condition is a type of "precision": if we recover a reward function, it has to be close to a possible true reward function. Note, that Metelli et al. (2021) consider a similar optimality criterion in their Definition 5.1. However, they consider a known target environment; hence, our Definition 2 is a stronger requirement.

## 4.2 Feasible Rewards in Finite-horizon MDPs

Ng et al. (2000) characterize the feasible reward set implicitly in the infinite horizon setting, whereas Metelli et al. (2021) characterize it explicitly. Here, we provide similar results for a finite horizon.

**Lemma 3 (Feasible Reward Set Implicit)** *A reward function $r$ is feasible if and only if for all $s, a, h$ it holds that: $A^{\pi^E,h}_{\mathcal{M} \cup r}(s,a) = 0$ if $\pi^E_h(a|s) > 0$ and $A^{\pi^E,h}_{\mathcal{M} \cup r}(s,a) \leq 0$ if $\pi^E_h(a|s) = 0$. Moreover, if the second inequality is strict, $\pi^E$ is uniquely optimal, i.e., $\Pi^*_{\mathcal{M} \cup r} = \{\pi^E\}$.*

These two conditions are expressed in the terms of the advantage function since $Q^{\pi^E,h}_{\mathcal{M} \cup r}(s,a) - V^{\pi^E,h}_{\mathcal{M} \cup r}(s) = A^{\pi^E,h}_{\mathcal{M} \cup r}(s,a)$. We can conclude from this lemma that a reward function $r$ belongs to the feasible reward set if the advantage function of the actions played by the agent is equal to $0$, and the advantage function of the actions not played by the agent is non-positive. In fact, if these conditions hold it is easy to see that the expert policy $\pi^E$ is optimal for the reward function $r$. The following lemma characterizes the feasible reward set *explicitly*.

**Lemma 4 (Feasible Reward Set Explicit)** *A reward function $r$ is feasible if and only if there exists an $\{A_h \in \mathbb{R}^{S\times A}_{\geq 0}\}_{h\in[H]}$ and $\{V_h \in \mathbb{R}^S\}_{h\in[H]}$ such that for all $s,a,h$ it holds that:*

$$r_h(s,a) = -A_h(s,a)\mathbb{1}_{\{\pi^E_h(a|s)=0\}} + V_h(s) - \sum_{s'} P(s'|s,a)V_{h+1}(s')$$

Here, the **first term** ensures there is an advantage function for $\pi^E$ and it is $0$ for actions the expert takes and $A_h(s,a)$ for actions the expert does not take. The **second term** corresponds to reward-shaping by the value function.

### 4.3 Error Propagation

Next, we study the error propagation of estimating the transition model $P$ with $\widehat{P}$ and the expert policy $\pi^E$ with $\hat{\pi}^E$. In particular, we bound the estimation error on the reward as a function of the estimation errors of $\widehat{P}$ and $\hat{\pi}^E$, extending a result by Metelli et al. (2021) to the finite horizon setting.

**Theorem 5 (Error Propagation)** *Let $(\mathcal{M},\pi^E)$ and $(\widehat{\mathcal{M}},\widehat{\pi}^E)$ be two IRL problems. Then, for any $r \in \mathcal{R}_{(\mathcal{M},\pi^E)}$ there exists $\widehat{r} \in \widehat{\mathcal{R}}_{(\widehat{\mathcal{M}},\widehat{\pi}^E)}$ such that:*

$$|r_h(s,a) - \widehat{r}_h(s,a)| \leq A_h(s,a)|\pi^E_h(a|s) - \widehat{\pi}^E_h(a|s)| + \sum_{s'} V_{h+1}(s')|P(s'|s,a) - \widehat{P}(s'|s,a)|$$

*and we can bound $V_h \leq (H-h)R_{\max}$ and $A_h \leq (H-h)R_{\max}$.*

It provides a bound on the distance between each reward function in the real feasible reward set $\mathcal{R}_{\mathfrak{B}}$ to the closest one in the estimated one $\mathcal{R}_{\widehat{\mathfrak{B}}}$. The error depends on the two estimated components and this is reflected as the sum of two terms, one depending on the estimation of the expert policy and the other of the transition model. The first is the error on the estimation of the expert policy; in fact, and this error is due to the fact that the recovered policy gives probability greater than zero to play an action never played by the expert. The second term depends only on the estimation of the transition model.

In IRL, we cannot hope to recover the expert's reward function perfectly. Instead, we aim to estimate a reward function that leads to an optimal policy with performance close to the expert's policy under the (unknown) real reward function. For example, suppose a specific state $s$ is difficult to reach in the environment. In that case, the error on the reward function $r(s,\cdot)$ will not impact the performance of the induced policy much. Formally, we are interested in studying the error propagation to the optimal value function. The next lemma will be crucial for analyzing this.

## 5. Recovering Feasible Rewards with a Generative Model

As a warmup, let us first study the sample complexity of a simple *uniform sampling* strategy with access to the generative model of $\mathcal{M}$. We assume we can query a generative model about a state-action pair $(s,a)$ to receive a next state $s' \sim P(\cdot|s,a)$ and an expert action $a_E \sim \pi^E(\cdot|s)$. This allows us to introduce key ideas and serves as a baseline to compare later results to. We adapt the infinite-horizon results by Metelli et al. (2021) to the finite-horizon setting, and our stronger PAC requirement in Definition 2. We first discuss how we can estimate the transition model and the policy (Section 5.1) before stating the sample complexity of the uniform sampling strategy (Section 5.2).

### 5.1 Estimating Transition Model and Expert Policy

In each iteration $k$, let $n^h_k(s,a,s')$ be the number of times we observed the transitions $(s,a,s)$ at time $h$ up to iteration $k$. Also, we define $n^h_k(s,a) = \sum_{s'} n^h_k(s,a,s')$, and $n^h_k(s) = \sum_a n^h_k(s,a)$. Then we can estimate the transition model and expert policy by

$$\hat{P}_k(s'|s,a) = \frac{\sum_{h=1}^{H} n_k^h(s,a,s')}{\max(1, \sum_{h=1}^{H} n_k^h(s,a))} \qquad \hat{\pi}^E_{k,h}(a|s) = \frac{n_k^h(s,a)}{\max(1, n_k^h(s))}.$$

In Appendix B.3 we derive Hoeffding's confidence intervals for the transition model and the expert policy. Combining these with Theorem 5, we can compute the uncertainty on the recovered reward as:

$$C_k^h(s,a) = (H-h)R_{\max}\min\left(1, 2\sqrt{\frac{2\ell_k^h(s,a)}{n_k^h(s,a)}}\right),$$

where $\ell_k^h(s,a) = \log\left(24SAH(n_k^h(s,a))^2/\delta\right)$. We can show that for any pair of reward functions $r \in \mathcal{R}_{\mathfrak{B}}$ and $\hat{r} \in \mathcal{R}_{\hat{\mathfrak{B}}}$, the difference $|r_h(s,a) - \hat{r}_{k,h}(s,a)| \leq C_k^h(s,a)$. This uncertainty estimate will be a key component in all of our theoretical analysis.

### 5.2 Uniform Sampling Strategy

In each iteration $k$, the *uniform sampling* strategy allocates $n_{\max}$ samples uniformly over $[H] \times S \times A$. It estimates the reward uncertainty and stops as soon as $H \max_{h,s,a} C_k^h(s,a) \leq \epsilon$. The next theorem characterizes the sample complexity of uniform sampling with a generative model.

**Theorem 6 (Sample Complexity of Uniform Sampling IRL)** *The uniform sampling strategy fulfills Definition 2 with a number of samples upper bounded by:*

$$n \leq \tilde{\mathcal{O}}\left(H^5 R_{\max}^2 SA/\epsilon^2\right),$$

*where $\tilde{\mathcal{O}}$ suppresses logarithmic terms.*

This sample complexity bound appears slightly worse than the one in Metelli et al. (2021), who find $(1-\gamma)^{-4}$ which would translate to $H^4$. This is, however, due to the fact that we consider reward functions that can depend on the timestep $h$. If we assume the reward function does not depend on $h$, we gain a factor of $H$, obtaining the same result.

## 6. Active Exploration for Inverse Reinforcement Learning

Let us now turn to our original problem: recovering the expert's reward function in an unknown environment *without* a generative model. This problem is harder since we need to create an exploration strategy to acquire the desired information about the environment. We now propose a novel sample-efficient exploration algorithm for IRL that we call **Ac**tive **e**xploration for **I**nverse **R**einforcement **L**earning (AceIRL). The algorithm takes inspiration from recent works on reward-free exploration (Kaufmann et al., 2021) and exploration strategies in RL (Auer et al., 2008). We divide the explanation of the algorithm into two parts. First, we introduce a simplified version of the algorithm, which comes with a problem independent sample complexity result (Section 6.1). Next, we introduce the full algorithm, which considers the expected reduction of uncertainty in the next iteration to improve exploration and maintains a confidence set of plausibly optimal policies to focus on the most relevant regions (Section 6.2). The full algorithm provides a tighter, problem-dependent sample complexity bound (Section 6.3). Algorithm 1 contains pseudo-code of AceIRL, and Appendix B contains the detailed theoretical analysis including proofs of all results discussed here.

### 6.1 Uncertainty-based Exploration for IRL

The first idea of AceIRL is similar to reward-free UCRL (Kaufmann et al., 2021). Our goal is to fulfill the PAC requirement in Definition 2. Hence, we start from an upper bound on the estimation error between the performance of the optimal policy $\hat{\pi}^*$ for a reward $\hat{r} \in \mathcal{R}_{\hat{\mathfrak{B}}}$ in the recovered feasible set and the optimal policy $\pi^*$ for a reward function $r \in \mathcal{R}_{\mathfrak{B}}$ in the true MDP $\mathcal{M}$. We will then use this upper bound to drive the exploration. For each timestep $h$ and iteration $k$, we define the error:

$$\hat{e}_k^h(s,a;\pi^*,\hat{\pi}^*) = Q^{\pi^*,h}_{\mathcal{M}\cup r}(s,a) - Q^{\hat{\pi}^*,h}_{\mathcal{M}\cup r}(s,a). \tag{1}$$

**Algorithm 1** AceIRL algorithm for IRL in an unknown environment.

```
Input: significance $\delta \in (0,1)$, target accuracy $\varepsilon$, IRL algorithm $\mathcal{A}$, number of episodes $N_E$
Initialize $k \leftarrow 0$, $\varepsilon_0 \leftarrow H/10$
while $\varepsilon_k > \varepsilon/4$ do
    Solve (convex) optimization problem (ACE) to obtain $\pi_k$
    Explore with policy $\pi_k$ for $N_E$ episodes, observing transitions and expert actions
    $k \leftarrow k + 1$
    Update $\hat{P}_k$, $\hat{\pi}_k$, $C_k^h$, and $\hat{r}_k \leftarrow \mathcal{A}(\mathcal{R}_{\hat{\mathfrak{B}}})$
    Update accuracy $\varepsilon_k \leftarrow \max_a \hat{E}_k^0(s_0, a)$
end while
return Estimated reward function $\hat{r}_k$
```

We can define an upper bound on these errors recursively with $C_k^H(s,a) = 0$ and

$$E_k^h(s,a) = \min\left((H-h)R_{\max}, C_k^h(s,a) + \sum_{s'} \hat{P}(s'|s,a) \max_{a' \in A} E_k^{h+1}(s', a')\right). \tag{EB1}$$

It is straightforward to show that $\hat{e}_k^h(s,a;\pi,\hat{\pi}) \leq E_k^h(s,a)$ for any two policies $\pi, \hat{\pi}$. Using this error bound, we can introduce a simplified version of AceIRL that explores greedily with respect to $E_k^h(s,a)$. We call this algorithm "AceIRL Greedy". Note that this is equivalent to solving the RL problem defined by $\mathcal{M} \cup C_k^h$; hence, we can use any RL solver to find the exploration policy in practice. If we explore with this greedy policy, we can stop if:

$$4 \max_a E_k^0(s_0, a) \leq \varepsilon. \tag{SP1}$$

We can show that when this stopping condition holds, the solution fulfills the PAC requirement 2. Furthermore, we show in Appendix B.4 that AceIRL Greedy achieves a sample complexity on order $\tilde{\mathcal{O}}\left(H^5 R_{\max}^2 SA/\varepsilon^2\right)$, which matches the sample complexity of uniform sampling *with a generative model*. This is already a strong result implying that we do not need a generative model to achieve a good sample complexity for IRL. However, it turns out we can improve the algorithm further.

### 6.2 Problem Dependent Exploration

AceIRL Greedy is limited in two ways: (i) it explores states that have high uncertainty so far, whereas our goal is to reduce uncertainty *in the next iteration*, and (ii) it explores to reduce the uncertainty about all policies, whereas our goal is to reduce the uncertainty primarily about *plausibly optimal* policies. To address these limitations, we propose two modifications that yield the full AceIRL algorithm.

**Reducing future uncertainty.** The greedy policy w.r.t. $E_k^h$ explores states in which the estimation error on the Q-functions is large. However, note that this is not exactly what we want, namely, to reduce the uncertainty the most. In particular, if we explore for more than one episode before updating the exploration policy, we should choose an exploration policy that considers how the uncertainty will reduce during exploration. Ideally, we would explore with a policy that minimizes $E_{k+1}^h$. However, we cannot compute this quantity exactly. Instead, we can approximate it using our current estimate of the transition model. Concretely, if we have an exploration policy $\pi$, we can estimate the reward uncertainty at the next iteration as:

$$\hat{C}_{k+1}^h(s,a) = (H-h)R_{\max} \min\left(1, 2\sqrt{\frac{2\ell_k^h(s,a)}{n_k^h(s,a) + \hat{n}^h(s,a)}}\right),$$

where $\hat{n}^h(s,a) = N_E \, \eta^{\pi',h}_{\hat{M}}(s,a|s_0)$ is the expected number of times $\pi$ visits $(s,a)$ at time $h$ and $N_E$ is the number of episodes we will explore with $\pi$. We can use this estimate to find a policy that minimizes our estimate of $E_k^{h+1}$. While our original approach was akin to "uncertainty sampling", we now have a better way to measure the "informativeness" of choosing an exploration policy. This is a common pattern when designing query strategies in active learning (Settles, 2012). Note, that this argument does not rely on the IRL problem and can be used to independently improve algorithms for reward-free exploration (cf. Appendix D).

**Focusing on plausibly optimal policies.** By exploring greedily w.r.t. $E_k^h$, we reduce the estimation error of all policies. However, we are primarily interested in estimating the distance between policies $2 \quad _{M [ r}$ and $\hat{} \quad 2 \quad _{M [ \hat{r}}$ with $r \ 2 \ R_{B}$ and $\hat{r} \ 2 \ R_{\hat{B}}$. Of course, we do not know these sets, so we cannot use them directly to target the exploration. Instead, assume we can construct a set of plausibly optimal policies $\hat{}_k$ that contains all and $\hat{}_k$ with high probability. Then, we can redefine our upper bounds recursively as $\hat{E}_k^H(s;a) = 0$ and:

$$\hat{E}_k^h(s;a) = \min \ (H \quad h)R_{\max}; C_k^h(s;a) + \sum_{s^0} \hat{P}(s^0 j s;a) \max_{2 \hat{}_{k \ 1}} (a^0 j s^0)\hat{E}_k^{h+1}(s^0;a^0) \ ; \tag{EB2}$$

In contrast to (EB1), we maximize over policies in $\hat{}_k$ rather than all actions. Acting greedily with respect to $\hat{E}_k^h(s;a)$ is equivalent to finding the optimal policy $_k \ 2 \ \hat{}_k$ for the RL problem defined by $M \ [ \ C_k^h$. To construct the set of plausibly optimal policies, we use an arbitrary IRL algorithm $A$ . We only assume that $A$ will return a reward function $\hat{r}_k \ 2 \ R_{\hat{B}}$. Then, we can construct a set of plausibly optimal policies as $\hat{}_k = f \ jV^{\ ;}_{\hat{M} [ \hat{r}_k}(s_0) \quad V^{\ ;}_{\hat{M} [ \hat{r}_k}(s_0) \quad 10 \ _k g$. We show in Appendix B.5 that $\hat{}_k$ contains both and $\hat{}_k$ with high probability. This choice is based on ideas by Zanette et al. (2019).

We can define a stopping condition analogously to (SP1):

$$4 \max_a \hat{E}_k^0(s_0;a) \qquad : \tag{SP2}$$

Again, we can prove that if the algorithms stops due to (SP2), then $R_{\hat{B}}$ respects Definition 2.

**Implementing AceIRL.** To implement the full algorithm, we need to solve an optimization problem:

$$_k \ 2 \ \mathrm{argmin} \max_{\hat{} 2 \hat{}_{k \ 1}} \ \hat{E}^0_{k+1}(s_0; \hat{}(s_0)) \tag{ACE}$$

The solution to this problem is the exploration policy that minimizes the uncertainty at the next iteration about plausibly optimal policies. This problem might seem difficult to solve at first, but, perhaps surprisingly, it can be formulated as a convex optimization problem solvable with standard techniques (cf. Appendix B.6).

### 6.3 Sample Complexity of AceIRL

In this section, we present our main result about the sample complexity of AceIRL. The result is problem-dependent, and, in particular, depends on the advantage function $A^{\ ;h}_{M [ r}(s;a)$, where $r$ is the reward function in the exact feasible set $R_{B}$ closest to the reward function $\hat{r}_k$ which belongs to the estimated feasible set $R_{\hat{B}}$. The advantage function $A^{\ ;h}_{M [ r}(s;a)$ acts similarly to a suboptimality gap: the closer the value of the second best action is to the best action, the harder it is to identify the best action and infer the correct reward function.

**Theorem 7** *[AceIRL Sample Complexity] AceIRL returns a ( , , $n$)-correct solution with*

$$n \quad O \quad \min \ " \ \frac{H^5 R_{\max}^2 SA}{2}; \frac{H^4 R_{\max}^2 SA \ \ ^2 \ \ _1}{\min_{s;a;h}(A^{\ ;h}_{M [ r}(s;a))^2 \ \ ^2} \ \# \, !$$

*where $_1$ depends on the choice of $N_E$, the number of episodes of exploration in each iteration. $A^{\ ;h}_{M [ r}(s;a)$ is the advantage function of $r \ 2 \ \mathrm{argmin}_{r 2 R_{B}} \max_{h;s;a}(r_h(s;a) \quad \hat{r}_{k;h}(s;a))$, the reward function from the feasible set $R_{B}$ closest to the estimated reward function $\hat{r}_k$.*

This result is the minimum of two terms. The first term is problem independent and it is achieved both by AceIRL Greedy and the full AceIRL. This bound matches the bound we saw previously with a generative model. Hence, AceIRL achieves the same results without access to the generative model. Using (ACE) can yield a better sample complexity, represented by the second term in the minimum. This bound depends on two main components: the ratio $_1 =$ and the advantage function $A^{\ ;h}_{M [ r}(s;a)$. The ratio depends on the choice of $N_E$, the number of exploration episodes per iteration. If $N_E$ is small, then the -ratio will be also small. If $N_E$ is large the algorithm will perform similarly to a uniform sampling strategy. Appendix B.5 provides the full proof of this theorem.

| | Uniform sampling (gener. model) | | TRAVEL (gener. model) (Metelli et al., 2021) | | Random Exploration | | AceIRL Greedy | | AceIRL (Full) | |
|---|---|---|---|---|---|---|---|---|---|---|
| Four Paths (Figure 1) | 1900 | 71 | | | 17840 | 1886 | | | | |
| – $N_E = 50$ | | | 1560 | 76 | | | 24180 | 1747 | **10780** | **1369** |
| – $N_E = 100$ | | | 2000 | 0 | | | 32760 | 2172 | 14080 | 1603 |
| – $N_E = 200$ | | | 4000 | 0 | | | 52000 | 4057 | 16160 | 2033 |
| Double Chain (Kaufmann et al., 2021) | 1980 | 66 | | | 23640 | 2195 | | | | |
| – $N_E = 50$ | | | 1120 | 46 | | | 16240 | 842 | **11580** | **870** |
| – $N_E = 100$ | | | 2000 | 0 | | | 22200 | 1329 | 15440 | 1031 |
| – $N_E = 200$ | | | 4000 | 0 | | | 37200 | 1664 | 20400 | 1629 |
| Metelli et al. (2021): | | | | | | | | | | |
| Random MDPs ($N_E = 1$) | 22 | 1 | 27 | 1 | **22** | **1** | 23 | 1 | **21** | **1** |
| Chain ($N_E = 1$) | 78 | 2 | 76 | 4 | 161 | 8 | 153 | 8 | **142** | **9** |
| Gridworld ($N_E = 1$) | 43 | 2 | 35 | 2 | **45** | **2** | **46** | **3** | **48** | **2** |

Table 1: Sample complexity of AceIRL compared to random exploration and methods that use a generative model. We show the number of samples necessary to find a policy with normalized regret less than $0{:}4$. We report means and standard errors computed over $50$ random seeds each. For each environment, we highlight in **bold** the method that achieves the best performance without access to a generative model. If multiple methods are within one standard error distance, we highlight all of them. Overall, AceIRL is the most sample efficient method without a generative model if $N_E$ is chosen small enough. In Appendix C.3, we show learning curves for all individual experiments.

## 7. Experiments

We perform a series of simulation experiments to evaluate AceIRL. We simulate a (deterministic) expert policy using an underlying true reward function, and compare it to the recovered reward functions. Our main evaluation metric is a *normalized regret*:

$$V_{M\,[\,r}^{\;;0}(s_0) \quad V_{M\,[\,r}^{\hat{}\;;0}(s_0) = V_{M\,[\,r}^{\;;0}(s_0) \quad V_{M\,[\,r}^{\;;0}(s_0)\;;$$

where is the optimal policy for $M\ [\ r$, $\hat{}$ is the optimal policy for $\hat{M}\ [\ \hat{r}$, and is the worst possible policy for $r$, i.e., the optimal policy for $M\ [\ (\ r)$.

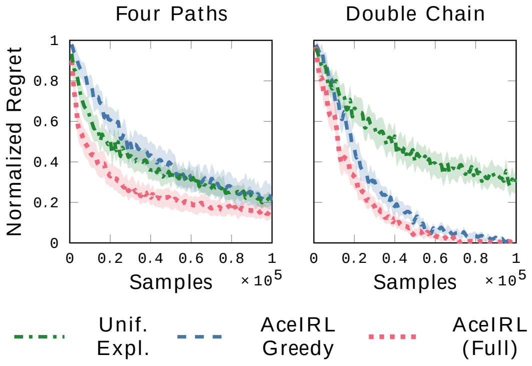

Figure 2: Normalized regret (lower is better) of the policy optimizing for the inferred reward in the estimated MDP as a function of the number of samples. The plots show the mean and 95% confidence intervals computed using 50 random seeds. We use $N_E = 50$.

We introduce the *Four Paths* environment shown in Figure 1, which consists of four chains of states that have different randomly sampled transition probabilities. One path has a goal with reward $1$; all other rewards are $0$. We also evaluate on *Random MDPs* with uniformly sampled transition dynamics and reward functions, the *Double Chain* environment proposed by Kaufmann et al. (2021), and the *Chain* and *Gridworld* environments proposed by Metelli et al. (2021). Appendix C.1 provides details on the transition dynamics and rewards of all environments.

We compare AceIRL and AceIRL Greedy to a uniformly random exploration policy, as a naive exploration strategy. Further, we consider uniform sampling with a generative model as well as TRAVEL (Metelli et al., 2021), which can be more sample efficient because they do not need to explore the environment. Note that TRAVEL is designed to learn a reward to be transferred to a known target environment. Instead, we use a modified version that uses the estimated MDP as a target. Appendix C.2 provides more details on our implementations.

Table 1 shows the sample efficiency of all algorithms in all environments, measured as the number of samples needed to achieve a regret threshold of $0{:}4$ (different thresholds yield similar conclusions; cf. Appendix C). AceIRL is the most sample efficient exploration strategy without access to a generative model; but the relative differences between the methods depend on the environment. In some cases, AceIRL even performs comparably to methods using a generative model.

In the *Four Paths* and *Double Chain* environments, we also vary the $N_E$ parameter. AceIRL performs better for small values at the computational cost of updating the exploration policy more often. If $N_E$ is too large, using AceIRL can be as bad as a uniformly random exploration policy. Increasing $N_E$ hurts the performance of AceIRL Greedy more severely, which does not consider $N_E$ explicitly. Figure 2 shows the normalized regret as a function of the number of samples in *Four Paths* and *Double Chain*. In both cases AceIRL performs best. However, AceIRL Greedy is worse than random exploration in the *Four Paths* environment. Hence, we find that the problem dependent exploration strategy of the full algorithm significantly improves the sample efficiency.

## 8. Conclusion

We considered active inverse reinforcement learning (IRL) with unknown transition dynamics and expert policy and introduced AceIRL, an efficient exploration strategy to learn about both the dynamic and the expert policy with the goal of inferring the reward function as efficiently as possible.

Our approach is a crucial step towards IRL algorithms with theoretical guarantees, but future work is needed to move to more practical settings. In particular, it would be interesting to extend the approach to continuous state and action spaces (e.g. using function approximation), and to obtain an efficient algorithm that does not require solving convex optimization problems. From a theoretical perspective, it would be useful to derive a lower bound on the sample complexity of the IRL problem, to understand if the IRL problem is inherently more difficult than usual RL. Beyond IRL, some of our methods could be useful for other settings, such as reward-free exploration (cf. Appendix D).

Sample efficient IRL is a promising way to apply RL in situations where there is no well-specified reward function available. Of course, even robust IRL algorithms pose a risk of misuse. But, we are optimistic that these methods will overall lead to safer RL systems that can be used in real applications.

## Acknowledgments

This project has received funding from the Microsoft Swiss Joint Research Center (Swiss JRC), Google Brain, and from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme grant agreement No 815943. We thank Bhavya Sukhija for valuable feedback on an earlier draft.

## References

Pieter Abbeel and Andrew Y Ng. Exploration and apprenticeship learning in reinforcement learning. In *Proceedings of International Conference on Machine Learning (ICML)*, 2005.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in AI safety. *arXiv:1606.06565*, 2016.

Peter Auer, Thomas Jaksch, and Ronald Ortner. Near-optimal regret bounds for reinforcement learning. *Advances in Neural Information Processing Systems*, 2008.

Daniel S Brown, Yuchen Cui, and Scott Niekum. Risk-aware active inverse reinforcement learning. In *Conference on Robot Learning (CoRL)*, 2018.

Robert Cohn, Edmund Durfee, and Satinder Singh. Comparing action-query strategies in semi-autonomous agents. In *AAAI Conference on Artificial Intelligence*, 2011.

Gregory Dexter, Kevin Bello, and Jean Honorio. Inverse reinforcement learning in a continuous state space with formal guarantees. In *Advances in Neural Information Processing Systems*, 2021.

Steven Diamond and Stephen Boyd. CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83):1–5, 2016.

Tuomas Haarnoja, Sehoon Ha, Aurick Zhou, Jie Tan, George Tucker, and Sergey Levine. Learning to walk via deep reinforcement learning. In *Proceedings of Robotics: Science and Systems (RSS)*, 2019.

Dan Hendrycks, Nicholas Carlini, John Schulman, and Jacob Steinhardt. Unsolved problems in ML safety. *arXiv:2109.13916*, 2021.

Thomas Jaksch, Ronald Ortner, and Peter Auer. Near-optimal regret bounds for reinforcement learning. *Journal of Machine Learning Research*, 11:1563–1600, 2010.

Chi Jin, Akshay Krishnamurthy, Max Simchowitz, and Tiancheng Yu. Reward-free exploration for reinforcement learning. In *Proceedings of International Conference on Machine Learning (ICML)*, 2020.

Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *Proceedings of International Conference on Machine Learning (ICML)*, 2002.

Emilie Kaufmann, Pierre Ménard, Omar Darwiche Domingues, Anders Jonsson, Edouard Leurent, and Michal Valko. Adaptive reward-free exploration. In *Algorithmic Learning Theory*, 2021.

Johannes Kulick, Marc Toussaint, Tobias Lang, and Manuel Lopes. Active learning for teaching a robot grounded relational symbols. In *Proceedings of International Joint Conferences on Artificial Intelligence*, 2013.

Sergey Levine, Zoran Popovic, and Vladlen Koltun. Nonlinear inverse reinforcement learning with Gaussian processes. *Advances in Neural Information Processing Systems*, 2011.

Manuel Lopes, Francisco Melo, and Luis Montesano. Active learning for reward estimation in inverse reinforcement learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 2009.

Dylan P Losey and Marcia K O'Malley. Including uncertainty when learning from human corrections. In *Conference on Robot Learning (CoRL)*, 2018.

Alberto Maria Metelli, Giorgia Ramponi, Alessandro Concetti, and Marcello Restelli. Provably efficient learning of transferable rewards. In *Proceedings of International Conference on Machine Learning (ICML)*, 2021.

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.

Andrew Y Ng, Stuart J Russell, et al. Algorithms for inverse reinforcement learning. In *Proceedings of International Conference on Machine Learning (ICML)*, 2000.

Brendan O'Donoghue, Eric Chu, Neal Parikh, and Stephen Boyd. Conic optimization via operator splitting and homogeneous self-dual embedding. *Journal of Optimization Theory and Applications*, 169(3):1042–1068, June 2016. URL http://stanford.edu/~boyd/papers/scs.html.

Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

Nived Rajaraman, Lin Yang, Jiantao Jiao, and Kannan Ramchandran. Toward the fundamental limits of imitation learning. In *Advances in Neural Information Processing Systems*, 2020.

Deepak Ramachandran and Eyal Amir. Bayesian inverse reinforcement learning. In *Proceedings of International Joint Conferences on Artificial Intelligence*, 2007.

Nathan D Ratliff, J Andrew Bagnell, and Martin A Zinkevich. Maximum margin planning. In *Proceedings of International Conference on Machine Learning (ICML)*, 2006.

Burr Settles. *Active Learning*. Morgan & Claypool Publishers, 2012.

Lior Shani, Tom Zahavy, and Shie Mannor. Online apprenticeship learning. In *AAAI Conference on Artificial Intelligence*, 2022.

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

Tian Xu, Ziniu Li, and Yang Yu. Error bounds of imitating policies and environments. In *Advances in Neural Information Processing Systems*, 2020.

Andrea Zanette, Mykel J Kochenderfer, and Emma Brunskill. Almost horizon-free structure-aware best policy identification with a generative model. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, 2019.

Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, Anind K Dey, et al. Maximum entropy inverse reinforcement learning. In *AAAI Conference on Artificial Intelligence*, 2008.

# Appendix

## Table of Contents

## Appendix A. Overview of Notation

In Table A.1, we provide a reference of the notation and symbols used in our paper.

## Appendix B. Proofs of Theoretical Results

### B.1 Simulation Lemmas

In this section, we establish several simulation lemmas that we will use throughout our analysis. Some of the results were already derived in prior work for the infinite horizon setting, e.g., by Zanette et al. (2019) and Metelli et al. (2021). For completeness, we provide proofs for all results in the finite-horizon setting.

**Definition 1 (Occupancy measures)** *We define $\eta^{h,h'}_{\mathcal{M},\pi}(s|s_0)$ as the probability of being in state $s$ at timestep $h' \geq h$ following a policy $\pi$ in MDP\R $\mathcal{M}$ starting in state $s_0$ at timestep $h$. We can compute it recursively as:*

$$\eta^{h,h}_{\mathcal{M},\pi}(s'|s) := \mathbb{1}_{\{s'=s\}}$$
$$\eta^{h,h'+1}_{\mathcal{M},\pi}(s'|s) := \sum_{s'',a} P(s'|s'',a)\,\pi_{h'}(a|s'')\,\eta^{h,h'}_{\mathcal{M},\pi}(s''|s)$$

Table A.1: Overview of our notation

| Symbol | Name | Signature |
|---|---|---|
| $M$ | Markov decision process without reward (MDP\R) | $(S, A, P, H, s_0)$ |
| $S$ | State space | |
| $A$ | Action space | |
| $P$ | Transition model | $S \times A \to \Delta_S$ |
| $H$ | Horizon | $H \in \mathbb{N}^+$ |
| $s_0$ | Initial state | $s_0 \in S$ |
| $\pi$ | Policy | $S \times [H] \to \Delta_A$ |
| $r$ | Reward function | $S \times A \times [H] \to [0, R_{\max}]$, $R_{\max} \in \mathbb{R}^+$ |
| $M \cup r$ | Markov decision process (MDP) | $(S, A, P, H, s_0, r)$ |
| $Q^{\pi,h}_{M \cup r}$ | Q-function (of $\pi$ in $M \cup r$) | $S \times A \times [H] \to \mathbb{R}$ |
| $V^{\pi,h}_{M \cup r}$ | Value function (of $\pi$ in $M \cup r$) | $S \times [H] \to \mathbb{R}$ |
| $A^{\pi,h}_{M \cup r}$ | Advantage function (of $\pi$ in $M \cup r$) | $S \times A \times [H] \to \mathbb{R}$ |
| $\eta^{h,\pi}_{M,\pi}(\cdot \mid s_0)$ | State-visitation frequency (conditioned on state) | $[H] \to \Delta_S$ |
| $\eta^{h,\pi}_{M,\pi}(\cdot \mid s_0, a_0)$ | State-visitation frequency (conditioned on state-action) | $[H] \to \Delta_S$ |
| $\eta^{h,\pi}_{M,\pi}(\cdot, \cdot \mid s_0)$ | State-action-visitation frequency (conditioned on state) | $[H] \times S \to \Delta_A$ |
| $\eta^{h,\pi}_{M,\pi}(\cdot, \cdot \mid s_0, a_0)$ | State-action-visitation frequency (conditioned on state) | $[H] \times S \to \Delta_A$ |
| $\mathcal{R}_{M \cup r}$ | Feasible set of $M \cup r$ | |
| $\mathcal{R}_{\mathfrak{B}} = \mathcal{R}_{M \cup \pi_E}$ | Exact feasible set | |
| $\mathcal{R}_{\hat{\mathfrak{B}}} = \mathcal{R}_{\hat{M} \cup \hat{\pi}_E}$ | Recovered feasible set | |
| $\epsilon$ | Target accuracy | $\epsilon \in \mathbb{R}^+$ |
| $\delta$ | Significancy | $\delta \in (0, 1)$ |
| $N_E$ | Number of exploration episodes | $N_E \in \mathbb{N}^+$ |

*We define the same probability for state-action pairs analogously:*

$$\eta^{h,h'}_{M,\pi}(s', a' \mid s, a) := \mathbb{1}_{\{s'=s, a'=a\}}$$

$$\eta^{h,h'+1}_{M,\pi}(s', a' \mid s, a) := \sum_{\bar{s}, \bar{a}} \pi_{h'}(a' \mid s') P(s' \mid \bar{s}, \bar{a}) \, \eta^{h,h'}_{M,\pi}(\bar{s}, \bar{a} \mid s, a)$$

*as well as*

$$\eta^{h,h}_{M,\pi}(s', a' \mid s) := \pi_h(a' \mid s') \mathbb{1}_{\{s'=s\}}$$

$$\eta^{h,h'+1}_{M,\pi}(s', a' \mid s) := \sum_{\bar{s}, \bar{a}} \pi_{h'}(a' \mid s') P(s' \mid \bar{s}, \bar{a}) \, \eta^{h,h'}_{M,\pi}(\bar{s}, \bar{a} \mid s)$$

*Because the environment is Markovian, it also holds for $h' > h$ that*

$$\eta^{h,h'}_{M,\pi}(s' \mid s) = \sum_{\bar{s}, a} \eta^{h+1,h'}_{M,\pi}(s' \mid \bar{s}) P(\bar{s} \mid s, a) \, \pi_h(a \mid s)$$

*and equivalently for state-action pairs.*

**Lemma 2** *The value function and Q-function of a policy $\pi$ in an MDP $M \cup r$ at timestep $h$ can be expressed as:*

$$V^{\pi,h}_{M \cup r}(s) = \sum_{h'=h}^{H} \sum_{s', a'} \eta^{h,h'}_{M,\pi}(s', a' \mid s) \, r_{h'}(s', a')$$

$$Q^{\pi,h}_{M \cup r}(s, a) = \sum_{h'=h}^{H} \sum_{s', a'} \eta^{h,h'}_{M,\pi}(s', a' \mid s, a) \, r_{h'}(s', a')$$

**Proof** We show the result for the value function; the derivation for the Q-function is analogous.

Note that for $h = H$ the statement holds because $V^{\pi,H}_{M\cup r}(s) = 0$. The general result follows by induction. Assume that for $h+1$ the statement holds. Then:

$$
\begin{aligned}
V^{\pi,h}_{M\cup r}(s) &\overset{(a)}{=} \sum_{a} \pi_h(a|s)\left(r_h(s,a) + \sum_{s'} P(s'|s,a) V^{\pi,h+1}_{M\cup r}(s')\right) \\
&\overset{(b)}{=} \sum_{a} \pi_h(a|s)\left(r_h(s,a) + \sum_{s'} P(s'|s,a)\left(\sum_{h'=h+1}^{H}\sum_{s'',a''} \eta^{h+1,h'}_{M,\pi}(s'',a''|s') r_{h'}(s'',a'')\right)\right) \\
&\overset{(c)}{=} \sum_{a} \pi_h(a|s) r_h(s,a) + \sum_{h'=h+1}^{H}\sum_{s',a'} \eta^{h,h'}_{M,\pi}(s'|s)\, \pi_{h'}(a'|s') r_{h'}(s',a') \\
&\overset{(d)}{=} \sum_{h'=h}^{H}\sum_{s',a'} \eta^{h,h'}_{M,\pi}(s'|s)\, \pi_{h'}(a'|s') r_{h'}(s',a')
\end{aligned}
$$

where (a) uses the Bellman equation, (b) the induction step, (c) uses Definition 1 and relabels $s'' \rightarrow s'$, $a'' \rightarrow a'$, and (d) uses Definition 1 again and relabels $a \rightarrow a'$. ■

**Lemma 3 (Simulation lemma 1 by Metelli et al. (2021))** *Let $M$ be an MDP$\backslash$R, and $r, \hat{r}$ two reward functions with corresponding optimal policies $\pi, \hat{\pi}$. Then,*

$$
Q^{\pi,h}_{M\cup r}(s,a) - Q^{\hat{\pi},h}_{M\cup \hat{r}}(s,a) \le \sum_{h'=h}^{H}\sum_{s',a'} \eta^{h,h'}_{M,\pi}(s',a'|s,a)\,(r_{h'}(s',a') - \hat{r}_{h'}(s',a'))
$$

$$
V^{\pi,h}_{M\cup r}(s) - V^{\hat{\pi},h}_{M\cup \hat{r}}(s) \le \sum_{h'=h}^{H}\sum_{s',a'} \eta^{h,h'}_{M,\pi}(s',a'|s)\,(r_{h'}(s',a') - \hat{r}_{h'}(s',a'))
$$

**Proof** Note that $Q^{\hat{\pi},h}_{M\cup \hat{r}}(s,a) \ge Q^{\pi,h}_{M\cup \hat{r}}(s,a)$ for all $s,a$ because $\hat{\pi}$ is optimal for $\hat{r}$. Hence

$$
\begin{aligned}
Q^{\pi,h}_{M\cup r}(s,a) - Q^{\hat{\pi},h}_{M\cup \hat{r}}(s,a) &\le Q^{\pi,h}_{M\cup r}(s,a) - Q^{\pi,h}_{M\cup \hat{r}}(s,a) \\
&\overset{(a)}{=} \sum_{h'=h}^{H}\sum_{s',a'} \eta^{h,h'}_{M,\pi}(s',a'|s,a)(r_{h'}(s',a') - \hat{r}_{h'}(s',a'))
\end{aligned}
$$

where (a) uses Lemma 2. After observing $V^{\hat{\pi},h}_{M\cup \hat{r}}(s) \ge V^{\pi,h}_{M\cup \hat{r}}(s)$, the second result follows analogously. ■

**Lemma 4** *Let $M$ be an MDP$\backslash$R, $r, \hat{r}$ two reward functions with optimal policies $\pi, \hat{\pi}$. Then,*

$$
Q^{\pi,h}_{M\cup r}(s,a) - Q^{\hat{\pi},h}_{M\cup r}(s,a) \le \sum_{h'=h}^{H}\sum_{s',a'} \eta^{h,h'}_{M,\pi}(s',a'|s,a) - \eta^{h,h'}_{M,\hat{\pi}}(s',a'|s,a) \quad (r_{h'}(s',a') - \hat{r}_{h'}(s',a'))
$$

**Proof**

$$Q^{;h}_{M[r}(s;a) \quad \hat{Q}^{;h}_{M[r}(s;a) = (Q^{;h}_{M[r}(s;a) \quad \hat{Q}^{;h}_{M[\hat{r}}(s;a)) + (\hat{Q}^{;h}_{M[\hat{r}}(s;a) \quad \hat{Q}^{;h}_{M[r}(s;a))$$

$$\overset{(a)}{} \sum_{h^0=h}^{H}\sum_{s^0;a^0} {}^{h;h^0}_{M;}(s^0;a^0js;a)\,(r_{h^0}(s^0;a^0) \quad \hat{r}_{h^0}(s^0;a^0)) + (\hat{Q}^{;h}_{M[\hat{r}}(s;a) \quad \hat{Q}^{;h}_{M[r}(s;a))$$

$$\overset{(b)}{=} \sum_{h^0=h}^{H}\sum_{s^0;a^0} {}^{h;h^0}_{M;}(s^0;a^0js;a)\,(r_{h^0}(s^0;a^0) \quad \hat{r}_{h^0}(s^0;a^0)) + \sum_{h^0=h}^{H}\sum_{s^0;a^0} {}^{h;h^0}_{M;\hat{}}(s^0;a^0js;a)\,(\hat{r}_{h^0}(s^0;a^0) \quad r_{h^0}(s^0;a^0))$$

$$= \sum_{h^0=h}^{H}\sum_{s^0;a^0} \quad {}^{h;h^0}_{M;}(s^0;a^0js;a) \quad {}^{h;h^0}_{M;\hat{}}(s^0;a^0js;a) \quad (r_{h^0}(s^0;a^0) \quad \hat{r}_{h^0}(s^0;a^0))$$

where (a) uses Lemma 3 and (b) uses Lemma 2. ■

**Lemma 5** *Let $M_1; M_2$ be two MDPnR with transition dynamics $P_1$, $P_2$ respectively, $r$ a reward function and a policy. Then, for any state $s$ and timestep $h$:*

$$V^{;h}_{M_2[r}(s) \quad V^{;h}_{M_1[r}(s) = \sum_{h^0=h}^{H}\sum_{s^0;a^0;s^{00}} {}^{h;h^0}_{M_2;}(s^0;s) \;{}_{h^0}(a^0js^0)(P_2(s^{00}js^0;a^0) \quad P_1(s^{00}js^0;a^0))V^{;h^0+1}_{M_1[r}(s^{00})$$

$$V^{;h}_{M_1[r}(s) \quad V^{;h}_{M_2[r}(s) = \sum_{h^0=h}^{H}\sum_{s^0;a^0;s^{00}} {}^{h;h^0}_{M_2;}(s^0;s) \;{}_{h^0}(a^0js^0)(P_1(s^{00}js^0;a^0) \quad P_2(s^{00}js^0;a^0))V^{;h^0+1}_{M_1[r}(s^{00})$$

*Moreover,*

$$V^{;h}_{M_2[r}(s) \quad V^{;h}_{M_1[r}(s) \quad \sum_{h^0=h}^{H}\sum_{s^0;a^0;s^{00}} {}^{h;h^0}_{M_2;}(s^0;s) \;{}_{h^0}(a^0js^0)\; P_2(s^{00}js^0;a^0) \quad P_1(s^{00}js^0;a^0)\; V^{;h^0+1}_{M_1[r}(s^{00})$$

**Proof** We start by writing explicitly the value-functions:

$$V^{;h}_{M_2[r}(s) \quad V^{;h}_{M_1[r}(s) = \sum_{a;s^0} {}_{h}(ajs) \quad P_2(s^0js;a)V^{;h+1}_{M_2[r}(s^0) \quad P_1(s^0js;a)V^{;h+1}_{M_1[r}(s^0) \quad P_2(s^0js;a)V^{;h+1}_{M_1[r}(s^0)$$

$$= \sum_{a;s^0} {}_{h}(ajs) \quad (P_2(s^0js;a) \quad P_1(s^0js;a))V^{;h+1}_{M_1[r}(s^0) + P_2(s^0js;a)(V^{;h+1}_{M_2[r}(s^0) \quad V^{;h+1}_{M_1[r}(s^0))$$

Unrolling the recursion gives the first result; the second result follows similarly:

$$V^{;h}_{M_1[r}(s) \quad V^{;h}_{M_2[r}(s) = \sum_{a;s^0} {}_{h}(ajs) \quad P_1(s^0js;a)V_{M}$$

$$= \sum_{a;s^0} {}_{h}(ajs) \quad (P_1(s^0js;a) \quad P_2(s^0js;a))V^{;h+1}_{M_1[r}(s^0) +$$

Together, the first two results imply the third one because all terms in the sums are non-negative. ■

**Lemma 6** *Let $\mathcal{M}_1, \mathcal{M}_2$ be two MDP\R with transition dynamics $P_1$, $P_2$ respectively, $r$ a reward function, and $\pi_1, \pi_2$ optimal policy in $\mathcal{M}_1 \cup r$ and $\mathcal{M}_2 \cup r$, respectively. Then, for any state $s$ and timestep $h$:*

$$V^{\pi_1,h}_{\mathcal{M}_1\cup r}(s) - V^{\pi_2,h}_{\mathcal{M}_2\cup r}(s) \le \sum_{h'=h}\sum_{s',a',s''} \eta^{h,h'}_{\mathcal{M}_2,\pi_1}(s';s)\,\pi_{1,h}(a'|s')(P_1(s''|s',a') - P_2(s''|s',a'))V^{\pi_1,h}_{\mathcal{M}_1\cup r}(s'')$$

$$V^{\pi_2,h}_{\mathcal{M}_2\cup r}(s) - V^{\pi_1,h}_{\mathcal{M}_1\cup r}(s) \le \sum_{h'=h}\sum_{s',a',s''} \eta^{h,h'}_{\mathcal{M}_2,\pi_2}(s';s)\,\pi_{2,h}(a'|s')(P_2(s''|s',a') - P_1(s''|s',a'))V^{\pi_2,h}_{\mathcal{M}_2\cup r}(s'')$$

**Proof**

$$\begin{aligned}
V^{\pi_1,h}_{\mathcal{M}_1\cup r}(s) - V^{\pi_2,h}_{\mathcal{M}_2\cup r}(s) &= \sum_{a,s'} \pi_{1,h}(a|s)P_1(s'|s,a)V^{\pi_1,h+1}_{\mathcal{M}_1\cup r}(s') - \pi_{2,h}(a|s)P_2(s'|s,a)V^{\pi_2,h+1}_{\mathcal{M}_2\cup r}(s') \\
&\qquad \pm \pi_{1,h}(a|s)P_2(s'|s,a)V^{\pi_1,h+1}_{\mathcal{M}_1\cup r}(s') \pm \pi_{1,h}(a|s)P_2(s'|s,a)V^{\pi_2,h+1}_{\mathcal{M}_2\cup r}(s') \\
&= \sum_{a,s'} \pi_{1,h}(a|s)P_2(s'|s,a)(V^{\pi_1,h+1}_{\mathcal{M}_1\cup r}(s') - V^{\pi_2,h+1}_{\mathcal{M}_2\cup r}(s')) \\
&\qquad + \pi_{1,h}(a|s)(P_1(s'|s,a) - P_2(s'|s,a))V^{\pi_1,h+1}_{\mathcal{M}_1\cup r}(s') \\
&\qquad + (\pi_{1,h}(a|s) - \pi_{2,h}(a|s))P_2(s'|s,a)V^{\pi_2,h+1}_{\mathcal{M}_2\cup r}(s') \\
&\le \sum_{a,s'} \pi_{1,h}(a|s)P_2(s'|s,a)(V^{\pi_1,h+1}_{\mathcal{M}_1\cup r}(s') - V^{\pi_2,h+1}_{\mathcal{M}_2\cup r}(s')) \\
&\qquad + \pi_{1,h}(a|s)(P_1(s'|s,a) - P_2(s'|s,a))V^{\pi_1,h+1}_{\mathcal{M}_1\cup r}(s')
\end{aligned}$$

where the last inequality uses that $\pi_2$ is optimal for $\mathcal{M}_2 \cup r$. Unrolling the recursion gives the first result. A similar argument yields the second results:

$$\begin{aligned}
V^{\pi_2,h}_{\mathcal{M}_2\cup r}(s) - V^{\pi_1,h}_{\mathcal{M}_1\cup r}(s) &= \sum_{a,s'} \pi_{2,h}(a|s)P_2(s'|s,a)V^{\pi_2,h+1}_{\mathcal{M}_2\cup r}(s') - \pi_{1,h}(a|s)P_1(s'|s,a)V^{\pi_1,h+1}_{\mathcal{M}_1\cup r}(s') \\
&\qquad \pm \pi_{2,h}(a|s)P_2(s'|s,a)V^{\pi_1,h+1}_{\mathcal{M}_1\cup r}(s') \\
&= \sum_{a,s'} \pi_{2,h}(a|s)P_2(s'|s,a)(V^{\pi_2,h+1}_{\mathcal{M}_2\cup r}(s') - V^{\pi_1,h+1}_{\mathcal{M}_1\cup r}(s')) \\
&\qquad + \pi_{2,h}(a|s)P_2(s'|s,a)V^{\pi_1,h+1}_{\mathcal{M}_1\cup r}(s') - \pi_{1,h}(a|s)P_1(s'|s,a)V^{\pi_1,h+1}_{\mathcal{M}_1\cup r}(s') \\
&\le \sum_{a,s'} \pi_{2,h}(a|s)P_2(s'|s,a)(V^{\pi_2,h+1}_{\mathcal{M}_2\cup r}(s') - V^{\pi_1,h+1}_{\mathcal{M}_1\cup r}(s')) \\
&\qquad + \pi_{2,h}(a|s)(P_2(s'|s,a) - P_1(s'|s,a))V^{\pi_1,h+1}_{\mathcal{M}_1\cup r}(s')
\end{aligned}$$

■

## B.2 Feasible Reward Set

In this section, we characterize the feasible reward set first implicitly, then explicitly, and prove a result about error propagation. Metelli et al. (2021) provide a similar analysis in the infinite horizon setting.

**Lemma 3 (Feasible Reward Set Implicit)** *A reward function $r$ is feasible if and only if for all $s, a, h$ it holds that: $A^{\pi^E,h}_{\mathcal{M}\cup r}(s,a) = 0$ if $\pi^E_h(a|s) > 0$ and $A^{\pi^E,h}_{\mathcal{M}\cup r}(s,a) \le 0$ if $\pi^E_h(a|s) = 0$. Moreover, if the second inequality is strict, $\pi^E$ is uniquely optimal, i.e., $\Pi^*_{\mathcal{M}\cup r} = \{\pi^E\}$.*

**Proof** The result follows directly from Definition 1. ■

**Lemma 7** *A Q-function satisfies the conditions of Lemma 3 if and only if there exists an* $\{A_h \in \mathbb{R}^{S\times A}_{\geq 0}\}_{h\in H}$ *and* $\{V_h \in \mathbb{R}^S\}$ *such that for every* $h, s, a \in [H]\times S\times A$*:*

$$Q^{\pi^E,h}_{\mathcal{M}\cup r}(s,a) = -A_h(s,a)\mathbb{1}_{\{\pi^E_h(a|s)=0\}} + V_h(s)$$

**Proof** We first show that if $Q^{\pi^E,h}_{\mathcal{M}\cup r}(s,a)$ has this form, the conditions of Lemma 3 are satisfied, and then the converse. Assume $Q^{\pi^E,h}_{\mathcal{M}\cup r}(s,a) = -A_h(s,a)\mathbb{1}_{\{\pi^E_h(a|s)=0\}} + V_h(s)$. Then,

$$V^{\pi^E,h}_{\mathcal{M}\cup r}(s) = \sum_a \pi^E_h(a|s) Q^{\pi^E,h}_{\mathcal{M}\cup r}(s,a) = V_h(s).$$

If $\pi^E_h(a|s) > 0$, then $Q^{\pi^E,h}_{\mathcal{M}\cup r}(s,a) = V^{\pi^E,h}_{\mathcal{M}\cup r}(s)$, which is the first condition of Lemma 3. If $\pi^E_h(a|s) = 0$, $Q^{\pi^E,h}_{\mathcal{M}\cup r}(s,a) = V^{\pi^E,h}_{\mathcal{M}\cup r}(s) - A_h(s,a) \leq V^{\pi^E,h}_{\mathcal{M}\cup r}(s)$, which is the second condition of Lemma 3.

For the converse, assume that the conditions of Lemma 3 hold, and let $V_h(s) = V^{\pi^E,h}_{\mathcal{M}\cup r}(s)$ and $A_h(s,a) = V^{\pi^E,h}_{\mathcal{M}\cup r}(s) - Q^{\pi^E,h}_{\mathcal{M}\cup r}(s,a)$. ■

**Lemma 4 (Feasible Reward Set Explicit)** *A reward function* $r$ *is feasible if and only if there exists an* $\{A_h \in \mathbb{R}^{S\times A}_{\geq 0}\}_{h\in[H]}$ *and* $\{V_h \in \mathbb{R}^S\}_{h\in[H]}$ *such that for all* $s, a, h$ *it holds that:*

$$r_h(s,a) = -A_h(s,a)\mathbb{1}_{\{\pi^E_h(a|s)=0\}} + V_h(s) - \sum_{s'} P(s'|s,a)V_{h+1}(s')$$

**Proof** Since $Q^{\pi^E,h}_{\mathcal{M}\cup r}(s,a) = r_h(s,a) + \sum_{s'} P(s'|s,a)V_{h+1}(s')$, using Lemma 7, we have:

$$\begin{aligned} r_h(s,a) &= Q^{\pi^E,h}_{\mathcal{M}\cup r}(s,a) - \sum_{s'} P(s'|s,a)V_{h+1}(s') \\ &= -A_h(s,a)\mathbb{1}_{\{\pi^E_h(a|s)=0\}} + V_h(s) - \sum_{s'} P(s'|s,a)V_{h+1}(s') \end{aligned}$$

■

**Theorem 5 (Error Propagation)** *Let* $(\mathcal{M},\pi^E)$ *and* $(\widehat{\mathcal{M}},\widehat{\pi}^E)$ *be two IRL problems. Then, for any* $r \in \mathcal{R}_{(\mathcal{M},\pi^E)}$ *there exists* $\widehat{r} \in \widehat{\mathcal{R}}_{(\widehat{\mathcal{M}},\widehat{\pi}^E)}$ *such that:*

$$|r_h(s,a) - \widehat{r}_h(s,a)| \leq A_h(s,a)|\pi^E_h(a|s) - \widehat{\pi}^E_h(a|s)| + \sum_{s'} V_{h+1}(s')|P(s'|s,a) - \widehat{P}(s'|s,a)|$$

*and we can bound* $V_h \leq (H-h)R_{\max}$ *and* $A_h \leq (H-h)R_{\max}$.

**Proof** We start by rewriting $r$ and $\hat{r}$ using Lemma 4:

$$\begin{aligned} r_h(s,a) &= -A_h(s,a)\mathbb{1}_{\{\pi^E_h(a|s)=0\}} + V_h(s) - \sum_{s'} P(s'|s,a)V_{h+1}(s') \\ \hat{r}_h(s,a) &= -\hat{A}_h(s,a)\mathbb{1}_{\{\hat{\pi}^E_h(a|s)=0\}} + \hat{V}_h(s) - \sum_{s'} \widehat{P}(s'|s,a)\hat{V}_{h+1}(s') \end{aligned}$$

**Algorithm 2** Uniform sampling IRL with a generative model.

```
Input: significance $\delta \in (0, 1)$, target accuracy $\epsilon$, maximum number of samples per iter. $n_{\max}$
Initialize $k \leftarrow 0$, $\epsilon_0 \leftarrow H$
while $\epsilon_k > \epsilon/2$ do
    Uniformly sample $\lceil \frac{n_{\max}}{SAH} \rceil$ samples from all $(s, a, h) \in \mathcal{S} \times \mathcal{A} \times [H]$
    For all samples, observe sample from transition dynamics and expert policy
    $k \leftarrow k + 1$
    Update $\widehat{P}_k$, $\hat{\pi}_k$, and $C_k^h$
    Update accuracy $\epsilon_k \leftarrow H \max_{s,a,h} C_k^h(s, a)$
end while
```

We can choose (w.l.o.g.) $V_h = \hat{V}_h$ and $\hat{A}_h = 1_{\{\pi_h^E(a|s)=0\}} A_h$:

$$\begin{aligned}
r_h(s,a) - \hat{r}_h(s,a) = & -A_h(s,a) 1_{\{\pi_h^E(a|s)=0\}} + V_h(s) + \sum_{s'} P(s'|s,a) V_{h+1}(s') \\
& + A_h(s,a) 1_{\{\hat{\pi}_h^E(a|s)=0\}} 1_{\{\pi_h^E(a|s)=0\}} - V_h(s) - \sum_{s'} \widehat{P}(s'|s,a) V_{h+1}(s') \\
= & A_h(s,a) 1_{\{\pi_h^E(a|s)=0\}} (1_{\{\hat{\pi}_h^E(a|s)=0\}} - 1) + \sum_{s'} V_{h+1}(s')(P(s'|s,a) - \widehat{P}(s'|s,a)) \\
= & -A_h(s,a) 1_{\{\pi_h^E(a|s)=0\}} 1_{\{\hat{\pi}_h(a|s) > 0\}} + \sum_{s'} V_{h+1}(s')(P(s'|s,a) - \widehat{P}(s'|s,a))
\end{aligned}$$

The result follows by taking the absolute value and applying the triangle inequality. ■

### B.3 Uniform Sampling IRL with a Generative Model

In this section, we derive sample complexity results for uniform sampling with a generative model (Algorithm 2). Metelli et al. (2021) proved an analogous result for the infinite horizon setting focusing on transferable rewards. In contrast, our focus is on the finite horizon setting. Moreover, Metelli et al. (2021) considers to learn a reward that is transferable to a known target environment. In our setting, instead, we suppose to use the recovered reward function in the unknown source environment.

**Definition 2 (Optimality Criterion)** *Let $\mathfrak{S}$ be a sampling strategy. Let $\mathcal{R}_{\mathfrak{B}}$ be the exact feasible set and $\mathcal{R}_{\hat{\mathfrak{B}}}$ be the feasible set recovered after observing $n \geq 0$ samples collected from $\mathcal{M}$ and $\pi^E$. We say that $\mathfrak{S}$ is $(\epsilon, \delta, n)$*-correct *if with probability at least $1 - \delta$ it holds that:*

$$\inf_{\hat{r} \in \mathcal{R}_{\hat{\mathfrak{B}}}} \sup_{\hat{\pi}^* \in \Pi^*_{\widehat{\mathcal{M}} \cup \hat{r}}} \max_{s,a,h} \left| Q^{\pi^*,h}_{\mathcal{M} \cup r}(s,a) - Q^{\hat{\pi}^*,h}_{\mathcal{M} \cup r}(s,a) \right| \leq \epsilon \qquad \text{for each } r \in \mathcal{R}_{\mathfrak{B}};$$

$$\inf_{r \in \mathcal{R}_{\mathfrak{B}}} \sup_{\pi^* \in \Pi^*_{\mathcal{M} \cup r}} \max_{s,a,h} \left| Q^{\pi^*,h}_{\mathcal{M} \cup r}(s,a) - Q^{\hat{\pi}^*,h}_{\mathcal{M} \cup r}(s,a) \right| \leq \epsilon \qquad \text{for each } \hat{r} \in \mathcal{R}_{\hat{\mathfrak{B}}};$$

*where $\pi^*$ is an optimal policy in $\mathcal{M} \cup r$ and $\hat{\pi}^*$ is an optimal policy in $\widehat{\mathcal{M}} \cup \hat{r}$.*

**Lemma 8 (Good Event)** *Let $\pi^E$ be a (possibly stochastic) expert policy. We estimate the expert policy with $\hat{\pi}^E$ and the transition model $P$ with an estimate $\widehat{P}_k$ from $k$ episodic interactions. Let $n_k^h(s,a)$ and $n_k^h(s)$ be the number of times state action pairs and states have been observed at time $h$ within the first $k$ episodes, and $n_k^{h+}(s,a) =$*

$\max\{1, n_k^h(s,a)\}$. *Then,*

$$\mathbb{1}_{\{\pi_h^E(a|s)=0\}}\mathbb{1}_{\{\hat\pi_h^E(a|s)>0\}}A_h(s,a) \le (H-h)R_{\max}\sqrt{\frac{\ell_k^h(s,a)}{n_k^{h+}(s,a)}}$$

$$\mathbb{1}_{\{\hat\pi_h^E(a|s)=0\}}\mathbb{1}_{\{\pi_h^E(a|s)>0\}}\hat A_h(s,a) \le (H-h)R_{\max}\sqrt{\frac{\ell_k^h(s,a)}{n_k^{h+}(s,a)}}$$

$$\sum_{s'}|(P(s'|s,a)-\hat P_k(s'|s,a))V_r^{\pi,h}(s')| \le (H-h)R_{\max}\sqrt{\frac{2\ell_k^h(s,a)}{n_k^{h+}(s,a)}}$$

$$\sum_{s'}|(P(s'|s,a)-\hat P_k(s'|s,a))\hat V_r^{\pi,h}(s')| \le (H-h)R_{\max}\sqrt{\frac{2\ell_k^h(s,a)}{n_k^{h+}(s,a)}}$$

*where* $\ell_k^h(s,a)=\log\left(24SAH(n_k^{h+}(s,a))^2/\delta\right)$*, holds simultaneously for all* $(s,a,h)\in S\times A\times[H]$ *and* $k\ge 1$ *with probability at least* $1-\delta$. *We call the event that these equations hold the* good event $E$ *and write* $P(E)\ge 1-\delta$.

**Proof** We show that each statement individually does not hold with probability less than $\delta/4$, which implies the result via a union bound. Let us denote $\beta_1(s,a,h) := (H-h)R_{\max}\sqrt{\frac{2\ell_k^h(s,a)}{n_k^{h+}(s,a)}}$. First, consider the last two inequalities. The probability that either of them does not hold is:

$$\begin{aligned}
&\Pr\left(\exists k\ge 1, (s,a,h)\in S\times A\times[H]: \sum_{s'}|(P(s'|s,a)-\hat P_k(s'|s,a))V_r^{\pi,h}(s')| > \beta_1(s,a,h)\right)\\
\overset{(a)}{\le}\ &\Pr\left(\exists m\ge 0, (s,a,h)\in S\times A\times[H]: \sum_{s'}|(P(s'|s,a)-\hat P_k(s'|s,a))V_r^{\pi,h}(s')| > \beta_1(s,a,h)\right)\\
\overset{(b)}{\le}\ &\sum_{m\ge 0}\sum_{s,a}\sum_{h=0}^{H}\Pr\left(\sum_{s'}|(P(s'|s,a)-\hat P_k(s'|s,a))V_r^{\pi,h}(s')| > \beta_1(s,a,h)\right)\\
\overset{(c)}{\le}\ &\sum_{m\ge 0}\sum_{s,a}\sum_{h=0}^{H} 2\exp\left(-\frac{2\beta_1(s,a,h)^2m^2}{4m(H-h)^2R_{\max}^2}\right) \le \sum_{m\ge 0}\sum_{s,a}\sum_{h=0}^{H} 2\exp(-\ell_k(s,a))\\
=\ &\sum_{m\ge 0}\sum_{s,a}\sum_{h=0}^{H}\frac{2\delta}{24SAH(m^+)^2} = \frac{\delta}{12}\left(1+\sum_{m>0}\frac{1}{m^2}\right) = \frac{\delta}{12}\left(1+\frac{\pi^2}{6}\right) \le \frac{\delta}{4}
\end{aligned}$$

Step (a) assumes that we visit a state action pair $m$ times, and focuses on these $m$ times the transition model for the given state-action pair is updated. Step (b) uses a union bound over $m$ and $(s,a)$. Step (c) applies Hoeffding's inequality using that we estimate $P$ with an average of samples, and $V_r^{\pi,h}\le(H-h)R_{\max}$.

We show the first two inequalities similarly, with $\beta_2(s,a,h) := (H-h)R_{\max}\sqrt{\frac{\ell_k^h(s,a)}{n_k^{h+}(s,a)}}$

$$
\begin{aligned}
&\Pr\left(\exists k \ge 1, (s,a,h) \in \mathcal{S}\times\mathcal{A}\times[H] : |(\pi_k^E(a|s) - \hat\pi_k^E(a|s))V_r^{\pi,h}(s')| > \beta_2(s,a,h)\right)\\
&\overset{(a)}{\le} \Pr\left(\exists m \ge 0, (s,a,h) \in \mathcal{S}\times\mathcal{A}\times[H] : |(\pi_k^E(a|s) - \hat\pi_k^E(a|s))V_r^{\pi,h}(s')| > \beta_2(s,a,h)\right)\\
&\overset{(b)}{\le} \sum_{m\ge 0}\sum_{s,a}\sum_{h=0}^{H} \Pr\left(|(\pi_k^E(a|s) - \hat\pi_k^E(a|s))V_r^{\pi,h}(s')| > \beta_2(s,a,h)\right)\\
&\overset{(c)}{\le} \sum_{m\ge 0}\sum_{s,a}\sum_{h=0}^{H} 2\exp\left(-\frac{2\beta_2(s,a,h)^2 m^2}{m(H-h)^2R_{\max}^2}\right) \le \sum_{m\ge 0}\sum_{s,a}\sum_{h=0}^{H} 2\exp\left(-\ell_k(s,a)\right)\\
&= \sum_{m\ge 0}\sum_{s,a}\sum_{h=0}^{H} \frac{2\delta}{24SAH(m^+)^2} = \frac{\delta}{12}\left(1 + \sum_{m\ge 0}\frac{1}{m^2}\right) = \frac{\delta}{12}\left(1+\frac{\pi^2}{6}\right) \le \frac{\delta}{4}
\end{aligned}
$$

A union bound over all equations results in $P(E) \ge 1-\delta$. ■

**Definition 9** *We define the reward uncertainty as*

$$C_k^h(s,a) = (H-h)R_{\max}\min\left(1, 2\sqrt{\frac{2\ell_k^h(s,a)}{n_k^h(s,a)}}\right)$$

**Corollary 10** *Under the good event $E$, in each iteration $k$ it holds for all $(s,a,h) \in \mathcal{S}\times\mathcal{A}\times[H]$ that:*

$$|r_h(s,a) - \hat r_h^k(s,a)| \le C_k^h(s,a)$$

**Proof**

$$
\begin{aligned}
|r_h(s,a) - \hat r_h^k(s,a)| &\overset{(a)}{\le} A_h(s,a)\mathbb{1}_{\{\pi_h^E(a|s)=0\}}\mathbb{1}_{\{\hat\pi_h^E(a|s)>0\}} + \sum_{s'} V_{h+1}(s')|P(s'|s,a) - \hat P(s'|s,a)|\\
&\overset{(b)}{\le} (H-h)R_{\max}\left(2\sqrt{\frac{2\ell_k^h(s,a)}{n_k^{h^+}(s,a)}}\right) = C_k^h(s,a)
\end{aligned}
$$

where (a) uses Theorem 5 and (b) uses Lemma 8. ■

**Corollary 11** *Let $\mathfrak{S}$ be a sampling strategy. Let $\mathcal{R}_{\mathfrak{B}}$ be the exact feasible set and $\mathcal{R}_{\hat{\mathfrak{B}}_k}$ be the feasible set recovered after $k$ iterations. If*

$$H\max_{s,a,h} C_k^h(s,a) \le \frac{\epsilon}{2},$$

*then the conditions of Definition 2 are satisfied.*

**Proof** For the first condition of Definition 2, observe:

$$
\begin{aligned}
&\inf_{\hat r\in\mathcal{R}_{\hat{\mathfrak{B}}_k}}\sup_{\pi\in\Pi^*_{\mathcal{M}\cup r}}\max_{s,a,h}\left(Q^{\pi,h}_{\mathcal{M}\cup r}(s,a) - \hat Q^{\pi,h}_{\mathcal{M}\cup r}(s,a)\right)\\
&\overset{(a)}{\le} \inf_{\hat r\in\mathcal{R}_{\hat{\mathfrak{B}}_k}}\sup_{\pi\in\Pi^*_{\mathcal{M}\cup r}}\max_{s,a,h}\sum_{h'=h}^{H}\sum_{s',a'} \eta^{h,h'}_{\mathcal{M},\pi}(s',a'|s,a) \quad \eta^{h,h'}_{\mathcal{M},\hat\pi}(s',a'|s,a)\left(r_{h'}(s',a') - \hat r_{h'}(s',a')\right)\\
&\overset{(b)}{\le} \inf_{\hat r\in\mathcal{R}_{\hat{\mathfrak{B}}_k}}\sup_{\pi\in\Pi^*_{\mathcal{M}\cup r}}\max_{s,a,h}\sum_{h'=h}^{H}\sum_{s',a'} \eta^{h,h'}_{\mathcal{M},\pi}(s',a'|s,a) \quad \eta^{h,h'}_{\mathcal{M},\hat\pi}(s',a'|s,a)\, C_k^{h'}(s',a')\\
&\le 2H\max_{s,a,h} C_k^h(s,a)
\end{aligned}
$$

where (a) uses **??** and (b) uses Corollary 10.

For the second condition of Definition 2, it follows similarly that:

$$\inf_{r 2 R_{\mathrm{B}}} \sup_{2_{\quad M [ r}} \max_{s;a;h}(Q_{M [ r}^{;h}(s;a) \qquad \hat{Q}_{M [ r}^{;h}(s;a)) \qquad 2H \max_{s;a;h} C_k^h(s;a)$$

Hence, if $H \max_{s;a;h} C_k^h(s;a) \qquad =2$, both conditions of Definition 2 are satisfied. ■

**Theorem 12 (Sample Complexity of Uniform Sampling IRL)** *With probability at least* $1$ *, Algorithm 2 stops at iteration fulfilling Definition 2 with a number of samples upper bounded by:*

$$n \quad O \quad \frac{H^5 R_{\max}^2 SA}{2}$$

**Proof** First, note

$$H \max_{s;a;h} C_k^h(s;a) = H^2 R_{\max} \max_{s;a;h} \quad 2 \quad \mathrm{s}\overline{\frac{2`_k^h(s;a)}{n_k^{h^+}(s;a)}}\,!$$

After iterations, we have collected $n_{\max}$ samples and for each $s;a;h$, we have: $n^{h^+}(s;a) \qquad \frac{n_{\max}}{SAH} \qquad 1$

To terminate at iteration , we need to have for all $s;a;h$:

$$2H^2 R_{\max} \quad \mathrm{s}\overline{\frac{2`^h(s;a)}{n^h(s;a)}} \qquad \overline{2}$$

which implies

$$n^h(s;a) \qquad \frac{32 H^4 R_{\max}^2 `^h(s;a)}{2}$$

By using Lemma B.8 by Metelli et al. (2021), we can conclude that the number of samples necessary to ensure accuracy " is:

$$n \quad O \quad \frac{H^5 R_{\max}^2 SA}{2}$$

■

**Corollary 13** *If the true reward function does not depend on the timestep* $h$*, i.e.,* $r_h(s;a) = r(s;a)$*, then we can modify Algorithm 2 to only need* $n \quad O \quad \frac{H^4 R_{\max}^2 SA}{2}$ *samples.*

**Proof** If we know that the reward function does not depend on $h$ we can choose $C_k(s;a) = \min_h C_k^h(s;a)$ as a confidence interval of the reward. Consequently, we can sample all states for a fixed $h$.

We still need for all $s;a$:

$$2H^2 R_{\max} \quad \mathrm{s}\overline{\frac{2`^h(s;a)}{n^h(s;a)}} \qquad \overline{2} \quad ) \quad n^h(s;a) \qquad \frac{32 H^4 R_{\max}^2 `^h(s;a)}{2}$$

Again, we use Lemma B.8 by Metelli et al. (2021), but we can eliminate one sum over $H$, ending up with:

$$n \quad O \quad \frac{H^4 R_{\max}^2 SA}{2}$$

■

### B.4 Sample Complexity of AceIRL in Unknown Environments (Problem Independent)

We are now ready to analyze the sample complexity of AceIRL (Algorithm 1). We first consider the simple version of the algorithm: AceIRL Greedy. Then, we consider the full version of the algorithm after introducing a few additional lemma about the policy confidence set. We start by defining the error upper bound and deriving two lemmas that will help us to show that it is indeed an upper bound on the error we want to reduce.

**Definition 14** *We define recursively:*

$$E_k^H(s,a) = 0, \quad E_k^h(s,a) = \min\Big((H-h)R_{\max}, C_k^h(s,a) + \sum_{s'} \hat{P}(s'|s,a) \max_{a' \in A} E_k^{h+1}(s',a')\Big)$$

*where $\hat{P}$ is the estimated transition model of the environment.*

The first lemma shows that the error upper bound can upper bound the error due to estimating the transition model.

**Lemma 15** *Under the good event $\mathcal{E}$, for all policies $\pi$ and reward functions $r$ and all $s, a, h$:*

$$|Q^{\pi,h}_{\hat{M}\cup r}(s,a) - Q^{\pi,h}_{M\cup r}(s,a)| \le E_k^h(s,a)$$

**Proof**

$$\begin{aligned}
|Q^{\pi,h}_{\hat{M}\cup r}(s,a) - Q^{\pi,h}_{M\cup r}(s,a)| &= \Big|\sum_{s'} \hat{P}(s'|s,a) \sum_{a'} \pi(a'|s') Q^{\pi,h+1}_{\hat{M}\cup r}(s',a') \\
&\quad - \sum_{s'} P(s'|s,a) \sum_{a'} \pi(a'|s') Q^{\pi,h+1}_{M\cup r}(s',a') \pm \sum_{s'} \hat{P}(s'|s,a) \sum_{a'} \pi(a'|s') Q^{\pi,h+1}_{M\cup r}(s',a')\Big| \\
&\le \Big|\sum_{s'} \big(\hat{P}(s'|s,a) - P(s'|s,a)\big) \sum_{a'} \pi(a'|s') Q^{\pi,h+1}_{M\cup r}(s',a')\Big| \\
&\quad + \sum_{s'} \hat{P}(s'|s,a) \sum_{a'} \pi(a'|s') \Big(Q^{\pi,h+1}_{\hat{M}\cup r}(s,a) - Q^{\pi,h+1}_{M\cup r}(s,a)\Big) \\
&\le C_k^h(s,a) + \sum_{s'} \hat{P}(s'|s,a) \sum_{a'} \pi(a'|s') \Big|Q^{\pi,h+1}_{\hat{M}\cup r}(s,a) - Q^{\pi,h+1}_{M\cup r}(s,a)\Big|
\end{aligned}$$

For $h = H$ the result holds trivially. Now assuming it holds for $h+1$, we consider step $h$:

$$\begin{aligned}
|Q^{\pi,h}_{\hat{M}\cup r}(s,a) - Q^{\pi,h}_{M\cup r}(s,a)| &\le C_k^h(s,a) + \sum_{s'} \hat{P}(s'|s,a) \sum_{a'} \pi(a'|s') \Big|Q^{\pi,h+1}_{\hat{M}\cup r}(s,a) - Q^{\pi,h+1}_{M\cup r}(s,a)\Big| \\
&\le C_k^h(s,a) + \sum_{s'} \hat{P}(s'|s,a) \max_{a'} \Big|Q^{\pi,h+1}_{\hat{M}\cup r}(s,a) - Q^{\pi,h+1}_{M\cup r}(s,a)\Big| \\
&\le C_k^h(s,a) + \sum_{s'} \hat{P}(s'|s,a) \max_{a'} E_k^{h+1}(s',a') = E_k^h(s,a)
\end{aligned}$$

■

The next lemma shows that the error upper bound can also upper bound the error in estimating the reward function, which is due to estimating the transition model and the expert policy.

**Lemma 16** *Under the good event $\mathcal{E}$, for all reward function $r$, all policies $\pi$, and all $s, a \in S \times A$:*

$$|Q^{\pi,h}_{\hat{M}\cup \hat{r}}(s,a) - Q^{\pi,h}_{\hat{M}\cup r}(s,a)| \le E_k^h(s,a)$$

**Proof** For $h = H$ the result holds trivially. Now assuming it holds for $h + 1$, we consider step $h$:

$$\begin{aligned}
&|Q^{\pi,h}_{\hat{M}\cup\hat{r}}(s,a) - Q^{\pi,h}_{\hat{M}\cup r}(s,a)| \\
&\leq |\hat{r}(s,a) - r(s,a)| + \sum_{s'} \hat{P}(s'|s,a) \sum_{a'} \pi(a'|s')|Q^{\pi,h+1}_{\hat{M}\cup\hat{r}}(s',a') - Q^{\pi,h+1}_{\hat{M}\cup r}(s',a')| \\
&\leq |\hat{r}(s,a) - r(s,a)| + \sum_{s'} \hat{P}(s'|s,a) \max_{a'} |Q^{\pi,h+1}_{\hat{M}\cup\hat{r}}(s',a') - Q^{\pi,h+1}_{\hat{M}\cup r}(s',a')| \\
&\leq |\hat{r}(s,a) - r(s,a)| + \sum_{s'} \hat{P}(s'|s,a) \max_{a'} E_k^{h+1}(s',a') = E_k^h(s,a)
\end{aligned}$$

■

We can now combine the previous two lemmas to show that $E$ is indeed an upper bound on the error we want to reduce. This implies correctness of AceIRL Greedy, which the following lemma formalizes.

**Lemma 17 (Correctness of AceIRL Greedy)** *If AceIRL Greedy stops in episode $k$, after sampling $n$ samples, i.e., $E_k^0(s_0, \pi_{k+1}(s_0)) \leq \frac{\epsilon}{4}$, then it fulfills Definition 2.*

**Proof** Let us define the error

$$e_k^h(s,a) := |Q^{\pi^*,h}_{M\cup r}(s,a) - Q^{\hat{\pi}^*,h}_{M\cup r}(s,a)|$$

where $\pi^*$ is the true optimal policy in $M \cup r$, and $\hat{\pi}^*$ is the optimal policy in $\hat{M} \cup \hat{r}$, i.e., in the estimated MDP using the inferred reward function. Then,

$$\begin{aligned}
e_k^h(s,a) &= |Q^{\pi^*,h}_{M\cup r}(s,a) - Q^{\hat{\pi}^*,h}_{M\cup r}(s,a) \pm Q^{\pi^*,h}_{\hat{M}\cup r}(s,a) \pm Q^{\hat{\pi}^*,h}_{\hat{M}\cup r}(s,a)| \\
&\leq \underbrace{|Q^{\pi^*,h}_{M\cup r}(s,a) - Q^{\pi^*,h}_{\hat{M}\cup r}(s,a)|}_{E_k^h(s,a)} + |Q^{\pi^*,h}_{\hat{M}\cup r}(s,a) - Q^{\hat{\pi}^*,h}_{\hat{M}\cup r}(s,a)| + \underbrace{|Q^{\hat{\pi}^*,h}_{\hat{M}\cup r}(s,a) - Q^{\hat{\pi}^*,h}_{M\cup r}(s,a)|}_{E_k^h(s,a)} \\
&\leq 2E_k^h(s,a) + |Q^{\pi^*,h}_{\hat{M}\cup r}(s,a) - Q^{\hat{\pi}^*,h}_{\hat{M}\cup r}(s,a)|
\end{aligned}$$

where, we used Lemma 15.

Let us consider the remaining term $|Q^{\pi^*,h}_{\hat{M}\cup r}(s,a) - Q^{\hat{\pi}^*,h}_{\hat{M}\cup r}(s,a)|$ in two steps. First, we have:

$$\begin{aligned}
Q^{\pi^*,h}_{\hat{M}\cup r}(s,a) - Q^{\hat{\pi}^*,h}_{\hat{M}\cup r}(s,a) \leq& \underbrace{Q^{\pi^*,h}_{\hat{M}\cup r}(s,a) - Q^{\pi^*,h}_{\hat{M}\cup\hat{r}}(s,a)}_{E_k^h(s,a)} + \underbrace{Q^{\pi^*,h}_{\hat{M}\cup\hat{r}}(s,a) - Q^{\hat{\pi}^*,h}_{\hat{M}\cup\hat{r}}(s,a)}_{\leq 0} + \\
&+ \underbrace{Q^{\hat{\pi}^*,h}_{\hat{M}\cup\hat{r}}(s,a) - Q^{\hat{\pi}^*,h}_{\hat{M}\cup r}(s,a)}_{E_k^h(s,a)} \leq 2E_k^h(s,a);
\end{aligned}$$

where we used Lemma 16 and the fact that $\hat{\pi}^*$ is optimal in the MDP $\hat{M} \cup \hat{r}$. Second, we have:

$$\begin{aligned}
Q^{\hat{\pi}^*,h}_{\hat{M}\cup r}(s,a) - Q^{\pi^*,h}_{\hat{M}\cup r}(s,a) \leq& \underbrace{Q^{\hat{\pi}^*,h}_{\hat{M}\cup r}(s,a) - Q^{\hat{\pi}^*,h}_{M\cup r}(s,a)}_{E_k^h(s,a)} + \underbrace{Q^{\hat{\pi}^*,h}_{M\cup r}(s,a) - Q^{\pi^*,h}_{M\cup r}(s,a)}_{\leq 0} + \\
&+ \underbrace{Q^{\pi^*,h}_{M\cup r}(s,a) - Q^{\pi^*,h}_{\hat{M}\cup r}(s,a)}_{E_k^h(s,a)} \leq 2E_k^h(s,a);
\end{aligned}$$

where we used Lemma 15 and the fact that $\pi^*$ is optimal in the MDP $M \cup r$. Overall, we find that

$$|Q^{\pi^*,h}_{\hat{M}\cup r}(s,a) - Q^{\hat{\pi}^*,h}_{\hat{M}\cup r}(s,a)| \leq 2E_k^h(s,a);$$

and consequently,

$$e_k^h(s;a) \quad 4E_k^h(s;a):$$

Note that, $E_k^h(s;a)$ only sums positive terms, hence:

$$\max_{s;a;h} E_k^h(s;a) \quad \max_a E_k^0(s_0;a) = E_k^0(s_0; \quad {}_{k+1}(s_0))$$

Hence, if $E_k^0(s_0; \quad {}_{k+1}(s_0)) \quad \overline{4}$, we have for all $s;a;h \; 2 \; S \quad A \quad [H]$:

$$e_k^h(s;a)$$

which implies correctness according to Definition 2. ■

Next, we will analyze the sample complexity of AceIRL Greedy. Let us first define pseudo-counts that will be crucial to deal with the uncertainty of the transition dynamics in our analysis. This is similar to the analysis of UCRL for reward-free exploration by Kaufmann et al. (2021).

**Definition 18** *We define the* pseudo-counts *of visiting a specific state action pair at timestep $h$ within the first $k$ iterations as*

$$n_k^h(s;a) := \sum_{i=1}^{k} {}^{0;h}_{M; \; i}(s;a|s_0);$$

*where* ${}_i$ *is the exploration policy in episode $i$.*

The following lemma allows us to introduce the pseudo-counts when considering the contraction of the reward confidence intervals.

**Lemma 19** *With probability at least* $1 \quad \overline{2}$ *for all* $s;a;h;k \; 2 \; S \quad A \quad [H] \quad \mathbb{N}^+$, *we have:*

$$\min \; \frac{2\ell_k^h(s;a)}{n_k^h(s;a)};1 \qquad \frac{8\ell_k^h(s;a)}{\max \;\; n_k^h(s;a);1}$$

*where* $\ell_k^h(s;a) = \log \;\; 24SAH(n_k^h(s;a))^2= \quad .$

**Proof** This result adapts Lemma 7 by Kaufmann et al. (2021) to our setting.

By Lemma 10 in Kaufmann et al. (2021), we have with probability at least $1 \quad \overline{2}$:

$$n_k^h(s;a) \quad \frac{1}{2}n_k^h(s;a) \quad {}_{\mathrm{cnt}}( \; );$$

where ${}_{\mathrm{cnt}}( \; ) = \log(2SAH= \; )$.

We distinguish two cases. First let ${}_{\mathrm{cnt}}( \; ) \quad \frac{1}{4}n_k^h(s;a)$. Then $n_k^h(s;a) \quad \frac{1}{4}n_k^h(s;a)$, and

$$\min \; \frac{2\ell_k^h(s;a)}{n_k^h(s;a)};1 \qquad \frac{2\ell_k^h(s;a)}{\max(n_k^h(s;a);1)} = \frac{2\log(24SAH(n_k^h(s;a))^2= \; )}{\max(n_k^h(s;a);1)}$$
$$\frac{2\log(24SAH(n_k^h(s;a)=4)^2= \; )}{(n_k^h(s;a)=4)} \qquad \frac{8\ell_k^h(s;a)}{\max(n_k^h(s;a);1)}$$

where we use that $\log(24SAHx^2= \; )=x$ is non-increasing for $x > 1$, and $\log(24SAHx^2= \; )$ is non-decreasing and ${}_{\mathrm{cnt}}( \; ) \quad 1$.

Now consider let $\ell_{\text{cnt}}(\delta) > \frac{1}{4}n_k^h(s,a)$. Then,

$$\min\left(\frac{2\ell_k^h(s,a)}{n_k^h(s,a)},1\right) \le 1 < 4\frac{\ell_{\text{cnt}}(\delta)}{\max(n_k^h(s,a),1)} \le \frac{4\ell_k^h(s,a)}{\max(n_k^h(s,a),1)}$$

where we used that $\ell_k^h(s,a) = \log\left(24SAH(n_k^h(s,a))^2/\delta\right) = \ell_{\text{cnt}}(\delta) + \log\left(6n_k^h(s,a)\right)^2 \ge \ell_{\text{cnt}}(\delta)$. ■

The final lemma we need shows relates the error upper bound which is defined using our estimated transition model to a similar quantity defined using the (unknown) real transitions.

**Lemma 20** *Under the good event $E$, we have for any $s,a,h$:*

$$E_k^h(s,a) \le 2C_k^h(s,a) + \sum_{s'} P(s'|s,a)\max_{a'} E_k^{h+1}(s',a')$$

*where $P$ is the true transition model that we do not know.*

**Proof** First note that $E_k^h(s,a) \le H$ by definition. Now, consider:

$$\begin{aligned}
E_k^h(s,a) &\le C_k^h(s,a) + \sum_{s'} \hat{P}(s'|s,a)\max_{a'} E_k^{h+1}(s',a') \\
&= C_k^h(s,a) + \sum_{s'} (\hat{P}(s'|s,a) - P(s'|s,a) + P(s'|s,a))\max_{a'} E_k^{h+1}(s',a') \\
&= C_k^h(s,a) + \underbrace{\sum_{s'} (\hat{P}(s'|s,a) - P(s'|s,a))\max_{a'} E_k^{h+1}(s',a')}_{\le C_k^h(s,a)} + \sum_{s'} P(s'|s,a)\max_{a'} E_k^{h+1}(s',a') \\
&\le 2C_k^h(s,a) + \sum_{s'} P(s'|s,a)\max_{a'} E_k^{h+1}(s',a')
\end{aligned}$$

where we used the good event and the fact that $C_k^h$ can only shrink over episodes. ■

Finally, we can analyze the sample complexity of AceIRL Greedy.

**Theorem 21 (AceIRL Greedy Sample Complexity (problem independent))** *AceIRL Greedy terminates with an $(\epsilon, \delta, n)$-correct solution, with*

$$n \le \tilde{O}\left(\frac{H^5 R_{\max}^2 SA}{\epsilon^2}\right).$$

**Proof** Lemma 17 shows that if AceIRL Greedy terminates, then it returns a $(\epsilon, \delta, n)$-correct solution. So, we need to show that it terminates within $\tau$ iterations and bound $\tau$.

Let us consider the average error, defined by

$$\begin{aligned}
q_k^h &:= \sum_{s,a} \eta^{0,h}_{\mathcal{M},\pi_{k+1}}(s,a|s_0)E_k^h(s,a) \\
&\overset{(a)}{\le} \sum_{s,a} \eta^{0,h}_{\mathcal{M},\pi_{k+1}}(s,a|s_0)\left(2C_k^h(s,a) + \sum_{s'} P(s'|s,a)\max_{a'} E_k^{h+1}(s',a')\right) \\
&= \sum_{s,a} \eta^{0,h}_{\mathcal{M},\pi_{k+1}}(s,a|s_0)\left(2C_k^h(s,a) + \sum_{s'} P(s'|s,a)\sum_{a'}\pi_{k+1}(a'|s')E_k^{h+1}(s',a')\right) \\
&= 2\sum_{s,a} \eta^{0,h}_{\mathcal{M},\pi_{k+1}}(s,a|s_0)C_k^h(s,a) + q_k^{h+1}
\end{aligned}$$

where we used Lemma 20 in step (a). Unrolling the recursion, results in:

$$q_k^h \le 2 \sum_{h'=h}^{H} \sum_{s,a} \eta^{0,h'}_{\mathcal{M},\pi_{k+1}}(s,a|s_0) C_k^{h'}(s,a)$$

If the algorithm terminates at $\tau$, we have for each $k < \tau$, and $s,a,h \in \mathcal{S} \times \mathcal{A} \times [H]$: $\epsilon < 4E_k^0(s_0,\pi_{k+1}(s_0))$. We have $q_k^0 = E_k^0(s_0,\pi_{k+1}(s_0))$; therefore, as long we haven't stopped, we have $\epsilon \le 4q_k^0$. Writing out this inequality, yields:

$$\epsilon \le 4q_k^0 \le 8 \sum_{h=0}^{H} \sum_{s,a} \eta^{0,h}_{\mathcal{M},\pi_{k+1}}(s,a|s_0) C_k^h(s,a) \le 4HR_{\max} \sum_{h=0}^{H} \sum_{s,a} \eta^{0,h}_{\mathcal{M},\pi_{k+1}}(s,a|s_0) \sqrt{\frac{8\log(12SAH(n_k^h(s,a))^2/\delta)}{\max(n_k^h(s,a),1)}}$$

Using Lemma 19, we can relate this to the pseudo-counts

$$\begin{aligned} &< 4HR_{\max} \sum_{h=0}^{H} \sum_{s,a} \eta^{0,h}_{\mathcal{M},\pi_{k+1}}(s,a|s_0) \sqrt{\frac{8\log(12SAH(n_k^h(s,a))^2/\delta)}{\max(n_k^h(s,a),1)}} \\ &\le 4HR_{\max} \sum_{h=0}^{H} \sum_{s,a} \eta^{0,h}_{\mathcal{M},\pi_{k+1}}(s,a|s_0) \sqrt{\frac{8\log(12SAHk^2/\delta)}{\max(n_k^h(s,a),1)}} \end{aligned}$$

Summing the inequality over $k = 0,\dots,T$ with $T < \tau$, we obtain

$$\begin{aligned} (T+1)\epsilon &\le 4HR_{\max}\sqrt{8\log(12SAHT^2/\delta)} \sum_{h=0}^{H} \sum_{s,a} \sum_{k=1}^{T} \eta^{0,h}_{\mathcal{M},\pi_{k+1}}(s,a|s_0) \frac{1}{\sqrt{\max(n_k^h(s,a),1)}} \\ &= 4HR_{\max}\sqrt{8\log(12SAHT^2/\delta)} \sum_{h=0}^{H} \sum_{s,a} \sum_{k=1}^{T} \frac{n_h^{k+1}(s,a) - n_h^k(s,a)}{\sqrt{\max(n_k^h(s,a),1)}} \end{aligned}$$

where we used the definition of the pseudo-counts in the last equality. Using Lemma 19 by Jaksch et al. (2010), we can further bound the sum in $k$:

$$\begin{aligned} (T+1)\epsilon &= 4HR_{\max}\sqrt{8\log(12SAHT^2/\delta)} \sum_{h=0}^{H} \sum_{s,a} \sqrt{n_h^{T+1}(s,a)} \\ &\le 4HR_{\max}\sqrt{8\log(12SAHT^2/\delta)}\sqrt{SA} \sum_{h=0}^{H} \sqrt{\sum_{s,a} n_h^{T+1}(s,a)} \\ &= 4H^2R_{\max}\sqrt{8\log(12SAHT^2/\delta)}\sqrt{SA}\sqrt{T+1} \end{aligned}$$

It follows that

$$\begin{aligned} \epsilon\sqrt{T+1} &\le 4H^2R_{\max}\sqrt{8SA\log(12SAHT^2/\delta)} \\ \tau\epsilon^2 &\le 128H^4R_{\max}^2SA\log(12SAH(\tau-1)^2/\delta) \end{aligned}$$

setting $\tau = T+1$.

For large enough $\tau$, this inequality cannot hold because $\sqrt{T+1}$ on the l.h.s grows faster than $\log(\tau)$ on the r.h.s. Hence, the stopping time $\tau$ is finite. Further, we can apply Lemma 15 by Kaufmann et al. (2021), and follow that

$$\tau \le O\left(\frac{H^4R_{\max}^2SA}{\epsilon^2}\right)$$

If we observe $H$ samples in each iteration, i.e., $N_E = 1$, we get a sample complexity of

$$n \le O\left(\frac{H^5R_{\max}^2SA}{\epsilon^2}\right)$$

∎

### B.5 Sample Complexity of AceIRL in Unknown Environments (Problem Dependent)

For the problem dependent analysis, we will need this additional lemma also used by Kakade and Langford (2002).

**Lemma 22 (Lemma 6.1 by Kakade and Langford (2002))** *For any policy :*

$$V^{;h}_{M\,[\,r}(s) \quad V^{;h}_{M\,[\,r}(s) = \sum_{s';a'}\sum_{h'=h}^{H} {}^{h;h'}_{M;}(s';a';s)A^{;h'}_{M\,[\,r}(s';a')$$

**Proof**

$$\begin{aligned} & V^{;h}_{M\,[\,r}(s) \quad V^{;h}_{M\,[\,r}(s) \\ = & \sum_a \,_h(a|s) \left( r_h(s;a) + \sum_{s'} P(s'|s;a)V^{;h+1}_{M\,[\,r}(s') \right) \\ & \sum_a \,_h(a|s) \left( r_h(s;a) + \sum_{s'} P(s'|s;a)V^{;h+1}_{M\,[\,r}(s') \right) \quad \sum_{a;s'} \,_h(a|s)P(s'|s;a)V^{;h+1}_{M\,[\,r}(s') \\ = & \sum_a (\,_h(a|s) \quad _h(a|s))r(s;a) + \sum_{a;s'} (\,_h(a|s) \quad _h(a|s))P(s'|s;a)V^{;h+1}_{M\,[\,r}(s') \\ & + \sum_{a;s'} \,_h(a|s)P(s'|s;a)(V^{;h+1}_{M\,[\,r}(s) \quad V^{;h+1}_{M\,[\,r}(s)) \\ = & \sum_a \,(a|s)A^{;h}_{M\,[\,r}(s;a) + \sum_{a;s'} \,_h(a|s)P(s'|s;a)(V^{;h+1}_{M\,[\,r}(s) \quad V^{;h+1}_{M\,[\,r}(s)) \end{aligned}$$

Unrolling the recursion yields the result. ■

We can now start with the analysis. First, we define the policy confidence set, and show that it indeed contains the relevant policies under the good event.

**Definition 23** *We define the* policy confidence set *as*

$$\hat{}_k = \{ \; | V^{;}_{\hat M\,[\,\hat r}(s_0) \quad V^{;}_{\hat M\,[\,\hat r}(s_0) \quad 10\,_k \}$$

*where* $\hat r = A(R_{\hat B})$ *is the reward estimated using an IRL algorithm* $A$. *We choose* $_k$ *recursively by solving the optimization problem*

$$_k = \max_{\in \hat{}_{k\;1}} \sum_{h=0}^{H}\sum_{s';a'} {}^{0;h}_{\hat M;}(s';a';s_0)C^h_k(s';a')$$

*starting with* $_0 = \frac{1}{10}H$.

The following lemma will help us to deal with uncertainty about the transition dynamics.

**Lemma 24** *Under the good event* $E$, *if* $\in \hat{}_k$, *then:*

$$|V^{;h}_{\hat M\,[\,\hat r}(s) \quad V^{;h}_{M\,[\,\hat r}(s)| \quad _k$$
$$|V^{;h}_{M\,[\,\hat r}(s) \quad V^{;h}_{\hat M\,[\,\hat r}(s)| \quad _k$$

**Proof** First by Lemma 5:

$$\begin{aligned} |V^{;h}_{\hat M\,[\,r}(s) \quad V^{;h}_{M\,[\,r}(s)| & \sum_{h'=h}^{H}\sum_{s';a';s''} {}^{h;h'}_{\hat M;}(s';s)\,_{h'}(a'|s')|\hat P(s''|s';a') \quad P(s''|s';a')||V^{;h'+1}_{M\,[\,r}(s'') \\ & \sum_{h'=h}^{H}\sum_{s';a'} {}^{h;h'}_{\hat M;}(s';s)\,_{h'}(a'|s')C_k(s';a') \quad _k \end{aligned}$$

Then, by Lemma 6:

$$V^{\;;h}_{M\,[\,r}(s) \quad V^{\;;h}_{\hat{M}\,[\,r}(s) \quad \underset{h^0=h}{\mathrm{X}}\;\underset{s^0;a^0;s^{00}}{\mathrm{X}}\;{}^{h;h^0}_{\hat{M}\,;}(s^0;s)\;{}_{h^0}(a^0js^0)(P(s^{00}js^0;a^0) \quad \hat{P}(s^{00}js^0;a^0))V^{\;;h}_{M\,[\,r}(s^{00})$$
$$\underset{h^0=h}{\mathrm{X}}\;\underset{s^0;a^0}{\mathrm{X}}\;{}^{h;h^0}_{\hat{M}\,;}(s^0;s)\;{}_{h^0}(a^0js^0)C_k(s^0;a^0) \qquad {}_k$$

And, similarly

$$V^{\;;h}_{\hat{M}\,[\,r}(s) \quad V^{\;;h}_{M\,[\,r}(s) \quad \underset{h^0=h}{\mathrm{X}}\;\underset{s^0;a^0;s^{00}}{\mathrm{X}}\;{}^{h;h^0}_{\hat{M}\,;\hat{}}(s^0;s)\hat{}_{h^0}(a^0js^0)(\hat{P}(s^{00}js^0;a^0) \quad P(s^{00}js^0;a^0))V^{\;;h}_{\hat{M}\,[\,r}(s^{00})$$
$$\underset{h^0=h}{\mathrm{X}}\;\underset{s^0;a^0}{\mathrm{X}}\;{}^{h;h^0}_{\hat{M}\,;\hat{}}(s^0;s)\hat{}_{h^0}(a^0js^0)C_k(s^0;a^0) \qquad {}_k$$

■

Now we show that the relevant policies are always in the policy confidence set, conditioned on the good event.

**Lemma 25** *Conditioned the good event* $E$*, if* $\;;\hat{}\;\;2\;\hat{}_{k\quad 1}$*, then* $\;2\;\hat{}_k$.

**Proof** Let $r\;2\;R_{\mathrm{B}}$. Then

$$V^{\;;h}_{\hat{M}\,[\,\hat{r}_k}(s) \quad V^{\;;h}_{\hat{M}\,[\,\hat{r}_k}(s) = V^{\;;h}_{\hat{M}\,[\,\hat{r}_k}(s) \quad V^{\;;h}_{\hat{M}\,[\,r}(s) + V^{\;;h}_{\hat{M}\,[\,r}(s) \quad V^{\;;h}_{\hat{M}\,[\,\hat{r}_k}(s)$$
$$\overset{(a)}{\;}\;\underset{h^0=h}{\overset{H}{\mathrm{X}}}\;\underset{s^0;a^0}{\mathrm{X}}\;{}^{h;h^0}_{\hat{M}\,;}(s^0;a^0js)C^{h^0}_k(s^0;a^0) + \underset{h^0=h}{\overset{H}{\mathrm{X}}}\;\underset{s^0;a^0}{\mathrm{X}}\;{}^{h;h^0}_{\hat{M}\,;}(s^0;a^0js)C^{h^0}_k(s^0;a^0) \overset{(b)}{\;}\;2\;{}_k$$

where (a) uses Lemma 2, Lemma 3 and Corollary 10, (b) uses that $\;2\;\hat{}_{k\quad 1}$ and the definition of ${}_k$. Hence,

$$\max_s\; V^{\;;h}_{\hat{M}\,[\,\hat{r}_k}(s) \quad V^{\;;h}_{\hat{M}\,[\,\hat{r}_k}(s) \qquad 2\;{}_k \qquad 10\;{}_k$$

and therefore $\;2\;\hat{}_k$. ■

**Lemma 26** *Conditioned on the good event* $E$*, for every policy and episodes* $k^0 > k$*, there exists* $\hat{r}_{k^0}\;2\;R_{\hat{\mathrm{B}}\,k^0}$*, such that:*

$$\max_s\; V^{\;;h}_{M\,[\,\hat{r}_{k^0}}(s) \quad V^{\;;h}_{M\,[\,\hat{r}_k}(s) \qquad 4\;{}_k$$

**Proof** Similarly to the proof of the previous lemma, we have

$$V^{\;;h}_{\hat{M}\,[\,\hat{r}_{k^0}}(s) \quad V^{\;;h}_{\hat{M}\,[\,\hat{r}_k}(s) = V^{\;;h}_{\hat{M}\,[\,\hat{r}_{k^0}}(s) \quad V^{\;;h}_{\hat{M}\,[\,r}(s) + V^{\;;h}_{\hat{M}\,[\,r}(s) \quad V^{\;;h}_{\hat{M}\,[\,\hat{r}_k}(s)$$
$$\underset{h^0=h}{\overset{H}{\mathrm{X}}}\;\underset{s^0;a^0}{\mathrm{X}}\;{}^{h;h^0}_{\hat{M}\,;}(s^0;a^0js)C^{h^0}_{k^0}(s^0;a^0) + \underset{h^0=h}{\overset{H}{\mathrm{X}}}\;\underset{s^0;a^0}{\mathrm{X}}\;{}^{h;h^0}_{\hat{M}\,;}(s^0;a^0js)C^{h^0}_k(s^0;a^0) \qquad 2\;{}_k$$

where we use that the confidence intervals are shrinking with increasing episode number, i.e., ${}_{k^0} \qquad {}_k$.

By combining this with Lemma 24, we get the result:

$$\max_s\; V^{\;;h}_{M\,[\,\hat{r}_{k^0}}(s) \quad V^{\;;h}_{M\,[\,\hat{r}_k}(s)$$
$$= \max_s\; \underbrace{V^{\;;h}_{M\,[\,\hat{r}_{k^0}}(s) \quad V^{\;;h}_{\hat{M}\,[\,\hat{r}_{k^0}}(s)}_{k} + \underbrace{V^{\;;h}_{\hat{M}\,[\,\hat{r}_{k^0}}(s) \quad V^{\;;h}_{\hat{M}\,[\,\hat{r}_k}(s)}_{2\;k} + \underbrace{V^{\;;h}_{\hat{M}\,[\,\hat{r}_k}(s) \quad V^{\;;h}_{M\,[\,\hat{r}_k}(s)}_{k} \qquad 4\;{}_k$$

■

**Lemma 27** *Under the good event $E$, if $\hat{\pi}_k, \pi \in \hat{\Pi}_{k-1}$ and $\pi \notin \hat{\Pi}_k$, then the policy $\pi$ is suboptimal for some reward $\hat{r}_{k'} \in \mathcal{R}_{\hat{\mathcal{B}}_{k'}}$ for all $k' \geq k$.*

**Proof** We can observe that

$$
\begin{aligned}
&V^{\pi;h}_{M\cup\hat{r}_{k'}}(s_0) - V^{\pi;h}_{M\cup\hat{r}_{k'}}(s_0) = V^{\pi;h}_{M\cup\hat{r}_{k'}}(s_0) - V^{\hat{\pi}_k;h}_{M\cup\hat{r}_{k'}}(s_0) \\
&= \underbrace{V^{\pi;h}_{M\cup\hat{r}_{k'}}(s_0) - V^{\pi;h}_{M\cup\hat{r}_k}(s_0)}_{\overset{(a)}{\geq} -4\epsilon_k} + \underbrace{V^{\pi;h}_{M\cup\hat{r}_k}(s_0) - V^{\pi;h}_{\hat{M}\cup\hat{r}_k}(s_0)}_{\overset{(b)}{\geq} -\epsilon_k} \\
&\quad + \underbrace{V^{\pi;h}_{\hat{M}\cup\hat{r}_k}(s_0) - V^{\hat{\pi}_k;h}_{\hat{M}\cup\hat{r}_k}(s_0)}_{\overset{(c)}{>} 10\epsilon_k} + \underbrace{V^{\hat{\pi}_k;h}_{\hat{M}\cup\hat{r}_k}(s_0) - V^{\hat{\pi}_k;h}_{M\cup\hat{r}_k}(s_0)}_{\overset{(b)}{\geq} -\epsilon_k} \\
&\quad + \underbrace{V^{\hat{\pi}_k;h}_{M\cup\hat{r}_k}(s_0) - V^{\hat{\pi}_k;h}_{M\cup\hat{r}_{k'}}(s_0)}_{\overset{(a)}{\geq} -4\epsilon_k} > 0
\end{aligned}
$$

where we applied (a) Lemma 24, (b) Lemma 26, and (c) the definition of $\hat{\Pi}_k$ and the fact that $\pi \notin \hat{\Pi}_k$. Consequently, $\pi$ is suboptimal for at least some reward function $\hat{r}_{k'} \in \mathcal{R}_{\hat{\mathcal{B}}_{k'}}$. ■

**Corollary 28** *For $\epsilon_0 = \frac{H}{10}$, for every $k \geq 0$ it holds that both $\pi, \hat{\pi}_{k+1} \in \hat{\Pi}_k$.*

**Proof** We show the statement by induction over $k$. For $k = 0$, we have $10\epsilon_0 = H$ and therefore $\hat{\Pi}_0$ contains all policies. Assume that for $k - 1$ the statement holds, i.e., $\pi, \hat{\pi}_k \in \hat{\Pi}_{k-1}$, and consider $k$. By Lemma 25, $\pi \in \hat{\Pi}_k$. Note, that $\hat{\pi}_{k+1} \in \hat{\Pi}_{k-1}$. Hence, by Lemma 26, it follows that $\hat{\pi}_{k+1} \in \hat{\Pi}_k$ because it would be suboptimal otherwise which is a contradiction. ■

The last result we need, is quantifying the size of the policy confidence set.

**Lemma 29** *Under the good event $E$, let $\tilde{r} \in \operatorname{argmin}_{r\in\mathcal{R}_{\mathcal{B}}} \max_{s,a}(r(s,a) - \hat{r}_k(s,a))$, where $\hat{r}_k = \mathcal{A}(\mathcal{R}_{\hat{\mathcal{B}}_k})$. If $\pi \in \hat{\Pi}_k$, then $\max_s(V^{\pi;h}_{\hat{M}\cup\tilde{r}}(s) - V^{\pi;h}_{\hat{M}\cup\tilde{r}}(s)) \leq 12\epsilon_k$.*

**Proof**

$$
V^{\pi;h}_{\hat{M}\cup\tilde{r}}(s) - V^{\pi;h}_{\hat{M}\cup\tilde{r}}(s) = \underbrace{V^{\pi;h}_{\hat{M}\cup\tilde{r}}(s) - V^{\pi;h}_{\hat{M}\cup\hat{r}_k}(s)}_{\epsilon_k} + \underbrace{V^{\pi;h}_{\hat{M}\cup\hat{r}_k}(s) - V^{\pi;h}_{\hat{M}\cup\hat{r}_k}(s)}_{10\epsilon_k} + \underbrace{V^{\pi;h}_{\hat{M}\cup\hat{r}_k}(s) - V^{\pi;h}_{\hat{M}\cup\tilde{r}}(s)}_{\epsilon_k} \leq 14\epsilon_k
$$

■

Next, we define the error upper bound based on the policy confidence set.

**Definition 30** *Using $\hat{\Pi}_k$, we define recursively:*

$$
\begin{aligned}
\hat{E}^H_k(s,a) &= 0 \\
\hat{E}^h_k(s,a) &= \min\Big((H-h)R_{\max}, C^h_k(s,a) + \sum_{s'} \hat{P}(s'|s,a) \max_{\pi\in\hat{\Pi}_{k-1}} \pi(a'|s')\hat{E}^{h+1}_k(s',a')\Big)
\end{aligned}
$$

*where $\hat{P}$ is the estimated transition model of the environment. In contrast to Definition 14, the maximization is over policies in $\hat{\Pi}_k$ rather than all actions.*

This definition allows us to derive results that are analogous to the problem independent case.

**Lemma 31** *Under the good event $E$, for all policies $\quad 2\ \hat{}_k$ and reward functions $r$ and all $s; a\ 2\ S \quad A$:*

$$jQ^{\ ;h}_{\hat{M}\,[\,r}(s;a) \quad Q^{\ ;h}_{M\,[\,r}(s;a)j \quad \hat{E}^h_k(s;a)$$

**Proof** The proof is the same as for Lemma 15, restricting the set of policies to $\hat{}_k$. ■

**Lemma 32** *Under the good event $E$, for all reward function $r$, all policies $\quad 2\ \hat{}_k$, and all $s; a\ 2\ S \quad A$:*

$$jQ^{\ ;h}_{\hat{M}\,[\,\hat{r}}(s;a) \quad Q^{\ ;h}_{\hat{M}\,[\,r}(s;a)j \quad \hat{E}^h_k(s;a)$$

**Proof** The proof is the same as for Lemma 16, restricting the set of policies to $\hat{}_k$. ■

**Lemma 33** *Under the good event $E$, we have for any $s; a; h$ :*

$$\hat{E}^h_k(s;a) \quad 2C^h_k(s;a) + \underset{s^0}{X} P(s^0js;a) \max_{2\ \hat{}_{k\ \ 1}} \quad (a^0js^0)\hat{E}^{h+1}_k(s^0;a^0)$$

**Proof** The proof is the same as for Lemma 33. ■

Finally, we can combine these results to analyze the algorithm's sample complexity.

**Theorem 7** *[AceIRL Sample Complexity] AceIRL returns a ( , , $n$)-correct solution with*

$$n \quad O \quad \min \; {}^{"}\frac{H^5R^2_{\max}SA}{2}; \frac{H^4R^2_{\max}SA\ {}^2\ {}_1}{\min_{s;a;h}(A^{\ ;h}_{M\,[\,r}(s;a))^2\ {}^2}{}^{\#!}$$

*where $\quad {}_1$ depends on the choice of $N_E$, the number of episodes of exploration in each iteration. $A^{\ ;h}_{M\,[\,r}(s;a)$ is the advantage function of $r\ 2\ \mathrm{argmin}_{r\,2\,R_{\mathrm{B}}} \max_{h;s;a}(r_h(s;a) \quad \hat{r}_{k;h}(s;a))$, the reward function from the feasible set $R_{\mathrm{B}}$ closest to the estimated reward function $\hat{r}_k$.*

**Proof** First note that the analysis of Theorem 21 still applies; so, in the worst case we get the same sample complexity. The key difference is that we no longer use the overall greedy policy w.r.t $E^h_k$, but restrict ourselves to policies in $\hat{}_k$.

Again, we consider the error

$$e_k^{\ ;h}(s;a) := jQ^{\ ;h}_{M\,[\,r}(s;a) \quad Q^{\hat{}\ ;h}_{M\,[\,r}(s;a)j$$

where $\quad$ is the true optimal policy in $M\ [\ r$, and $\hat{}$ is the optimal policy in $\hat{M}\ [\ \hat{r}$, i.e., in the estimated MDP using the inferred reward function.

Similar, to the proof of Lemma 17, we can use Lemma 31 and Lemma 32 to show for all policies $\quad 2\ \hat{}^h_k$, that:

$$e_k^{\ ;h}(s;a) \quad 4\hat{E}^h_k(s;a)$$

which implies the correctness of the algorithm according to Corollary 11 when stopping at

$$\hat{E}^0_k(s_0; \quad {}_{k+1}(s_0)) \quad \frac{}{4} \tag{2}$$

Now, consider the following condition for all $s; a; h$:

$$C_k^h(s;a) \qquad A^{\ ;h}_{M\,[\,\hat{r}}(s;a)\frac{}{48 \ \ _k \ \ _1}; \tag{3}$$

where $\hat{r}\ 2$ argmin$_{r\,2\,R_{\,B}}$ max$_{h;s;a}(r_h(s;a) \quad \hat{r}_{k;h}(s;a))$. We will (a) show that when this condition holds the previous stopping condition also holds, and (b) analyze after how many iterations this condition will certainly hold. Together this will yield the result.

To show that Equation (3) implies Equation (2), we assume that Equation (3) holds. Then, we get by applying Lemma 33 recursively:

$$\begin{aligned}\hat{E}_k^0(s_0; \quad _{k+1}(s_0)) \quad & 2 \max_{2\,\hat{}\,_k\ \ _1} \max_a \sum_{h=0}^{\hat{H}} \sum_{s^0;a^0} \ ^{0;h}_{M;}\ (s^0;a^0;s_0;a)C_k^h(s^0;a^0) \\ & 2 \max_{2\,\hat{}\,_k\ \ _1} \max_a \sum_{h=0}^{\hat{H}} \sum_{s^0;a^0} \ ^{0;h}_{M;}\ (s^0;a^0;s_0;a) \qquad A^{\ ;h}_{M\,[\,\hat{r}}(s^0;a^0)\frac{}{48 \ \ _k \ \ _1} \\ & \overset{(a)}{} \ 2 \max_{2\,\hat{}\,_k\ \ _1} (V^{\ ;0}_{M\,[\,r}(s_0) \qquad V^{\ ;0}_{M\,[\,r}(s_0))\frac{}{48 \ \ _k \ \ _1} \quad \overset{(b)}{} \quad \frac{}{4}\end{aligned}$$

where (a) uses Lemma 22 and (b) uses Lemma 29.

Next, we analyze after how many iterations Equation (3) holds, which will give a lower bound on the sample complexity result. The argument proceeds similar to the proof of Theorem 21.

Before the algorithm terminates at , we have for all $k <$ :

$$\min_{s;a;h}( \quad A^{\ ;h}_{\hat{M}\,[\,\hat{r}}(s;a))\frac{}{48 \ \ _k \ \ _1} < \max_{s;a;h} C_k^h(s;a) \qquad HR_{\max} \ \mathrm{s}\frac{2\grave{}_k^h(s;a)}{\max(N_k^h(s;a); \ )}$$

Using similar argument to the proof of Theorem 21, using the same pseudo-counts, we arrive at:

$$\min_{s;a;h}( \quad A^{\ ;h}_{M\,[\,\hat{r}}(s;a))\frac{}{48 \qquad _1} \quad \mathrm{p}\overline{\qquad + 1} \qquad HR_{\max} \quad \mathrm{p}\overline{8SA\log(12SAH \quad ^2= \ )}$$

Again, we can use Lemma 15 by Kaufmann et al. (2021) to find that

$$O \quad \frac{H^3R^2_{\max}SA \quad ^2 \quad _1}{\min_{s;a;h}(A^{\ ;h}_{M\,[\,\hat{r}}(s;a))^2 \ \ ^2} \ !$$

■

## B.6 Computing the Exploration Policy

To run AceIRL, we need to solve the optimization problem:

$$^h_k = \min_{\hat{}\,2\,\hat{}\,_k \ \ _1} \max \sum_{h=0}^{\hat{H}} \sum_{s^0;a^0} \ ^{0;h}_{\hat{M};\hat{}}(s^0;a^0;s_0)\hat{C}_k^h(s^0;a^0j \quad )$$

For simplicity let us denote the state visitation frequencies by

$$\begin{aligned}{}_h(s;a) := & \ ^{0;h}_{\hat{M};} \quad (s;a;s_0) \\ \hat{}_h(s;a) := & \ ^{0;h}_{\hat{M};\hat{}}(s;a;s_0)\end{aligned}$$

Let us introduce the following matrix notation

$$\tilde{A} = \begin{bmatrix} I & 0 & 0 & 0 & \dots & 0 \\ \hat{P} & I & 0 & 0 & \dots & 0 \\ 0 & \hat{P} & I & 0 & \dots & 0 \\ & & \dots & & & \\ 0 & 0 & \dots & 0 & \hat{P} & I \\ I & 0 & 0 & \dots & 0 & 0 \\ 0 & I & 0 & \dots & 0 & 0 \\ & & \dots & & & \\ 0 & 0 & 0 & \dots & I & 0 \\ 0 & 0 & 0 & \dots & 0 & I \end{bmatrix}; \qquad a = \begin{bmatrix} \hat{r}^0_{k\ 1} \\ \hat{r}^1_{k\ 1} \\ \dots \\ \hat{r}^H_{k\ 1} \end{bmatrix}; \qquad A = \begin{bmatrix} A & 0 \\ a^T & 1 \end{bmatrix};$$

$$x = \begin{bmatrix} _0 \\ _1 \\ \dots \\ _H \\ t \end{bmatrix}; \qquad \hat{x} = \begin{bmatrix} \hat{}_0 \\ \hat{}_1 \\ \dots \\ \hat{}_H \end{bmatrix}; \qquad b = \begin{bmatrix} _0 \\ 0 \\ \dots \\ 0 \\ 1 \\ \dots \\ 1 \\ 10\ _{k\ 1} \end{bmatrix}; \qquad c = \begin{bmatrix} C_0 \\ C_1 \\ \dots \\ C_H \\ 1 \end{bmatrix};$$

where $_0$ is the actual initial state distribution of the environment (which we assume to know). We can now write the inner maximization problem above as a linear program:

$$\begin{aligned} \max_x \quad & c^T x \\ & Ax = b \\ & x \quad 0 \end{aligned}$$

The corresponding dual problem is:

$$\begin{aligned} \min_y \quad & b^T y \\ & A^T y \quad c \end{aligned}$$

Using this we can write the full min-max problem as:

$$\begin{aligned} \min_{\hat{x};y} \quad & b^T y \\ & A^T y \quad c(x) \\ & \tilde{A}x = b \\ & x \quad 0 \end{aligned}$$

which is a convex optimization problem, if we use:

$$C_h(s;a) = 2(H \quad h)R_{\max} \sqrt{\frac{2\log \ 24SAH(\max(1;n^h_k(s;a)))^2=}{\max(1;\tilde{n}^h_{k+1}(s;a))}})$$

where $\tilde{n}^h_{k+1}(s;a) = n^h_k(s;a) + \ _h(s;a) \quad N_E$ is the number of times we expect $h;s;a$ to be visited at the next iteration.

Solving this optimization problem yields the state-visitation frequencies $\hat{}_k(s;a)$. We can then find the exploration policy that induces these state-visitations simply as:

$$_{k;h}(a|s) := \frac{\hat{}^h_k(s;a)}{\sum_{a'} \hat{}^h_k(s;a')}:$$

## Appendix C. Experimental Details

In this section, we provide more details on our experiments. We discuss the environments in detail (Appendix C.1), provide some information on the implementation and the libraries and computational resources we used (Appendix C.2), and we provide more full plots of all experiments we discussed in the main paper (Appendix C.3).

### C.1 Details on the Environments

**Four Paths.** The four paths environment has $41$ states and $4$ actions:

$$S = \{c, l_1, \dots, l_{10}, u_1, \dots, u_{10}, r_1, \dots, r_{10}, d_1, \dots, d_{10}\}, \qquad A = \{a_1, a_2, a_3, a_4\},$$

and a time horizon of $H = 20$. The agent starts in the center state $c$, from which can move in four directions: left ($a_1$), up ($a_2$), right ($a_3$), or down ($a_4$). Each action $a_i$ has a probability $p_i$ of failing. If an action fails it moves in the opposite direction. $p_1, \dots, p_4$ are sampled uniformly from $(0, 0.3)$. One of the states $(l_{10}, u_{10}, r_{10}, d_{10})$ is chosen as the goal state at random. The reward in the goal state is $1$, all other rewards are $0$.

**Double Chain.** The *Double Chain* MDP, proposed by Kaufmann et al. (2021), consists of $L$ states $S = \{s_0, \dots, s_{L-1}\}$, and two actions $A = \{\text{left}, \text{right}\}$, which correspond to a transition to the left or to the right. When the agent takes an action, there is a $0.1$ probability of moving to the other direction. The state $s_{L-1}$ has reward $1$, all other states have reward $0$, and the agent starts in the center of the chain at $s_{(L-1)/2}$. We choose $L = 31$, similar to Kaufmann et al. (2021). The environment has horizon $H = 20$.

**Chain.** The *Chain* MDP, proposed by Metelli et al. (2021) has $6$ states $S = \{s_1, s_2, s_3, s_4, s_5, s_u\}$ and $10$ actions $A = \{a_1, \dots, a_{10}\}$. The agent starts in a random initial state. Taking action $a_{10}$ moves it right along the chain with probability $0.7$ and to state $s_u$ with probability $0.3$. Any other action moves the agent right with probability $0.3$ and to state $s_u$ with probability $0.7$. If the agent is in state $s_u$, action $a_{10}$ moves it back to state $s_1$ with probability $0.05$. Any other action moves it to $s_1$ with probability $0.01$. The reward is $1$ in all states except $s_u$ where the reward is $0$. Metelli et al. (2021) provide an illustration of the environment in Figure 3. We chose $H = 10$ for the chain.

**Gridworld.** The *Gridworld*, proposed by Metelli et al. (2021), is a $3 \times 3$ gridworld with an obstacle in the center cell $(2, 2)$ and a goal cell at the right center cell $(2, 1)$. The agent starts in a random non-goal cell, and it has $4$ action one to move in each direction. If the agent takes an action with probability $0.3$ the action fails and the agent moves in a random direction instead. If the agent is in the center cell $(2, 2)$ which has the obstacle, if the agent would move right it instead stays in the center cell with probability $0.8$. The reward in the goal cell is $1$, all other rewards are $0$. Metelli et al. (2021) provide an illustration of the gridworld in Figure 6. We chose $H = 10$ for the gridworld.

**Random MDPs.** We generate random MDPs by uniformly sampling an initial state distribution and transition matrix and normalizing them. The rewards are sampled uniformly between $0$ and $1$. Our random MDPs have $9$ states, $4$ actions and horizon $10$.

### C.2 Implementation Details

We provide a full implementation of AceIRL in Python, using multiple open sources libraries, including `cvxpy` and the SCS optimizer (Diamond and Boyd, 2016; O'Donoghue et al., 2016) for solving the optimization problem in Appendix B.6, and standard libraries for numerical computing, including `numpy`, and `scipy`. We choose Maximum Entropy IRL (Ziebart et al., 2008) as an IRL algorithm, but AceIRL is agnostic to this choice.

We ran experiments in parallel on a server with two 64 Core AMD EPYC 7742 2.25GHz processors. We estimate a total wall-clock time of less than 48 hours for running all experiments presented in this paper, including $50$ random seeds each.

### C.3 Additional Results

We provide full learning curves for all experiments discussed in the main paper in Figure C.1.

Figure C.1: Full learning curves for all experiments shown in Table 1. Similar to Figure 2, we show the mean and 95% confidence intervals computed over 50 random seeds. In addition to the exploration algorithms, we also show uniform sampling and TRAVEL which are much faster in most cases because they have access to a generative model.

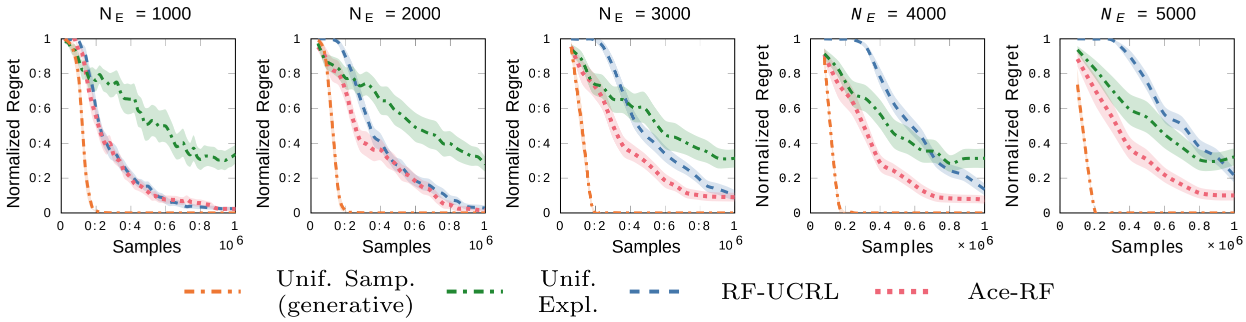

Figure D.2: Illustrative experiments for reward-free exploration in the *Double Chain* environment proposed by Kaufmann et al. (2021). The difference to our Active IRL setting is that the agent does not have access to the expert policy during exploration, but still tries to learn a good model of the environment. During testing it then gets access to the reward function, and the regret measures the suboptimality of the policy trained in the agent's transition model. We find that the ideas used in AceIRL are also useful for batched reward-free exploration with larget $N_E$.

## Appendix D. Connection to Reward-free Exploration

In the *reward-free exploration* problem, introduced by Jin et al. (2020), the agent explores an MDP$n$R to learn a transition model. In each iteration it chooses a new exploration policy based on previous data. The goal is to ensure that if the agent is given a reward function $r$ after the exploration phase it can find a good policy using its transition model. Jin et al. (2020) formalize this goal as reducing the error:

$$V_{M\,[\,r}^{\;;0}(s_0) \qquad V_{M\,[\,r}^{\hat{}\;;0};$$

where ^ is the optimal policy in the estimated MDP $\hat{M}\ [\ r$. Note the striking similarity between this problem, and the active IRL problem, we study in this paper. We want to reduce a similar error (cf. Definition 2), but we have additional information about the reward in form of the expert policy.

The *Reward-free UCRL* algorithm, proposed by Kaufmann et al. (2021), is essentially analogous to AceIRL Greedy (Section 6.1). Reward-free UCRL explores greedily with respect to an upper bound on the value function error. However, the exploration policy needs to be updated after each episode to adapt to the new uncertainty estimates. This might be expensive or not possible in practice. Instead, we could consider a *batched* version of reward-free exploration, where in each iteration the agent explores for $N_E$ episodes, similar to our Active IRL problem. In this setting, a greedy policy w.r.t. uncertainty is suboptimal because it does not adapt to the reduced uncertainty over the $N_E$ episodes.

Instead, we can consider reducing the expected uncertainty at the next iteration, similar to our discussion in Section 6.2. If our error estimate is denoted by $E_k(s;a)$, we do no longer act greedily w.r.t. $E_k$. Instead we try to estimate the error at the next iteration $\hat{E}_{k+1}(s;aj\ \ )$ as a function of the policy and try to select the policy that reduces this error. In the tabular case, we can formulate this as a convex optimization problem, analogous to Appendix B.6. We call this adaptation of AceIRL to the reward-free exploration problem *Ace-RF*.

Figure D.2 shows illustrative results of this algorithm in the batched reward-free exploration setting in the *Double Chain* environment. We find that for larger batch sizes, choosing an exploration policy that reduces future uncertainty is significantly better than reward-free UCRL.