On the Effective Horizon of Inverse Reinforcement Learning

Yiqing Xu
National University of Singapore
School of Computing, Singapore
xuyiqing@comp.nus.edu.sg

Finale Doshi-Velez
Harvard University
Department of Computer Science
Cambridge, USA
finale@seas.harvard.edu

David Hsu
National University of Singapore
School of Computing, Smart System Institute, Singapore
dyhsu@comp.nus.edu.sg

ABSTRACT

Inverse reinforcement learning (IRL) algorithms often rely on (forward) reinforcement learning or planning, over a given time horizon, to compute an approximately optimal policy for a hypothesized reward function; they then match this policy with expert demonstrations. The time horizon plays a critical role in determining both the accuracy of reward estimates and the computational efficiency of IRL algorithms. Interestingly, an *effective time horizon* shorter than the ground-truth value often produces better results faster. This work formally analyzes this phenomenon and provides an explanation: the time horizon controls the complexity of an induced policy class and mitigates overfitting with limited data. This analysis provides a guide for the principled choice of the effective horizon for IRL. It also prompts us to re-examine the classic IRL formulation: it is more natural to learn jointly the reward and the effective horizon rather than the reward alone with a given horizon. To validate our findings, we implement a cross-validation extension and the experimental results support the theoretical analysis. The project page and code are publicly available.

KEYWORDS

Inverse Reinforcement Learning; Imitation Learning

ACM Reference Format:
Yiqing Xu, Finale Doshi-Velez, and David Hsu. 2025. On the Effective Horizon of Inverse Reinforcement Learning. In *Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Detroit, Michigan, USA, May 19 – 23, 2025*, IFAAMAS, 18 pages.

1 INTRODUCTION

Inverse reinforcement learning (IRL) [23] aims to infer the underlying task objective from expert demonstrations. One common approach is to estimate a reward function that induces a policy matching closely the demonstrated data. This model-based approach holds the promise of generalizing the learned reward function and the associated policy over states not seen in the demonstrations [25].

Many existing IRL algorithms follow the classic formulation and assume a known ground-truth discount factor (or equivalently, time horizon) for the expert demonstrations [1, 3–6, 11, 12, 15, 18, 21, 23, 24, 27–29, 32, 34]. They estimate the reward function based on this specified time horizon. However, we do have the flexibility to choose a different time horizon when learning the reward function and optimizing the policy.

Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Y. Vorobeychik, S. Das, A. Nowé (eds.), May 19 – 23, 2025, Detroit, Michigan, USA. © 2025 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

Surprisingly, we find that using a *horizon* shorter than the ground-truth value often produces better results faster, especially when expert data is scarce. Why? Intuitively, the time horizon controls the complexity of an induced policy class. With limited data, a shorter time horizon is preferred, as the induced policy class is simpler and mitigates overfitting. We refer to the horizon or discount factor used during learning as the *effective horizon* or *effective discount factor*, as it holds promise to enhance learning effectiveness.

We present a formal analysis showing that, with limited expert data, using a shorter discount factor or horizon improves the generalization of the learned reward function to unseen states. In IRL settings where the effective horizon varies, the performance gap between the induced policy and the expert policy arises from two sources: (i) reward estimation error due to limited data during IRL, and (ii) policy optimization error from using an effective horizon shorter than the ground-truth. We prove that the effective horizon controls the complexity of the approximated policy class during IRL. As the horizon increases, reward estimation error grows—overfitting occurs because we estimate policies from a more complex class using limited expert data. On the other hand, as the horizon approaches the ground-truth, the policy optimization error decreases. These opposing errors suggest that an intermediate effective horizon balances this trade-off and produces the most expert-like policy.

Based on our theoretical findings, we propose a more natural and higher-performing formulation of the IRL problem: jointly learning the reward function and the effective horizon/discount factor. To validate this approach, we extend the linear programming IRL algorithm (LP-IRL) [23] and the maximum entropy IRL algorithm (MaxEnt-IRL) [34], using cross-validation to simultaneously learn the reward function and the effective horizon. Our experimental evaluation of these extended algorithms across four different tasks corroborates our theoretical analysis.

This work presents the first formal analysis of the relationship between the horizon and the performance of the learned reward function in IRL. Previous studies have examined the impact of the horizon in Reinforcement Learning (RL) [2, 10, 13, 16], approximate dynamic programming [26], and planning [14, 20], but the effects when the reward function is unknown and inferred from data remain under-explored. Our work addresses this gap by providing a theoretical analysis on how changes in the horizon affect policy performance in IRL. Some IRL algorithms employ smaller effective time horizons for computational efficiency [17, 19, 33], while others learn discount factors from data to better align with demonstrations, sometimes incorporating multiple feedback types [8, 9]. Our theoretical analysis on the role of varying time horizons complements existing work and offers valuable insights to guide future IRL research.

2 RELATED WORKS

Effective Horizon of Imitation Learning. Imitation learning learns desired behaviors by imitating expert data and comprises two classes of methods: model-free behavior cloning (BC) and model-based inverse reinforcement learning (IRL) [25]. The primary distinction between BC and IRL lies in the horizons used to align the learned behaviors with expert data. BC matches step-wise expert actions, resulting in poor generalization to unseen states. In contrast, IRL addresses this issue by either matching multi-step trajectory distributions [3–5, 18, 27, 29, 32, 34] or their marginalized approximations [6, 7, 11, 12, 15, 24]. The former employs a double-loop structure to interleave the policy optimization and reward function update, while the latter learns a discriminator to distinguish expert-like behaviors. Both approaches utilize the ground-truth horizon/discount factor for policy optimization, ensuring global temporal consistency between the learned policy and expert. Notably, few IRL methods adopt receding horizons to reduce computational cost[17, 19, 33], claiming shorter optimization horizons yields sub-optimal policies. Moreover, several works focus onfi nding the optimal discount factors in an IRL context [8, 9]. However, the absence of a theoretical analysis on the horizon's impact in IRL leaves an important yet overlooked gap in thefield.

Theoretical Analysis on Effective Horizon. Several studies have examined how planning horizons affect Reinforcement Learning (RL) [2, 10, 13, 16], approximate dynamic programming [26], and planning [20], particularly when transition models are estimated from samples [14]. In these works, which assume a known reward function, differences in planning horizons directly affect policy performance. However, the effect of horizons remains unexplored when the reward function is learned from data. Our study addresses this gap by investigating how horizon changes affect policy performance when the reward function is derived from expert demonstrations. Under this setting, policy optimization depends on both the horizon and the horizon-dependent reward function estimation.

This dual dependency introduces unique analytical challenges not present in forward RL or planning. The key challenge lies in examining how varying discount factors influence reward estimates when expert data is limited. While Metelli et al. [21] define a set of reward functions that remain compatible with limited expert demonstrations under afi xed, known horizon, we extend this formulation to accommodate unknown and varying horizons. By considering this extended class of feasible reward functions, we analyze how horizon variations affect reward estimation and subsequent policy optimization, particularly with scarce expert data.

3 PROBLEM FORMULATION

We consider an MDP (S, A, P, R_0, γ_0), where S and A represent the state and action spaces, respectively. The transition function is denoted by $P : S \times A \times S \to [0, 1]$, and the ground-truth reward function is $R_0 : S \times A \to [0, R_{max}]$. The discount factor, γ_0, implicitly determines the value of future rewards at the current time step. The optimal policy, $\pi^*_{R_0,\gamma_0}$, maximizes the total discounted reward based on R_0 and γ_0. In our setting, we are given the MDP without the reward function R_0 or the discount factor γ_0. Instead, we have a set of N expert demonstrated state-action pairs $D = \{(s_0, a_0), (s_1, a_1), ...(s_{N-1}, a_{N-1})\}$ sampled from a deterministic policy $\pi^*_{R_0,\gamma_0}$. Our analysis employs the discount factor for simplicity; however, thefi ndings are equally applicable to the planning horizon since both determine how much future rewards are valued. A smaller discount factor effectively limits the agent's planning horizon by diminishing the importance of distant rewards.

We examine the scenario where both the reward function and discount factor $(\widehat{R}, \widehat{\gamma})$ are jointly learned from the limited expert demonstrations. The scarcity of data suggests that $(\widehat{R}, \widehat{\gamma})$ is susceptible to approximation errors, which consequently affects the induced optimal policy $\pi^*_{\widehat{R},\widehat{\gamma}}$. The quality of the pair $(\widehat{R}, \widehat{\gamma})$ is assessed by comparing the value of its induced policy $\pi^*_{\widehat{R},\widehat{\gamma}}$ with the ground-truth optimal policy $\pi^*_{R_0,\gamma_0}$, both evaluated under the true (R_0, γ_0) for fairness. We define the loss as $\left\| V^{\pi^*_{R_0,\gamma_0}}_{R_0,\gamma_0} - V^{\pi^*_{\widehat{R},\widehat{\gamma}}}_{R_0,\gamma_0} \right\|_\infty$, where $V^\pi_{R,\gamma}$ is the value function of π under (R,γ). The "best" policy $\pi^*_{\widehat{R},\widehat{\gamma}}$ is the one that minimizes this loss. During learning, each choice of $\widehat{\gamma}$ induces a different policy class $\Pi_{\widehat{\gamma}}$ with its own complexity, as defined below.

Definition 3.1 (Policy Class and Complexity Measure)**.** For a given $\widehat{\gamma}$, the policy class $\Pi_{\widehat{\gamma}}$ comprises all optimal policies for afi xed state space S, action space A, and transition function P, for any reward function $R \in \mathcal{F}_R$ satisfying the following mild condition. Formally,

$$\Pi_{\widehat{\gamma}} = \left\{\pi \;\middle|\; \exists R \in \mathcal{F}_R : \pi \text{ is optimal in } (S, A, P, R, \widehat{\gamma})\right\},$$

where $\mathcal{F}_R$ is the set of reward functions such that, for each state $s \in S$, there is exactly one action $a^*(s)$ for which $R(s, a^*(s))$ strictly surpasses $R(s, a)$ for all $a \neq a^*(s)$. The complexity of this policy class is measured by $|\Pi_{\widehat{\gamma}}|$, the number of distinct optimal policies.

For a given discount factor $\widehat{\gamma}$, IRL algorithms learn a reward function $\widehat{R}$ from limited expert data (N). This reward function yields an optimal policy $\pi^*_{\widehat{R},\widehat{\gamma}}$ from the policy set $\Pi_{\widehat{\gamma}}$ that closely matches the expert data. The complexity of $\Pi_{\widehat{\gamma}}$, measured by $|\Pi_{\widehat{\gamma}}|$, influences the degree of overfitting in this low-data regime, making it a key factor in IRL. The optimal effective discount factor $\widehat{\gamma}^*$ is defined as the one whose induced optimal policy minimizes the loss:

$$\widehat{\gamma}^* = \underset{0 \leq \widehat{\gamma} \leq \gamma_0}{\arg\min} \left\| V^{\pi^*_{R_0,\gamma_0}}_{R_0,\gamma_0} - V^{\pi^*_{\widehat{R},\widehat{\gamma}}}_{R_0,\gamma_0} \right\|_\infty .$$

We use this loss to examine the interplay among the amount of expert data, the discount factor, and the resulting policy's performance. In particular, we investigate how to choose $\widehat{\gamma} \leq \gamma_0$ to minimize this loss.

4 ANALYSIS

4.1 Overview

We define the *effective horizon* as the horizon (or discount factor) used in learning the reward function and optimizing policies, which may differ from the ground-truth horizon. We formally analyze how this effective horizon influences the quality of the learned reward function under varying amounts of expert data. Our main result, Theorem 4.1, shows that when expert data is limited, using a discount factor smaller than the ground-truth value enables IRL methods to learn reward functions that induce policies more closely aligned with the expert.

Theorem 4.1. *Let (S, A, P) be a controlled Markov process shared by two MDPs: the ground-truth MDP (S, A, P, R_0, γ_0) with reward function $R_0 : S \times A \to [0, R_{\max}]$ and discount factor $\gamma_0 \in (0, 1)$; and the estimated MDP $(S, A, P, \widehat{R}, \widehat{\gamma})$ with reward function $\widehat{R} : S \times A \to [0, R_{\max}]$ and discount factor $\widehat{\gamma} \in (0, 1)$, estimated from N expert state-action pairs. Let $|\Pi_{\widehat{\gamma}}|$ denote the complexity of the policy class induced by the estimated effective horizon $\widehat{\gamma}$, and suppose the optimal policy $\pi^*_{\widehat{R},\widehat{\gamma}}$ derived from $(\widehat{R}, \widehat{\gamma})$ belongs to this class, i.e., $\pi^*_{\widehat{R},\widehat{\gamma}} \in \Pi_{\widehat{\gamma}}$. Then, for the optimal policies $\pi^*_{R_0,\gamma_0}$ and $\pi^*_{\widehat{R},\widehat{\gamma}}$ induced by the ground-truth and estimated parameters, respectively, the difference between their value functions evaluated under R_0 and γ_0 is bounded with probability at least $1 - \delta$ by*

$$\left\| V^{\pi^*_{R_0,\gamma_0}}_{R_0,\gamma_0} - V^{\pi^*_{\widehat{R},\widehat{\gamma}}}_{R_0,\gamma_0} \right\|_\infty \leq \frac{2R_{\max}}{(1-\widehat{\gamma})^2}\sqrt{\frac{1}{2N}\log\frac{|S||\Pi_{\widehat{\gamma}}|}{2\delta}} + \frac{\gamma_0 - \widehat{\gamma}}{(1-\gamma_0)(1-\widehat{\gamma})}R_{max}.$$

Intuitively, Theorem 4.1 bounds the performance disparity between the policy induced by the learned $(\widehat{R}, \widehat{\gamma})$ and the expert policy as a sum of two terms: the first term bounds the *reward function estimation error* that arises from using a limited expert data and an effective $\widehat{\gamma}$ during IRL, while the second term measures the *policy performance gap* when optimized using $\widehat{\gamma} < \gamma_0$ using that estimated reward function. When $\widehat{\gamma}$ increases, the first term, *i.e.*, the *reward function estimation error*, grows due to overfitting arising from estimating a policy from an increasingly complex class $\Pi_{\widehat{\gamma}}$ using only limited expert data, while the second term, *i.e.*, *the policy optimization error*, diminishes to encourage fidelity to the ground-truth γ_0 and approaches 0 when $\widehat{\gamma} \to \gamma_0$. Consequently, these opposing error terms imply that an intermediate value of $\widehat{\gamma}$ yields a better reward function that induces the most expert-like policy.

We build up intermediate theorems and lemmas to formally prove Theorem 4.1 in the remaining subsections. The overall strategy is the following: The overall bound in Theorem 4.1 measures the performance gap between the policy induced by the learned $(\widehat{R}, \widehat{\gamma})$ pair and the optimal policy. This gap arises from two sources: (i) differences in the reward functions, and (ii) differences in the horizons over which the policies are optimized. Therefore, we decompose the overall bound into two error terms: (i) the *reward function estimation error* resulting from limited expert data during IRL, and (ii) the *policy optimization error* due to using an effective horizon different from the ground truth.

After deriving bounds for both error terms, we combine them to prove Theorem 4.1 in Section 4.6. The second term is straightforward to bound (see Section 4.6), The main challenge lies in bounding the reward function estimation error in the first term (Sections 4.5 to 4.4). We summarize our strategy on bounding the first error term—reward function estimation error—in terms of the effective horizon $\widehat{\gamma}$ and the number of expert state-action pairs N below.

IRL methods learn the reward function by minimizing the discrepancy between the induced policy and the expert data. Therefore, we bound the reward function estimation error using the *expert policy estimation error*, which measures the gap between the policy induced by the learned reward function and the ground-truth expert policy (see Sections 4.2 and 4.3). To do this, we first establish a correspondence between the feasible reward function set and an expert policy (Lemma 4.3 in Section 4.2). Then, we use this correspondence to bound the error in reward function estimation by the expert policy estimation error (Theorem 4.4 in Section 4.3). Next, we show that, this expert policy estimation error, in turn, depends on the complexity of the policy class $|\Pi_{\widehat{\gamma}}|$ and the number of expert pairs N used to fit it (see Section 4.4). Furthermore, we prove that the policy class complexity $|\Pi_{\widehat{\gamma}}|$ is controlled by the effective horizon $\widehat{\gamma}$: as $\widehat{\gamma}$ increases, the complexity of the policy class $\Pi_{\widehat{\gamma}}$ rises (Theorem 4.5 in Section 4.5). By combining these results, we derive the bound for the reward function estimation error in terms of $\widehat{\gamma}$ and N in Section 4.4. This completes the proof.

In the remainder of this section, we provide the detailed proofs, following the structure outlined above.

4.2 Feasible Reward Function Set

In this section, we establish a correspondence between the given expert demonstrations and the *feasible reward function set*—the set of all pairs of reward functions and horizons consistent with the expert data. This correspondence is essential for later bounding the reward function estimation error by the expert policy estimation error.

To create an algorithm-agnostic mapping from the fixed set of expert data and the effective horizon to the learned reward functions, we extend Metelli et al.'s [21] definition of feasible reward function sets—originally formulated for known discount factors—to variable and unknown effective discount factors. We begin by implicitly defining the feasible reward function set based on the foundational IRL formulation [23], which includes all pairs of reward-horizon whose induced policies match the expert data. From this implicit definition, we derive an explicit characterization of the feasible reward function set as a function of the expert policy.

We start by first implicitly defining the feasible reward set based on the IRL definition [23], adapting for the varying discount factors.

Definition 4.2 (IRL Problem, adapted to the setting of varying discount factors)**.** Let $\mathcal{M} = (S, A, P)$ be the MDP without the reward function or discount factor. An IRL problem, denoted as $\mathfrak{R} = (\mathcal{M}, \pi^E)$, consists of the MDP and an expert's policy π^E. A reward $\widehat{R} \in \mathbb{R}^{S\times A}$ is feasible for $\mathfrak{R}$ if there exists a $\widehat{\gamma}$ such that π^E is optimal for the MDP $\mathcal{M} \cup (\widehat{R}, \widehat{\gamma})$, *i.e.*, $\pi^E \in \Pi^*_{\widehat{R},\widehat{\gamma}}$. We use $\mathcal{R}_{\mathfrak{R}}$ to denote the set of feasible rewards for $\mathfrak{R}$.

This IRL formulation implies an implicit correspondence between the expert policy π^E and a *feasible* $(\widehat{R}, \widehat{\gamma})$ pair through their advantage function $A^{\pi^E}_{\widehat{R},\widehat{\gamma}}(s,a) = Q^{\pi^E}_{\widehat{R},\widehat{\gamma}}(s,a) - V^{\pi^E}_{\widehat{R},\widehat{\gamma}}(s)$. Specifically, the optimal policy induced by $(\widehat{R}, \widehat{\gamma})$ matches π^E when the following two conditions on the advantage function are met:

(1) if $\pi^E(a|s) > 0$, then $A^{\pi^E}_{\widehat{R},\widehat{\gamma}}(s,a) = 0$,

(2) if $\pi^E(a|s) = 0$, then $A^{\pi^E}_{\widehat{R},\widehat{\gamma}}(s,a) \leq 0$.

The first condition ensures the expert's chosen actions have zero advantage, while the second guarantees unchosen actions have non-positive advantages.

Next, we establish an explicit correspondence between the estimated $(\widehat{R}, \widehat{\gamma})$ and their compatible expert policy π^E by enforcing these two conditions on the advantage function $A^{\pi^E}_{\widehat{R},\widehat{\gamma}}$. To do this, we introduce two operators for any given policy π:

1. the *expert-filter*: $(B^\pi A)(s,a) = A(s,a)\mathbb{1}\{\pi(a|s) > 0\}$, that retains the advantage $A(s,a)$ values for actions taken by the expert policy $\pi^E(a|s)$,
2. the *expert-filter-complement*: $(\bar{B}^\pi A)(s,a) = A(s,a)\mathbb{1}\{\pi(a|s) = 0\}$, that preserves the advantage values for actions **not** taken by the expert.

Using the Bellman equation, we express the advantage functions in terms of the estimated $(\widehat{R}, \widehat{\gamma})$. We then apply the twofi lters to the advantage function $A^{\pi^E}_{\widehat{R},\widehat{\gamma}}$ to enforce the optimality conditions, simplifying the expression in the process. This allows us to derive the explicit expression for the feasible reward set presented in Lemma 4.3. The detailed derivation is provided in Appendix B.1.

Lemma 4.3 (Feasible Reward Function Set, extended from Metelli et al. [21]). *Let $\mathfrak{R} = (\mathcal{M}, \pi^E)$ be an IRL problem. Let $\widehat{R} \in \mathbb{R}^{S\times A}$ and $0 < \widehat{\gamma} < 1$, then $\widehat{R}$ is a feasible reward,* i.e., *$\widehat{R} \in \mathcal{R}_{\mathfrak{R}}$ if and only if there exists $\zeta \in \mathbb{R}^{S\times A}_{\geq 0}$ and $V \in \mathbb{R}^S$ such that:*

$$\widehat{R} = -\bar{B}^{\pi^E}\zeta + (E - \widehat{\gamma}P)V,$$

whereas $E : \mathbb{R}^{|S|} \to \mathbb{R}^{|S|\times|A|}$ is an operator that marginalizes the action for each state on a function $f(\cdot)$ such that $(Ef)(s,a) = f(s)$.

The feasible reward function in Lemma 4.3 comprises two terms: thefi rst term depends on the expert policy π^E and the second term depends on the underlying MDP's transition function. Thefirst term, $-\bar{B}^{\pi^E}\zeta$, is derived using the *expert-filter-complement* on a non-negative function ζ. This ensures that actions taken by the expert (i.e., $\pi^E(a|s) > 0$) are assigned a value of zero, while actions not taken (i.e., $\pi^E(a|s) = 0$) have non-positive values. The second term represents the policy's temporal effect that relies on the MDP's transition function. This can be viewed as reward shaping through the value function, which preserves the expert policy's optimality.

We have thus established an explicit correspondence between the feasible $(\widehat{R}, \widehat{\gamma})$ and the expert policy π^E in Lemma 4.3. This explicit expression will later be used to bound the difference in reward functions by the expert policy estimation error.

4.3 Reward Function Estimation Error from Expert Policy Estimation Error

In this section, we establish a bound on the reward function estimation error in terms of the *expert policy estimation error*—the discrepancy between the estimated expert policy (from limited data) and the true expert policy. Since IRL methods learn the reward function by matching the induced policy to expert data, establishing this bound is crucial, as it allows us to relate the reward estimation error to the amount of expert data and the induced policy complexity in later analysis.

Let's consider two IRL problems, $\mathfrak{R} = (\mathcal{M}, \pi^E)$ and $\hat{\mathfrak{R}} = (\mathcal{M}, \hat{\pi}^E)$, which differ only in expert policies: $\mathfrak{R}$ utilizes the ground-truth expert policy, while $\hat{\mathfrak{R}}$ employs an estimated policy from samples. Since an IRL algorithm aligns its induced policy with the estimated expert policy, its feasible sets will be equivalent to that of the estimated expert policy. Intuitively, inaccuracies in estimating the expert policy π^E lead to errors in estimating the feasible sets $\mathcal{R}_{\mathfrak{R}}$. Our goal is to obtain a reward function $\widehat{R}$ with a feasible set "close" to the ground-truth R_0's feasible set. Specifically, "closeness" is determined by the distance between the nearest reward functions in each set. The estimated $\mathcal{R}_{\hat{\mathfrak{R}}}$ is considered close to the exact $\mathcal{R}_{\mathfrak{R}}$ if, for every reward $R_0 \in \mathcal{R}_{\mathfrak{R}}$, there exists an estimated reward $\widehat{R} \in \mathcal{R}_{\hat{\mathfrak{R}}}$ with a small $|R_0 - \widehat{R}|$ value.

Given the form of the feasible reward functions corresponding to an expert policy as derived in Lemma 4.3, we express the estimation error $|R_0 - \widehat{R}|$ as a function of the ground-truth expert policy π^E and the estimated expert policy $\hat{\pi}^E$ from limited data. The bound on the reward function estimation error is shown below.

Theorem 4.4 (Extension of Theorem 3.1 in Metelli et al. [21]). *Let $\mathfrak{R} = (\mathcal{M}, \pi^E)$ and $\hat{\mathfrak{R}} = (\mathcal{M}, \hat{\pi}^E)$ be two IRL problems. Then for any $R_0 \in \mathcal{R}_{\mathfrak{R}}$ such that $R_0 = -\bar{B}^{\pi^E}\zeta + (E - \gamma_0 P)V$ and $\|R_0\|_\infty \leq R_{\max}$ there exist $\widehat{R} \in \mathcal{R}_{\hat{\mathfrak{R}}}$ and $0 < \widehat{\gamma} < 1$, such that element-wise it holds that:*

$$\left|R_0 - \widehat{R}\right| \leq \bar{B}^{\pi^E} B^{\hat{\pi}^E}\zeta.$$

Furthermore, $\|\zeta\|_\infty \leq \frac{R_{\max}}{1-\gamma_0}$.

The theorem above bounds the reward function estimation error by the discrepancy between the true expert policy π^E and the estimated expert policy $\hat{\pi}^E$ derived from limited data. Intuitively, it states that there exists a reward function $\widehat{R}$ in the estimated feasible set $\mathcal{R}_{\hat{\mathfrak{R}}}$ whose estimation error is controlled by the expert policy estimation error, under the corresponding $\widehat{\gamma}$. Specifically, the error term on the right-hand side is non-zero only for state-action pairs where $\pi^E(a \mid s) = 0$ but $\hat{\pi}^E(a \mid s) > 0$. This means the reward estimation error is zero for state-action pairs observed in the expert demonstrations, while errors arise where the estimated expert policy incorrectly assigns positive probability to actions the expert did not take. We refer readers to Appendix B.2 for the detailed proof.

4.4 Expert Policy Estimation Error from Limited Data

In this section, we derive a bound on the expert policy estimation error in terms of the amount of expert data N, the effective horizon $\widehat{\gamma}$, and the induced policy complexity $|\Pi_{\widehat{\gamma}}|$. We measure this error using the term $\bar{B}^{\pi^E} B^{\hat{\pi}^E}\zeta$, which quantifies how much the estimated policy $\hat{\pi}^E$ deviates from the true expert policy π^E.

Our derivation proceeds in three steps. First, we derive the expected value of the expert policy estimation error $\mathbb{E}[\bar{B}^{\pi^E} B^{\hat{\pi}^E}\zeta]$ under our estimation strategy. Then, we apply McDiarmid's inequality to obtain a probabilistic bound on how likely this error deviates from its expected value. Finally, we derive a uniform bound on the estimation error by substituting appropriate threshold values and simplifying the expression.

Expected Value of the Expert Policy Estimation Error. To estimate the expert policy $\hat{\pi}^E \in \{0,1\}^{|S|\times|A|}$ from N independent state-action samples, we aggregate the demonstrated pairs by summing their counts into $\widehat{\Pi}_N^E$. For each state s, we estimate $\hat{\pi}^E(s,a)$ by selecting the action a with the highest count:

$$\hat{a}_s = \arg\max_a \widehat{\Pi}_N^E(s,a).$$

If the count of $\hat{a}_s$ is uniquely the highest, we set $\hat{\pi}^E(s,a)$ to 1 for $\hat{a}_s$ and 0 for other actions. If no action has a unique maximum count, we set all entries to 0, indicating uncertainty. As the number of samples N increases, the estimated policy is more likely to reflect the expert's true decisions. For further details, see Appendix B.4.1.

Next, based on the estimation strategy described above, wefirst compute the expected value for the approximated expert policy $\mathbb{E}[\hat{\pi}^E]$ estimated from N expert samples, and in turn that of the estimation error $\mathbb{E}[\bar{B}^{\pi^E} B^{\hat{\pi}^E}\zeta]$.

Wefi rst compute the expected value of $\hat{\pi}^E(s,a)$ given N expert state-action pairs. The expected number state s is sampled across the N samples is $N \cdot \frac{1}{|S|}$. Moreover, since the expert policy is deterministic, the probability that a_s^* is the unique maximum at state s is the same as the probability that state s is sampled at least once. Therefore, the expectation of the estimated policy $\mathbb{E}[\hat{\pi}^E]$ for each state s is given by:

$$\mathbb{E}[\hat{\pi}^E(s,a)] = \begin{cases} 1 - \left(1 - \frac{1}{|S|}\right)^N & \text{for } a = a_s^*, \\ 0 & \text{otherwise.} \end{cases}$$

Next, we compute the expected value of the expert policy estimation error $\mathbb{E}[\bar{B}^{\pi^E} B^{\hat{\pi}^E}\zeta]$. Recall the operator $\bar{B}^{\pi^E}$ retains the values of $\zeta(s,a)$ for actions a that are not part of the expert policy. The true expert policy π^E deterministically selects action a_s^* for each state s, so the expert-filter complement will only retain the values for actions $a \neq a_s^*$. Thus, after applying the expert-filter complement, the expected value becomes:

$$\mathbb{E}[\bar{B}^{\pi^E} B^{\hat{\pi}^E}\zeta(s,a)] = \mathbb{E}[B^{\hat{\pi}^E}\zeta(s,a)] \cdot \mathbb{E}[\mathbb{1}\{\pi^E(a|s) = 0\}].$$

Moreover, the operator $B^{\hat{\pi}^E}$ preserves the function values for actions that are chosen by the estimated policy $\hat{\pi}^E(s,a)$. Thus, the expected value of $B^{\hat{\pi}^E}\zeta(s,a)$ is:

$$\mathbb{E}[B^{\hat{\pi}^E}\zeta(s,a)] = \zeta(s,a) \cdot \mathbb{E}[\mathbb{1}\{\hat{\pi}^E(a|s) > 0\}].$$

Simplifying the expressions above, we have $\mathbb{E}[\bar{B}^{\pi^E} B^{\hat{\pi}^E}\zeta(s,a)] = 0$, since $\pi^E(a|s) = 0$ for all $a \neq a_s^*$. Intuitively, an expected expert policy estimation error of 0 implies that, on average, the estimated policy $\hat{\pi}^E$ correctly matches the expert policy π^E. This means it does not assign non-zero probabilities to actions the expert would not take, indicating no systematic bias in the estimation process.

Applying McDiarmid's Inequality. Next, we obtain a probabilistic bound on how likely the expert policy estimation error deviates from its expected value $\mathbb{E}[\bar{B}^{\pi^E} B^{\hat{\pi}^E}\zeta(s,a)]$. We do so by applying McDiarmid's Inequality to the term $\bar{B}^{\pi^E} B^{\hat{\pi}^E}\zeta$. Since ζ is bounded by $\frac{R_{\max}}{1-\widehat{\gamma}}$, altering any single state-action sample affects at most one row of the estimated policy $\hat{\pi}^E(s,:)$, which leads to a bounded change in the value of $\bar{B}^{\pi^E} B^{\hat{\pi}^E}\zeta$ by at most $\frac{R_{\max}}{1-\widehat{\gamma}}$. This satisfies the bounded difference condition necessary for McDiarmid's Inequality.

By applying McDiarmid's Inequality, we derive that the probability of the expert policy estimation error exceeding a threshold t is bounded by:

$$\Pr\left(\bar{B}^{\pi^E} B^{\hat{\pi}^E}\zeta(s,a) \geq t\right) \leq \exp\left(\frac{-2Nt^2(1-\widehat{\gamma})^2}{R_{\max}^2}\right).$$

This bound quantifies how likely this estimation error exceeds a threshold t. As the number of expert-demonstrated state-action pairs N increases, the probability of significant deviations from the true policy decreases exponentially.

Uniform Bound on the Expect Policy Estimation Error. To obtain a uniform bound on the expert policy estimation error across all (s,π) pairs, we apply the union bound and set the right-hand side of the bound to $\frac{\delta}{|S||\Pi_{\widehat{\gamma}}|}$. Solving for t, we derive the threshold that bounds the error with probability at least $1-\delta$:

$$t = \frac{R_{\max}}{1-\widehat{\gamma}}\sqrt{\frac{1}{2N}\ln\left(\frac{|S||\Pi_{\widehat{\gamma}}|}{\delta}\right)}.$$

Since the reward function estimation error is bounded by the expert policy estimation error, $\left|R_0 - \widehat{R}\right| \leq \bar{B}^{\pi^E} B^{\hat{\pi}^E}\zeta$, the error bound threshold t applies to the reward function estimation error with probability at least $1-\delta$. As the amount of expert data N increases, the estimation error decreases, reflecting improved accuracy with more data. Moreover, the effective horizon $\widehat{\gamma}$ plays an important role: smaller values of $\widehat{\gamma}$ reduce both $\frac{1}{1-\widehat{\gamma}}$ and $|\Pi_{\widehat{\gamma}}|$, tightening the bound and better controlling the estimation error.

4.5 Policy Class Complexity Increases with $\widehat{\gamma}$

In the previous section, we bounded the reward estimation error in terms of the policy class complexity $|\Pi_{\widehat{\gamma}}|$ and the effective horizon $\widehat{\gamma}$. In fact, this policy class complexity also depends on the horizon that induces it. In this section, we examine how $\widehat{\gamma}$ influences policy complexity and prove that, for the policy class in Definition 3.1, this complexity grows monotonically with the discount factor $\widehat{\gamma}$. Consequently, we could deduce how changes in $\widehat{\gamma}$ affect the overall reward estimation error. The full proof on the monotonicity is provided in Appendix A.

Theorem 4.5. *Consider afi xed MDP $M = (S, A, P, \cdot, \cdot)$ with state space S, action space A, and transition function P. For any reward function $R \in \mathcal{F}_R$ that satisfies the mild condition in Definition 3.1* [1]*, we have the following claims about its policy class:*

(1) $\forall \widehat{\gamma}, \gamma' \in [0,1)$, if $\widehat{\gamma} < \gamma'$, then $\Pi_{\widehat{\gamma}} \subseteq \Pi_{\gamma'}$.
(2) When $\widehat{\gamma} = 0$, $|\Pi_0| = 1$.
(3) If $\widehat{\gamma} \to 1$, $|\Pi_{\widehat{\gamma}}| \geq (|A|-1)^{|S|-1}|S|$ under mild conditions.

Intuitively, claim 1 asserts that as the discount factor $\widehat{\gamma}$ grows, the number of potentially optimal policies increases monotonically. Together with claim 2 and 3, Theorem 4.5 demonstrates that there is a steep increase in policy complexity associated with the increment of $\widehat{\gamma}$. Specifically, when the discount factor is at its lowest ($\widehat{\gamma} = 0$), there is only one optimal policy, as the reward function has a

[1] This reward function form is not overly restrictive, as it merely excludes reward functions that assign equal rewards to multiple actions for any given state. In systems where the true reward $R \notin \mathcal{F}_R$, we can always construct a policy-invariant $R' \in \mathcal{F}_R$. We assume this specific form of reward function to ensure that discussions about the policy class remain meaningful, as any policy could be considered optimal when arbitrary reward functions are allowed.

unique maximum state-action pair for each state. However, as $\widehat{\gamma}$ increases and approaches the largest value ($\widehat{\gamma} \to 1$), the optimal policy class can eventually encompass nearly all possible policies, with $|\Pi_{\widehat{\gamma}}| = (|A| - 1)^{|S|-1}|S|$. In essence, $\widehat{\gamma}$ effectively controls the complexity of the policy class.

4.6 Error Decomposition and Deriving the Overall Bound

In this section, we use the reward function estimation error bound from earlier to establish thefi nal bound on $\left\|V_{R_0,\gamma_0}^{\pi^*_{R_0,\gamma_0}} - V_{R_0,\gamma_0}^{\pi^*_{\widehat{R},\widehat{\gamma}}}\right\|_\infty$ presented in Theorem 4.1. Recall that $\pi^*_{R_0,\gamma_0} \in \Pi_{\gamma_0}$ is the optimal policy derived from the ground-truth parameters (R_0, γ_0), while $\pi^*_{\widehat{R},\widehat{\gamma}} \in \Pi_{\widehat{\gamma}}$ is the optimal policy derived from the learned parameters $(\widehat{R}, \widehat{\gamma})$. We aim to bound the difference between their value functions evaluated under the ground-truth R_0 and γ_0.

To simplify the analysis, we decompose the overall error into two manageable terms: thefi rst accounts for the difference in value functions caused by the reward function estimation error, $\left\|R_0 - \widehat{R}\right\|_\infty$; the second accounts for the value difference due to optimizing the policy under different horizons (γ_0 versus $\widehat{\gamma}$).

Theorem 4.6 (Value Function Difference Bound). *Let $\mathcal{M} = (S, A, P)$ be a partial Markov Decision Process shared by two MDPs. Let $R_0 : S \times A \to [0, R_{\max}]$ be the ground-truth reward function, with $\gamma_0 \in (0, 1)$ as the ground-truth discount factor. Let $\widehat{R} : S \times A \to [0, R_{\max}]$ and $\widehat{\gamma} \in (0, 1)$ be the estimated reward function and discount factor obtained from data. Consider the optimal policies $\pi^*_{R_0,\gamma_0}$ and $\pi^*_{\widehat{R},\widehat{\gamma}}$ induced by (R_0, γ_0) and $(\widehat{R}, \widehat{\gamma})$, respectively. Then, the difference between their value functions, evaluated using the ground-truth reward R_0 and discount factor γ_0, is bounded above by*

$$\left\|V_{R_0,\gamma_0}^{\pi^*_{R_0,\gamma_0}} - V_{R_0,\gamma_0}^{\pi^*_{\widehat{R},\widehat{\gamma}}}\right\|_\infty \leq \frac{2}{1-\widehat{\gamma}}\left\|R_0 - \widehat{R}\right\|_\infty + \frac{|\gamma_0 - \widehat{\gamma}|}{(1-\gamma_0)(1-\widehat{\gamma})}R_{\max}.$$

Proof of Theorem 4.6 is in Appendix B.3.

Recall that we have established a bound on the reward estimation error $\left\|R_0 - \widehat{R}\right\|_\infty$ in terms of the number of expert data N, the effective horizon $\widehat{\gamma}$, and the policy complexity $|\Pi_{\widehat{\gamma}}|$ in Section 4.4. Specifically, with probability at least $1 - \delta$, the reward function estimation error is bounded by $\frac{R_{\max}}{1-\widehat{\gamma}}\sqrt{\frac{1}{2N}\ln\left(\frac{|S||\Pi_{\widehat{\gamma}}|}{\delta}\right)}$. Substituting this bound into thefi rst error term on the right-hand side of Theorem 4.6, we obtain:

$$\left\|V_{R_0,\gamma_0}^{\pi^*_{R_0,\gamma_0}} - V_{R_0,\gamma_0}^{\pi^*_{\widehat{R},\widehat{\gamma}}}\right\|_\infty \leq \frac{2R_{\max}}{(1-\widehat{\gamma})^2}\sqrt{\frac{1}{2N}\ln\left(\frac{|S||\Pi_{\widehat{\gamma}}|}{\delta}\right)} + \frac{|\gamma_0 - \widehat{\gamma}|}{(1-\gamma_0)(1-\widehat{\gamma})}R_{\max}.$$

This completes the proof of Theorem 4.1. The overall error consists of two terms. Thefi rst term arises from the reward function estimation error, which increases with larger $\widehat{\gamma}$. A larger $\widehat{\gamma}$ leads to a more complex induced policy class, making overfitting more likely when expert data is limited. Consequently, the reward function estimation error increases as the expert policy estimation error grows significantly. The second term represents the performance loss from using a smaller discount factor ($\widehat{\gamma} < \gamma_0$), and decreases as $\widehat{\gamma}$ approaches γ_0. These opposing dependencies on $\widehat{\gamma}$ suggest that there exists an intermediate value $0 < \widehat{\gamma} < \gamma_0$ that minimizes the overall loss. We will empirically demonstrate this in the following section.

5 EXPERIMENT

In this section, we empirically examine Theorem 4.1 by exploring how discount factors influence IRL performance with varying amounts of expert data. We adapt Linear Programming IRL (LP-IRL) [23] and Maximum Entropy IRL (MaxEnt-IRL) [34] to accommodate different discount factor settings and expert data sizes (implementation details are provided in Appendices D and E).

To jointly learn $(\widehat{R}, \widehat{\gamma})$, we incorporate cross-validation into our modified IRL methods to optimize the discount factor (details in Section 5.2). By evaluating policies across a range of $\widehat{\gamma}$s, cross-validation allows us to directly observe how varying $\widehat{\gamma}$ affects IRL performance with different amounts of expert data. While specialized IRL methods exist for optimizing $\widehat{\gamma}$ [8, 9], we choose cross-validation because it not only identifies the optimal discount factor, but also reveals performance trends across various settings, which is crucial for validating the nuanced implications of our theoretical findings. Specifically, we answer the following questions through cross-validation:

Q.1 Can a lower $\widehat{\gamma} < \gamma_0$ improve IRL policy performance?
Q.2 How does $\widehat{\gamma}^*$ change with increasing expert data N?
Q.3 Is the cross-validation extension effective infinding $\widehat{\gamma}^*$?

We evaluate the performance of LP-IRL and MaxEnt-IRL on four Gridworld and Objectworld tasks with varying reward complexity. We use the number of incorrectly induced actions as a proxy for value estimation error in the overall error bound. This measure counts the number of states where the induced policy $\hat{\pi}^E$ differs from π^E in action selection. For Q.1, we measure the number of incorrectly induced actions under varying discounted factors and different amount of expert data. Ourfi ndings show that the optimal $\widehat{\gamma}$s across all expert amount are smaller than γ_0 for both algorithms. For Q.2, we plot how the optimal discount factors change as the number of expert data increases. The consistent U-shaped curves observed in all cases align with the anticipated overfitting effect implied by the second error term in Equation 4.1. For Q.3, we compare the performance of policies selected via cross-validation with the best policy learnable from the available expert data. Our results indicate that the discrepancy in performance is negligible for all tasks, demonstrating the effectiveness of cross-validation in selecting $\widehat{\gamma}^*$.

5.1 Task Setup

We design four tasks of varying complexity in reward functions: Gridworld-simple, Gridworld-hard, Objectworld-linear, and Objectworld-nonlinear, adapted from [18] and [23]. We illustrate each task instance in thefi rst three columns of Tables 1 and more details on the task specification are in Appendix C. The ground-truth discount factor is $\gamma_0 = 0.99$. All task MDPs are assumed to be ergodic, such that all states are reachable and interconnected, allowing IRL

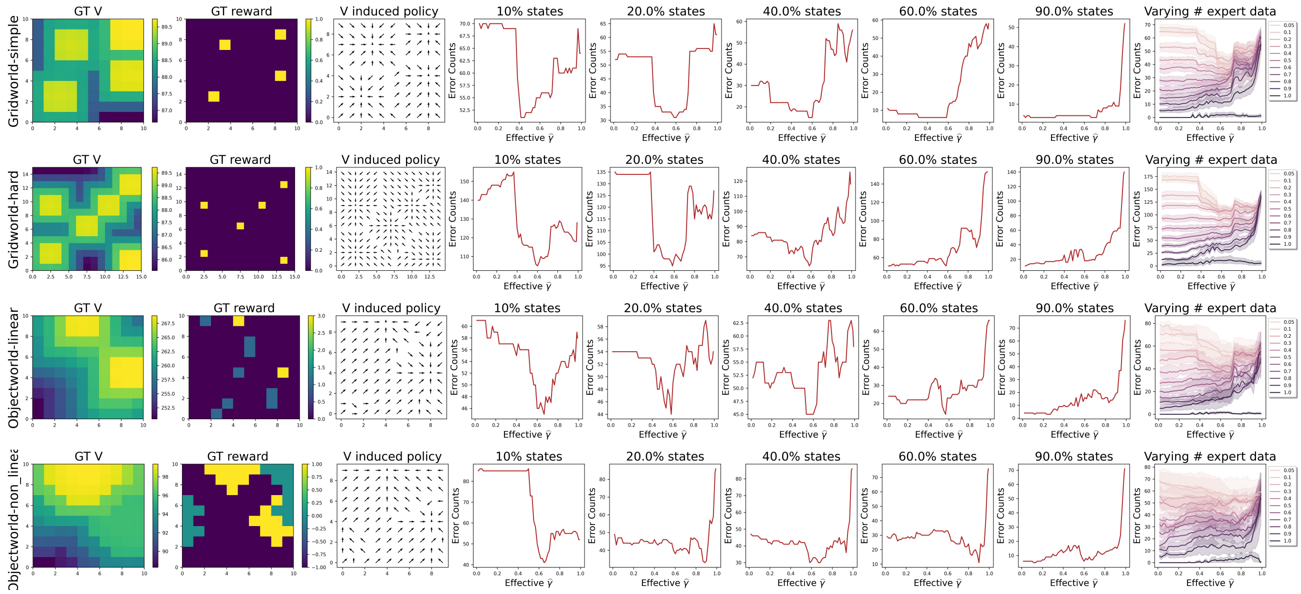

Figure 1: Summary of LP-IRL with varying discount factors across four tasks. The *error counts* measure the number of states for which a policy's action selection deviates from the expert's actions. Each task displays the ground-truth value function (column 1), reward function (column 2), expert policy (column 3), error count curves for different amount of expert data in a single instance (columns 4-8), and the error count curve summary for a batch of 10 MDPs across varying amount of expert data (column 9). In all four tasks, $\gamma_0 = 0.99$. The optimal discount factor $\widehat{\gamma}^* < \gamma_0$ for varying amount of expert data. MaxEnt-IRL has similar curves in Figure 4.

methods to estimate rewards and policies for all states from limited expert data by propagating values from observed states during value iteration.

The Gridworld tasks provide sparse rewards only at randomly sampled goals: Gridworld-simple has fewer goals (4) and a smaller state space (10×10 states), while Gridworld-hard has more goals (6) and a larger state space (15×15 states). On the other hand, the Objectworld tasks have denser ground-truth rewards that are functions of nearby object features. The reward function for Objectworld-linear is linear with respect to the features of nearby objects, while that of Objectworld-nonlinear is non-linear. Intuitively, learning a complex reward function may be more susceptible to overfitting, especially when expert data is sparse compared to the state space.

We vary the amount of expert data by adjusting the number of state-action pairs included, measuring this as a percentage relative to the state space size. Since states may be sampled multiple times in the trajectories, the total number of state-action pairs N can exceed the number of states $|S|$. Given expert trajectories $D = \{\tau_0, \tau_1, \dots\}$, we define the percentage of expert data as $K\% = \frac{N}{|S|} \times 100\%$. We evaluate the performance of the induced policy by counting the number of states where it selects a different action from the expert policy—referred to as the *state error count*.

5.2 Cross Validation Extension

We use cross-validation to determine the optimal $\widehat{\gamma}^*$ from the expert data D containing N state-action pairs. We split D into non-overlapping training (80%) and validation (20%) sets. We uniformly sample $M = 20$ discount factors from the interval $(0, \gamma_0)$. For each sampled γ, we learn R_γ using the training set and evaluate the induced policy on the validation set by counting the number of states where it selects a different action from the expert policy—the *state error count*. The γ that minimizes this error count is selected as the optimal $\widehat{\gamma}$. For all tasks, we randomly sample 10 environments per task and report the mean and standard deviation of errors. To assess the effectiveness of cross-validation, we employ an oracle representing the best policy learnable from all available expert data. This oracle, considered "cheating," uses both the training and validation sets for training and validates on the entire state space

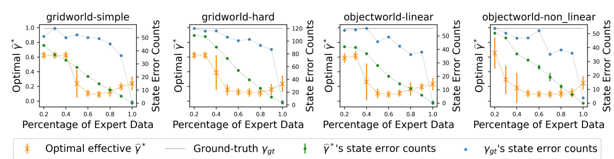

Figure 2: Optimal $\widehat{\gamma}^*$ for LP-IRL at varying amount of expert data. For each task, we select $\widehat{\gamma}^*$ for all 10 sampled environments through cross-validation. The orange curves illustrate how the optimal discount factor $\widehat{\gamma}^*$ changes with the amount of expert data, while the green curves show the corresponding error counts. The ground-truth $\gamma_0 = 0.99$ is depicted in grey, with its error counts displayed in blue. As the amount of expert data increases, $\widehat{\gamma}^*$ initially decreases sharply and then gradually increases, indicating that overfitting is prominent when expert data is scarce.

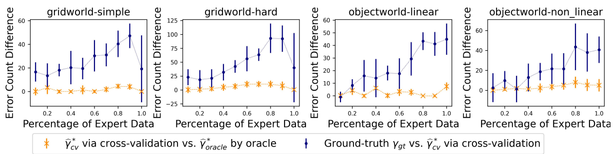

Figure 3: The cross-validation results for LP-IRL on four tasks are shown. The x-axis represents the amount of expert data; the y-axis shows policy error count differences. We compare discount factors $\widehat{\gamma}^*_{\mathrm{cv}}$ (learned from cross-validation) and $\widehat{\gamma}^*_{\mathrm{oracle}}$ (chosen by the oracle). Orange dots depict error differences between policies induced by $\widehat{\gamma}^*_{\mathrm{cv}}$ and $\widehat{\gamma}^*_{\mathrm{oracle}}$; blue dots show differences between policies induced by $\widehat{\gamma}^*_{\mathrm{cv}}$ and the ground-truth γ_0. The orange curves near zero indicate that cross-validation effectively selects $\widehat{\gamma}^*$, while the positive blue curves show that cross-validation consistently yields better policies than using γ_0.

(both observed and unobserved states). We use this oracle only to compare whether the $\widehat{\gamma}^*$ chosen by cross-validation corresponds to that of the oracle policy, not for selecting the optimal $\widehat{\gamma}$.

5.3 Results

We assess the impact of the *effective horizon* on IRL by evaluating LP-IRL and MaxEnt-IRL across four tasks. For simplicity, we treat LP-IRL's discount factor γ and MaxEnt-IRL's horizon T interchangeably, withfi ndings for γ also applying to T unless specified otherwise. Policy performance results are presented in Tables 1 (LP-IRL) and 4 (MaxEnt-IRL). We report *state error counts*, measuring discrepancies between induced and expert policies by counting the states where their action selections differ.

Q.1 Optimal $\widehat{\gamma}^*$ is Lower than Ground-Truth.

As shown in Figures 1 and 4, the optimal discount factor $\widehat{\gamma}^* < \gamma_0$ for all four tasks and across various amounts of expert data in both LP-IRL and MaxEnt-IRL. With limited expert data, the error count curves are generally U-shaped: discrepancies with the expert policy decrease as $\widehat{\gamma}$ increases to a "sweet spot" and then rise sharply. This pattern confirms our error bounds in Theorem 4.1: for small $\widehat{\gamma}$, the overfitting-related error (second term in Equation 4.1) is less significant, and increasing $\widehat{\gamma}$ allows temporal extrapolation, reducing the overall error. However, beyond the optimal $\widehat{\gamma}$, overfitting becomes more pronounced, and the overall error increases as thefi rst error term outweighs the benefits.

With abundant expert data, error counts remain low (in LP-IRL) or initially decrease (in MaxEnt-IRL) for small $\widehat{\gamma}$ and then increase as $\widehat{\gamma}$ grows, indicating that $\widehat{\gamma}^* < \gamma_0$ yields the most expert-like policy, confirming our theoretical results. Interestingly, in LP-IRL, the error counts do not initially drop as $\widehat{\gamma}$ increases. This is because, with dense expert data, LP-IRL accurately matches step-wise behaviors, making performance gains from temporal reasoning negligible. This observation supports Spencer et al. [31]'s insight that naive behavioral cloning can outperform IRL algorithms when ample expert data is available. In contrast, MaxEnt-IRL does not exhibit low error counts for small $\widehat{\gamma}$ because its reward function is parameterized linearly in state features, limiting its ability to precisely replicate actions even with abundant data.

Q.2 Optimal $\widehat{\gamma}^*$s Vary with the Amount of Expert Data.

Figures 2 and 5 show how the optimal discount factor $\widehat{\gamma}^*$ varies with increasing amounts of expert data for LP-IRL and MaxEnt-IRL, respectively. When data is scarce, $\widehat{\gamma}^*$ is high because the benefits of temporal reasoning outweigh overfitting concerns—the estimation error of the expert policy remains high regardless of overfitting. A large $\widehat{\gamma}^*$ enables extrapolation of actions to nearby unobserved states, reducing thefi rst error term in Theorem 4.1 and improving overall error reduction.

With a moderate amount of data, $\widehat{\gamma}^*$ decreases: although expert policy estimation improves with more data, it is still limited. The overall error bound favors a smaller $\widehat{\gamma}^*$ to mitigate overfitting, reducing the second error term even at the cost of some temporal reasoning benefits. As data increases further, $\widehat{\gamma}^*$ rises again since overfitting becomes less significant, and larger values enhance temporal reasoning.

Overall, as expert data increases, error counts for $\widehat{\gamma}^*$ strictly decrease and remain below those for γ_0, indicating that $\widehat{\gamma}^*$ enables IRL to learn more effectively from additional expert data.

Q.3 Cross-Validation Effectively Selects Optimal $\widehat{\gamma}^*$s.

Figures 3 and 6 summarize the cross-validation results for LP-IRL and MaxEnt-IRL, respectively. The performance discrepancy between policies induced by $\widehat{\gamma}^*_{\mathrm{cv}}$ and the oracle $\widehat{\gamma}^*_{\mathrm{oracle}}$ (orange curves) is consistently near zero across all four tasks, indicating that cross-validation effectively selects $\widehat{\gamma}^*$ similar to the oracle. Moreover, the blue curves represent the error differences between policies induced by the ground-truth γ_0 and $\widehat{\gamma}^*_{\mathrm{cv}}$, which are significantly higher than zero. This suggests that $\widehat{\gamma}^*_{\mathrm{cv}}$ yields better-performing policies than using the ground-truth γ_0, confirming our theoreticalfi ndings in Theorem 4.1.

6 CONCLUSION

In this paper, we present a theoretical analysis on IRL that unveils the potential of a reduced horizon in inducing a more expert-like policy, particularly in data-scarce situations. Ourfi ndings reveals an important insight on role of the horizon in IRL: it controls the complexity of the induced policy class, therefore reduces overfitting to the limited expert data. We, therefore, propose a more natural IRL function class that jointly learns reward-horizon pairs and empirically substantiate our analysis using a cross-validation extension for the existing IRL algorithms. As overfitting remains a challenge for IRL, especially with scarce expert data, we believe ourfindings offer valuable insights for the IRL community on better IRL formulations.

ACKNOWLEDGMENTS

This research is supported in part by the National Research Foundation (NRF), Singapore and DSO National Laboratories under the AI Singapore Program (AISG Award No: AISG2-RP-2020-016), the National University of Singapore (AcRF Tier-1 grant A-8002616-00-00), the National Science Foundation (Grant No. IIS-2007076). Any opinions,fi ndings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of NRF Singapore or National Science Foundation.

REFERENCES

[1] Pieter Abbeel and Andrew Ng. 2004. Apprenticeship Learning via Inverse Reinforcement Learning. In *Proc. Int. Conf. on Machine Learning*.

[2] Ron Amit, Ron Meir, and Kamil Ciosek. 2020. Discount factor as a regularizer in reinforcement learning. In *Proc. Int. Conf. on Machine Learning*.

[3] Abdeslam Boularias, Jens Kober, and Jan Peters. 2011. Relative Entropy Inverse Reinforcement Learning. In *Proc. Int. Conf. on Artificial Intelligence & Statistics*.

[4] Chelsea Finn, Paul Christiano, Pieter Abbeel, and Sergey Levine. 2016. A Connection Between Generative Adversarial Networks, Inverse Reinforcement Learning, and Energy-Based Models. In *Advances in Neural Information Processing Systems*.

[5] Chelsea Finn, Sergey Levine, and Pieter Abbeel. 2016. Guided Cost Learning: Deep Inverse Optimal Control via Policy Optimization. In *Proc. Int. Conf. on Machine Learning*.

[6] Justin Fu, Katie Luo, and Sergey Levine. 2018. Learning Robust Rewards with Adversarial Inverse Reinforcement Learning. In *Proc. Int. Conf. on Learning Representations*.

[7] Seyed Kamyar Seyed Ghasemipour, Richard Zemel, and Shixiang Gu. 2019. A Divergence Minimization Perspective on Imitation Learning Methods. In *Proc. Conference on Robot Learning*.

[8] Gaurav R Ghosal, Matthew Zurek, Daniel S Brown, and Anca D Dragan. 2023. The effect of modeling human rationality level on learning rewards from multiple feedback types. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

[9] Babatunde H Giwa and Chi-Guhn Lee. 2021. Estimation of Discount Factor in a Model-Based Inverse Reinforcement Learning Framework. In *Bridging the Gap Between AI Planning and Reinforcement Learning Workshop at ICAPS*.

[10] Xin Guo, Anran Hu, and Junzi Zhang. 2022. Theoretical guarantees offictitious discount algorithms for episodic reinforcement learning and global convergence of policy gradient methods. In *Proc. AAAI Conf. on Artificial Intelligence*.

[11] Jonathan Ho and Stefano Ermon. 2016. Generative Adversarial Imitation Learning. In *Advances in Neural Information Processing Systems*.

[12] Hana Hoshino, Kei Ota, Asako Kanezaki, and Rio Yokota. 2022. OPIRL: Sample Efficient Off-Policy Inverse Reinforcement Learning via Distribution Matching. In *Proc. IEEE Int. Conf. on Robotics & Automation*.

[13] Hao Hu, Yiqin Yang, Qianchuan Zhao, and Chongjie Zhang. 2022. On the role of discount factor in offline reinforcement learning. In *Proc. Int. Conf. on Machine Learning*.

[14] Nan Jiang, A. Kulesza, S. Singh, and R. Lewis. 2015. The Dependence of Effective Planning Horizon on Model Accuracy. In *Proc. Int. Conf. on Autonomous Agents & Multiagent Systems*.

[15] Liyiming Ke, Sanjiban Choudhury, Matt Barnes, Wen Sun, Gilwoo Lee, and Siddhartha Srinivasa. 2020. Imitation Learning as f-Divergence Minimization. In *Algorithmic Foundations of Robotics XIV—Proc. Int. Workshop on the Algorithmic Foundations of Robotics (WAFR)*.

[16] Cassidy Laidlaw, Stuart J Russell, and Anca Dragan. 2023. Bridging rl theory and practice with the effective horizon. *Advances in Neural Information Processing Systems* 36 (2023), 58953–59007.

[17] Keuntaek Lee, David Isele, Evangelos A. Theodorou, and Sangjae Bae. 2022. Spatiotemporal Costmap Inference for MPC via Deep Inverse Reinforcement Learning. In *IEEE Robotics & Automation Letters*.

[18] Sergey Levine, Zoran Popovic, and Vladlen Koltun. 2011. Nonlinear Inverse Reinforcement Learning with Gaussian Processes. In *Advances in Neural Information Processing Systems*.

[19] James MacGlashan and Michael L. Littman. 2015. Between Imitation and Intention Learning. In *Proc. Int. Jnt. Conf. on Artificial Intelligence*.

[20] Jacob Mattingley, Yang Wang, and Stephen Boyd. 2011. Receding horizon control. *IEEE Control Systems Magazine* 31, 3 (2011), 52–65.

[21] Alberto Maria Metelli, Giorgia Ramponi, Alessandro Concetti, and Marcello Restelli. 2021. Provably Efficient Learning of Transferable Rewards. In *Proc. Int. Conf. on Machine Learning*.

[22] Andrew Y Ng, Daishi Harada, and Stuart Russell. 1999. Policy Invariance Under Reward Transformations: Theory and Application to Reward Shaping. In *Proc. Int. Conf. on Machine Learning*.

[23] Andrew Y. Ng and Stuart J. Russell. 2000. Algorithms for Inverse Reinforcement Learning. In *Proc. Int. Conf. on Machine Learning*.

[24] Tianwei Ni, Harshit Sikchi, Yufei Wang, Tejus Gupta, Lisa Lee, and Benjamin Eysenbach. 2020. f-IRL: Inverse Reinforcement Learning via State Marginal Matching. In *Proc. Conference on Robot Learning*.

[25] Takayuki Osa, Joni Pajarinen, Gerhard Neumann, J Andrew Bagnell, Pieter Abbeel, Jan Peters, et al. 2018. An Algorithmic Perspective on Imitation Learning. *Foundations and Trends® in Robotics* 7, 1-2 (2018), 1–179.

[26] Marek Petrik and Bruno Scherrer. 2008. Biasing approximate dynamic programming with a lower discount factor. In *Advances in neural information processing systems*.

[27] Matteo Pirotta and Marcello Restelli. 2016. Inverse Reinforcement Learning through Policy Gradient Minimization. In *Proc. AAAI Conf. on Artificial Intelligence*.

[28] Deepak Ramachandran and Eyal Amir. 2007. Bayesian Inverse Reinforcement Learning. In *Proc. Int. Jnt. Conf. on Artificial Intelligence*.

[29] Giorgia Ramponi, Gianluca Drappo, and Marcello Restelli. 2020. Inverse Reinforcement Learning from a Gradient-Based Learner. *Advances in Neural Information Processing Systems* 33 (2020), 2458–2468.

[30] Nathan D. Ratliff, J. Andrew Bagnell, and Martin A. Zinkevich. 2006. Maximum Margin Planning. In *Proc. Int. Conf. on Machine Learning*.

[31] Jonathan Spencer, Sanjiban Choudhury, Arun Venkatraman, Brian D. Ziebart, and J. Andrew Bagnell. 2021. Feedback in Imitation Learning: The Three Regimes of Covariate Shift. *ArXiv* abs/2102.02872 (2021).

[32] Markus Wulfmeier, Peter Ondruska, and Ingmar Posner. 2016. Maximum Entropy Deep Inverse Reinforcement Learning. In *Advances in Neural Information Processing Systems*.

[33] Yiqing Xu, Wei Gao, and David Hsu. 2022. Receding Horizon Inverse Reinforcement Learning. In *Advances in Neural Information Processing Systems*.

[34] Brian D. Ziebart, Andrew Maas, J. Andrew Bagnell, and Anind K. Dey. 2008. Maximum Entropy Inverse Reinforcement Learning. In *Proc. AAAI Conf. on Artificial Intelligence*.