UNDERSTANDING CONSTRAINT INFERENCE IN SAFETY-CRITICAL INVERSE REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

In practical applications, the underlying constraint knowledge is often unknown and difficult to specify. To address this issue, recent advances in Inverse Constrained Reinforcement Learning (ICRL) have focused on inferring these constraints from expert demonstrations. However, the ICRL approach typically characterizes constraint learning as a tri-level optimization problem, which is inherently complex due to its interdependent variables and multiple layers of optimization. Considering these challenges, a critical question arises: *Can we implicitly embed constraint signals into reward functions and effectively solve this problem using a classic reward inference algorithm?* The resulting method, known as Inverse Reward Correction (IRC), merits investigation. In this work, we conduct a theoretical analysis comparing the sample complexities of both solvers. Our findings confirm that the IRC solver achieves lower sample complexity than its ICRL counterpart. Nevertheless, this reduction in complexity comes at the expense of generalizability. Specifically, in the target environment, the reward correction terms may fail to guarantee the safety of the resulting policy, whereas this issue can be effectively mitigated by transferring the constraints via the ICRL solver. Advancing our inquiry, we investigate conditions under which the ICRL solver ensures ε-optimality when transferring to new environments. Empirical results across various environments validate our theoretical findings, underscoring the nuanced trade-offs between complexity reduction and generalizability in safety-critical applications.

1 INTRODUCTION

To solve decision-making problems in safety-critical applications, a crucial prerequisite is aligning the decision process with the underlying constraints in the environment. To realize this vision, existing safe Reinforcement Learning (RL) algorithms typically optimize a control policy based on a known or manually-specified constraint (Liu et al., 2021; Gu et al., 2022). However, in many real-world applications, the ground-truth constraints are often unknown. Moreover, given the inherent complexity of environmental dynamics, safety constraints must accurately model the interdependencies among numerous variables, which are difficult to capture solely with prior knowledge.

To resolve the above challenges, Inverse Constrained Reinforcement Learning (ICRL) designs a data-driven constraint inference method to learn the constraints from expert demonstrations (Scobee & Sastry, 2020). Specifically, an ICRL algorithm (Malik et al., 2021) typically addresses a tri-level optimization problem involving the update of 1) the feasibility functions to represent constraints, 2) the Lagrange parameters to balance reward maximization and constraint satisfaction, and 3) the policy function to guide the agent's behaviors. Under this setting, the variables subject to optimization are interdependent, and the sub-optimality of one variable can influence the performance of the others. To mitigate the complexity of this problem, Hugessen et al. (2024) proposed to simplify the ICRL solver by incorporating the impact of both constraint and Lagrange parameters into a reward correction term. This modification reduces the ICRL solver to a bi-level solver known as Inverse Reward Correction (IRC) (Li et al., 2023). Hugessen et al. (2024) empirically demonstrated that such simplification does not compromise the performance of constraint learning. These observations raise significant questions about the necessity of explicitly modeling constraints. It remains unclear whether the canonical reward learning framework is adequate for capturing an agent's preferences within a Constrained Markov Decision Process (CMDP).

To address this issue, in this work, we conduct a rigorous study to understand the impact of modeling constraints. Specifically, we theoretically and empirically compare the performance of IRC and ICRL from the following perspectives:

Training Efficiency. Unlike previous studies that primarily compared IRC and ICRL via empirical evaluations (Hugessen et al., 2024), we are the first to offer a theoretical quantification of training efficiency for both IRC and ICRL solvers in capturing the safety preferences in expert agents' decisions. Achieving this objective, however, presents an inherent challenge due to the unidentifiability of the optimal solution from expert demonstrations, making it an ill-posed problem (Ng et al., 2000). Therefore, rather than modeling point-wise solutions, we follow Metelli et al. (2021) and propose a theoretical framework by characterizing the feasible region. This approach allows us to calculate the sample complexity of both IRC and ICRL. Our results indicate that ICRL requires more training samples, which aligns with previous empirical findings in Hugessen et al. (2024). Besides, our theoretical results provide an in-depth analysis of the increased complexity. They demonstrate that ICRL solvers additionally capture constraint-violating movements by utilizing known reward signals (Quan et al., 2024). In the update of a CRL policy, these movements correspond to the decision-making patterns that induce an increase in the Lagrange parameters.

Cross-environment Transferability. While previous constraint learning methods primarily focus on imitating experts' behavior (Scobee & Sastry, 2020; Liu et al., 2023; Hugessen et al., 2024), an important motivation of inferring experts' safety preferences aims to generalize this knowledge to guide policy learning under similar contexts (Feng et al., 2023; Zhang et al., 2024). In this study, we compare IRC and ICRL in terms of guaranteeing *safety* and *optimality* of policies across different environments. We begin with an illustrative example to demonstrate situations where IRC could potentially induce unsafe behaviors. We identify and summarize the necessary conditions under which IRC leads to such unsafe behaviors and explain why the performance of ICRL is more robust by explicitly modeling constraints. Regarding the optimality of generalized policies, we derive the sub-optimality gap in transferring the constraint learned by ICRL. This result offers a theoretical guarantee for studying how mismatches in environmental dynamics and reward signals affect the acquisition of optimal policies based on learned constraints. In the absence of constraint satisfaction, such results can not be extended to IRC.

Contributions. We compare the training efficiency and cross-environment transferability of IRC and ICRL solvers to answer the critical question raised in the abstract.

- Following the theoretical framework from Metelli et al. (2021), we introduce the IRC solver (Definition 3.3) to overcome the limitation of the IRL solver, which lacks a mechanism to leverage existing reward signals and may not be compatible with different rewards. Furthermore, we analyze the sample complexity of IRC (Section 4.1) and compare it with existing results for ICRL (Yue et al., 2024), showing that IRC achieves lower sample complexity than ICRL under the same optimality criterion (Theorem 4.2).
- We conduct a formal study of transferability in safety to compare IRC and ICRL. We show that transferred cost functions by ICRL are guaranteed to preserve safety in overlapping critical regions (Lemma 5.2), whereas transferred reward correction terms by IRC can be offset by the difference in reward functions and transition dynamics between source and target environments. (Theorem 5.3).
- Extending the transferability definition from Schlaginhaufen & Kamgarpour (2024, Definition 3.1) from regular MDP settings to CMDP settings, we analyze the optimality of constraint knowledge inferred by ICRL in target environments. Specifically, we define the suboptimality gap under CMDP settings (Definition 5.5). Based upon this, we derive conditions that limit the similarity between source and target environments to ensure ε-optimality for ICRL (Theorem 5.7).
- Finally, we empirically validate our results on training efficiency and cross-environment transferability in various environments (Section 6).

2 Related Work

Inverse Constrained Reinforcement Learning (ICRL). Unlike IRL, which focuses primarily on recovering reward functions, ICRL seeks to align with expert agents' preferences by inferring the constraints they adhere to. The bulk of existing ICRL algorithms update cost functions by maximizing the likelihood of generating expert demonstrations under the maximum (casual) entropy framework (Scobee & Sastry, 2020). Subsequent works extended this approach from discrete to continuous state-action spaces (Malik et al., 2021; Liu et al., 2023; Baert et al., 2023; Qiao et al., 2023; Xu & Liu, 2024b; Quan et al., 2024). To improve training efficiency, Liu & Zhu (2022); Gaurav et al. (2023) combined ICRL with bi-level optimization techniques. Towards theoretical groundings of ICRL, Yue et al. (2024) recently proposed efficient constraint inference through exploration strategies with tractable sample complexity. However, these works primarily evaluate their performance by applying inferred constraints within the same environment used for learning. Although Xu & Liu (2024a) considered transition discrepancies for policies under the robust optimization framework, the challenge of transferring constraints to new environments remains largely unexplored.

Transferability in Inverse Reinforcement Learning (IRL). A significant application of IRL algorithms is to guide policy learning in similar environments. However, obtaining guarantees of transferability in unregularized settings is more challenging than in entropy-regularized contexts. To facilitate knowledge transfer in unregularied settings, Metelli et al. (2021) assumed that the feasible reward functions recovered remain valid in target environments, and Amin et al. (2017) reduced the dimension of the reward class to state-only rewards. In contrast, entropy-regularized settings have been explored more extensively. Cao et al. (2021) and Skalse et al. (2023) demonstrated that under entropy regularization, the expert's reward can be identified up to potential shaping transformations (Ng et al., 1999). In addition, Rolland et al. (2022) showed that to guarantee transferability across any transition laws, the expert's reward must be identified up to a constant. Building upon these insights, Cao et al. (2021) and Rolland et al. (2022) learned the reward function from multiple experts who shared rewards but had sufficiently different transition laws distinguished by a specific rank condition. Continuing this line of research, Schlaginhaufen & Kamgarpour (2024) extended this approach to an offline setting and derived a sufficient condition for transferability to local changes in the transition law when learning from a single expert. However, these methods focus on transferring reward functions in regular MDPs, without addressing the generalization of constraints in CMDPs.

3 Preliminaries and Problem Formulation

Notation. Let $\mathcal{X}$ be a finite set and $\mathcal{Y}$ be a space. The notation $\mathcal{Y}^{\mathcal{X}}$ represents the set of functions $f : \mathcal{X} \to \mathcal{Y}$. The probability measure over $\mathcal{X}$ is denoted as $\Delta^{\mathcal{X}} = \{\nu \in [0,1]^{\mathcal{X}} : \sum_{x\in\mathcal{X}} \nu(x) = 1\}$ and we denote $\Delta^{\mathcal{Y}}_{\mathcal{X}}$ as the set of functions $\mathcal{X} \to \Delta^{\mathcal{Y}}$. We define $\min^+_{x\in\mathcal{X}} f(x)$ returns the minimum positive value of f over $\mathcal{X}$. For a linear operator A, we denote its image by $\operatorname{im} A$. Let c^E represent the underlying cost function obeyed by the expert and μ^E denote the expert occupancy measure. Let $\mathbb{1}(\cdot)$ denote the indicator function. The expansion operator $E : \mathbb{R}^{\mathcal{S}} \to \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$ satisfies $(Ef)(s,a) = f(s)$. The complete notation is provided in Appendix Table 1.

Constrained Markov Decision Process (CMDP). The environment is modeled as a stationary CMDP $\mathcal{M}_c := (\mathcal{S}, \mathcal{A}, P_{\mathcal{T}}, r, c, \epsilon, \mu_0, \gamma)$, where $\mathcal{S}$ and $\mathcal{A}$ are the finite state and action spaces, with cardinalities $S = |\mathcal{S}|$ and $A = |\mathcal{A}|$; $P_{\mathcal{T}}(s'|s,a) \in \Delta^{\mathcal{S}}_{\mathcal{S}\times\mathcal{A}}$ defines the transition distribution; $r \in [0, R_{\max}]^{\mathcal{S}\times\mathcal{A}}$ and $c \in [0, C_{\max}]^{\mathcal{S}\times\mathcal{A}}$ denote the reward and cost functions; ϵ defines the threshold of the constraint; $\mu_0 \in \Delta^{\mathcal{S}}$ denotes the initial state distribution; and $\gamma \in [0,1)$ is the discount factor. The agent's behavior is modeled by a policy $\pi \in \Delta^{\mathcal{A}}_{\mathcal{S}}$. This work focuses on the stationary CMDP where the planning horizon H goes to infinity, and our theoretical results are mainly based on a discrete finite state-action space.

Value and advantage functions. We define the action value functions in a CMDP $\mathcal{M}_c$ for costs and rewards as $Q^{c,\pi}_{\mathcal{M}_c}$ and $Q^{r,\pi}_{\mathcal{M}_c}$. The superscript c or r specifies the actual costs or rewards evaluated. The reward action-value function is $Q^{r,\pi}_{\mathcal{M}_c}(s,a) = \mathbb{E}_{\pi,P_{\mathcal{T}}}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t,a_t)\right]$, and the reward advantage function follows $A^{r,\pi}_{\mathcal{M}_c}(s,a) = Q^{r,\pi}_{\mathcal{M}_c}(s,a) - V^{r,\pi}_{\mathcal{M}_c}(s)$, where the reward state-value function is $V^{r,\pi}_{\mathcal{M}_c}(s) = \mathbb{E}_{\pi}[Q^{r,\pi}_{\mathcal{M}_c}(s,a)]$. The same notation scheme applies to cost value functions in $\mathcal{M}_c$ by replacing the superscript r with c. It also applies to CMDP without knowing the cost, i.e., $\mathcal{M} = \mathcal{M}_c \backslash c$ by replacing the subscript $\mathcal{M}_c$ with $\mathcal{M}$.

Constrained Reinforcement Learning (CRL). Within a CMDP environment, CRL learns a policy π that maximizes the cumulative rewards subject to a known constraint:

$$\mathsf{CRL}(r,c) = \max_{\pi} \ \mathbb{E}_{\mu_0,\pi,P_{\mathcal{T}}}\Big[\sum_{t=0}^{\infty} \gamma^t r(s_t,a_t)\Big] \qquad \text{s.t. } \mathbb{E}_{\mu_0,\pi,P_{\mathcal{T}}}\Big[\sum_{t=0}^{\infty} \gamma^t c(s_t,a_t)\Big] \le \epsilon, \tag{PI}$$

where $\epsilon > 0$ indicates a soft constraint and $\epsilon = 0$ represents a hard constraint.

Inverse Constraint Inference (ICI). In many practical applications, constraints are not readily available, requiring us to infer the constraints followed by expert agents based on their behavior. A common solver for this ICI problem under the RL setting is known as Inverse Constrained Reinforcement Learning (ICRL), which can be formally defined as follows:

Definition 3.1. (*ICRL solver for constraint inference (Malik et al., 2021)*). An ICRL solver is denoted as $\mathbb{S}_{\text{ICRL}}(\mathcal{M}, \pi^E, r)$, where $\mathcal{M}$ is a $\mathcal{M}_c \backslash c$ (CMDP without knowing the cost) and $\pi^E \in \Delta_{\mathcal{S}}^{\mathcal{A}}$ is an expert's policy. A cost representation c is a *feasible* solution from $\mathbb{S}_{\text{ICRL}}$ if and only if π^E is an optimal policy for the CMDP $\mathcal{M} \cup c$ (CMDP with c as costs). We denote by $\mathcal{C}_{\mathbb{S}_{\text{ICRL}}}$ the set of feasible cost functions derived by $\mathbb{S}_{\text{ICRL}}$.

Previous ICRL solvers (Scobee & Sastry, 2020; Malik et al., 2021) explicitly model constraints and infer the cost function by alternatively optimizing the policy and the constraint function. In the phase of policy optimization, they commonly solve a CRL problem (PI) by studying its Lagrangian dual:

$$D\left[\mathsf{CRL}(r, c)\right] = \min_{\lambda > 0} \max_{\pi} \mathcal{J}(\pi, r - \lambda c) + \lambda\epsilon, \tag{DI}$$

where $\mathcal{J}(\pi, r-\lambda c) = \mathbb{E}_{\mu_0, \pi, P_{\mathcal{T}}}\Big[\sum_{t=0}^{\infty} \gamma^t \Big(r(s_t, a_t) - \lambda c(s_t, a_t)\Big)\Big]$. Paternain et al. (2019) showed that CRL problems have zero duality gap:

Theorem 3.2. *(CRL has zero duality gap (Paternain et al., 2019)). Suppose that r and c are bounded and the Slater's condition holds for (PI), then strong duality holds for (PI), i.e., PI* = DI*.*

Accordingly, the optimal policy in CRL objective (PI) can be equivalently solved by utilizing an *unconstrained* objective (DI). Based on the dual representation of CRL problem, ICRL solvers are essentially solving a tri-level optimization problem (Kim et al., 2023):

$$\max_{c} \max_{\lambda} \min_{\pi} \mathcal{J}(\pi^E, r - \lambda c) - \mathcal{J}(\pi, r - \lambda c). \tag{1}$$

Given the complexity of tri-level optimization, Hugessen et al. (2024) recently explored whether we can 1) apply an IRL algorithm to recover $\tilde{r} = r - \lambda c = r - \tilde{c}$ by optimizing λ and c collectively in $\tilde{c} = \lambda c$ if the range of $\tilde{c}$ is a convex cone, and 2) learn an imitation policy by directly maximizing the cumulative rewards $\mathbb{E}[\sum_{t=0}^{\infty} \gamma^t \tilde{r}(s_t, a_t)]$ without considering the constrained optimization objective. In this work, we formally define this method as Inverse Reward Correction (IRC) as follows:

Definition 3.3. (*IRC solver for constraint inference (Li et al., 2023)*). An IRC solver is denoted as $\mathbb{S}_{\text{IRC}}(\mathcal{M}, \pi^E, r)$. A correction term Δr is a *feasible* solution from $\mathbb{S}_{\text{IRC}}$ if and only if π^E is an optimal policy for $(\mathcal{M} \backslash r) \cup r^c$, where corrected rewards $r^c(s,a) = r(s,a) + \Delta r(s,a), \forall (s,a)$. We denote by $\mathcal{R}_{\mathbb{S}_{\text{IRC}}}$ the set of feasible reward correction terms derived by $\mathbb{S}_{\text{IRC}}$.

For clarity, we simplify $(\mathcal{M} \backslash r) \cup r^c$ to $\mathcal{M} \cup r^c$ in the following discussion. Under this setting, the correction term can play the role of negative collective cost function such that $\Delta r = -\tilde{c}$. If the negative optimal $\tilde{c}$ can be represented within the bounded range of the correction term, i.e., $-\tilde{c}^* = -\lambda^* c^* \in \text{range}(\Delta r)$, where λ^* and c^* are optimal solutions of (1), the tri-level optimization can be transferred to a bi-level one as defined in the following:

$$\min_{\Delta r} \min_{\pi} \mathcal{J}(\pi^E, r + \Delta r) - \mathcal{J}(\pi, r + \Delta r). \tag{2}$$

Hugessen et al. (2024) demonstrated that this simplification results in a more performant solver. In the following sections, we will provide a more formal comparison of these solvers, focusing on their *sample complexity* and *transferability*, i.e., performance in transferring to new environments.

4 Training Efficiency: A Formal Study of Sample Complexity

In this section, we compare the training efficiency of the aforementioned ICRL and IRC solvers by deriving their sample complexity and analyzing their performance gaps.

4.1 Sample Complexity of the IRC Solver

To compute the sample complexity of IRC solver, we adopt the theoretical framework from Metelli et al. (2021) and define the feasible set of reward correction terms as follows:

Lemma 4.1. *(Feasible reward correction set implicit). Let $\mathbb{S}_{IRC}(\mathcal{M}, \pi^E, r)$ be an IRC solver. Δr is a feasible reward correction term, i.e., $\Delta r \in \mathcal{R}_{\mathbb{S}_{IRC}}$ if and only if for all $(s,a) \in \mathcal{S} \times \mathcal{A}$, it holds that:*

(i) if $\pi^E(a|s) > 0$, then $Q^{r+\Delta r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s,a) = V^{r+\Delta r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s)$,

(ii) if $\pi^E(a|s) = 0$, then $Q^{r+\Delta r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s,a) \leq V^{r+\Delta r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s)$.

Intuitively, in a CMDP, reward function r alone does not align with the expert policy π^E due to potential constraint violations, i.e. $\pi^E(a|s) = 0$ but $Q^{r,\pi^E}_{\mathcal{M}}(s,a) > V^{r,\pi^E}_{\mathcal{M}}(s)$. The correction term Δr adjusts the reward r so that $r + \Delta r$ collectively ensures the optimality of the expert policy. This approach differs from the IRL solver (Metelli et al., 2021; Lindner et al., 2022) in two key aspects: 1) IRL lacks a mechanism to leverage the known reward function r for constraint inference, and 2) IRL is not compatible with new reward signals in new environments.

To provide a fair comparison of sample complexity between the two solvers, we study a *uniform sampling exploration strategy* facilitated by a generative model. We have detailed this strategy in Appendix Algorithm 1. Let significance $\delta \in (0,1)$. This strategy guarantees that with probability greater than $1-\delta$, the Hausdorff distance d_H between the ground-truth and estimated feasible reward correction set, i.e., $\mathcal{R}_{\mathbb{S}_{\text{IRC}}}$ and $\mathcal{R}_{\widehat{\mathbb{S}}_{\text{IRC}}}$, is bounded.

In the case of the IRC solver, this Hausdorff distance is upper bounded by

$$d_H(\mathcal{R}_{\mathbb{S}_{\text{IRC}}}, \mathcal{R}_{\widehat{\mathbb{S}}_{\text{IRC}}}) \leq \max_{(s,a)\in\mathcal{S}\times\mathcal{A}} \mathcal{I}^{\Delta r}_{k+1}(s,a), \text{with } \mathcal{I}^{\Delta r}_{k+1}(s,a) = \frac{2\gamma R_{\max}}{1-\gamma}\sqrt{\frac{2\ell_{k+1}(s,a)}{N^+_{k+1}(s,a)}}, \tag{3}$$

where $N^+_{k+1}(s,a)$ is the positive cumulative count of visitations to (s,a) (formally defined in Appendix B.2) and $\ell_{k+1}(s,a) = \log\left(12SA(N^+_{k+1}(s,a))^2/\delta\right)$. Towards reducing this upper bound below a targeted accuracy, we derive the sample complexity of the IRC solver, inspired by (Metelli et al., 2021, Theorem 5.1).

Theorem 4.2. *(Sample Complexity of the IRC Solver). If an IRC solver stops at iteration K with updated accuracy ε_K, then with probability at least $1-\delta$ it converges, with the number of samples upper bounded by:*

$$n \leq \tilde{\mathcal{O}}\left(\frac{4\gamma^2 R^2_{\max} SA}{(1-\gamma)^4 \varepsilon^2_K}\right), \tag{4}$$

where $\widetilde{\mathcal{O}}$ notation suppresses logarithmic terms.

4.2 Sample Complexity of the ICRL Solver

Following a similar theoretical framework, Yue et al. (2024) derived the sample complexity for the ICRL solver by defining the following representation of the feasible cost set. We briefly recap and discuss the results below. Appendix B.6 provides a detailed review.

Lemma 4.3. *(Feasible cost set implicit (Yue et al., 2024, Lemma 4.3)). Under (Yue et al., 2024, Assumption 4.1) and let $\mathbb{S}_{ICRL}(\mathcal{M}, \pi^E, r)$ be an ICRL solver. c is a feasible cost, i.e., $c \in \mathcal{C}_{\mathbb{S}_{ICRL}}$ if and only if $\forall(s,a) \in \mathcal{S}\times\mathcal{A}$:*

(i) if $\pi^E(a|s) > 0$, $Q^{c,\pi^E}_{\mathcal{M}\cup c}(s,a) - V^{c,\pi^E}_{\mathcal{M}\cup c}(s) = 0$;

(ii) if $\pi^E(a|s) = 0$ and $A^{r,\pi^E}_{\mathcal{M}\cup c}(s,a) > 0$, $Q^{c,\pi^E}_{\mathcal{M}\cup c}(s,a) - V^{c,\pi^E}_{\mathcal{M}\cup c}(s) > 0$;

(iii) if $\pi^E(a|s) = 0$ and $A^{r,\pi^E}_{\mathcal{M}\cup c}(s,a) \leq 0$, $Q^{c,\pi^E}_{\mathcal{M}\cup c}(s,a) - V^{c,\pi^E}_{\mathcal{M}\cup c}(s) \leq 0$.

Remark 4.4. In contrast to the representation for the set of feasible reward corrections in Lemma 4.1, this feasible cost set differentiates the values of the cost value function based on whether the advantage values are above or below zero. Intuitively, when $A^{r,\pi^E}_{\mathcal{M}\cup c}(s,a) > 0$, the action a represents a movement aimed at achieving rewards that exceed those of the expert. Such actions are likely to be unsafe and thus violate the underlying constraints (Quan et al., 2024), leading to an increase in the Lagrange multiplier λ in (1) to penalize the relevant policies.

Similarly, to compute the sample complexity of ICRL solver, we utilize the strategy in Appendix Algorithm 1. Guaranteed by the strategy, the corresponding Hausdorff distance d_H between the ground-truth and estimated feasible cost set, i.e., $\mathcal{C}_{\mathbb{S}_{\text{ICRL}}}$ and $\mathcal{C}_{\widehat{\mathbb{S}}_{\text{ICRL}}}$, is upper bounded by

$$d_H(\mathcal{C}_{\mathbb{S}_{\text{ICRL}}}, \mathcal{C}_{\widehat{\mathbb{S}}_{\text{ICRL}}}) \leq \max_{(s,a)\in\mathcal{S}\times\mathcal{A}} \mathcal{I}^c_{k+1}(s,a), \text{with } \mathcal{I}^c_{k+1}(s,a) = \frac{\sigma}{(1-\gamma)^2}\sqrt{\frac{\ell_{k+1}(s,a)}{2N^+_{k+1}(s,a)}},$$

where $\sigma = \sqrt{3}\gamma C_{\max}\left(R_{\max}(3+\gamma)/\min^+\left|A^{r,\pi^E}_{\mathcal{M}\cup c}\right| + (1-\gamma)\right)$.

Theorem 4.5. *(Sample Complexity of ICRL solver (Yue et al., 2024, Theorem C.9)). If an ICRL solver terminates at iteration K with the updated accuracy ε_K, then with probability at least $1-\delta$, it converges with a number of samples upper bounded by*

$$n \leq \widetilde{\mathcal{O}}\left(\frac{\gamma^2\sigma^2 SA}{(1-\gamma)^6\varepsilon_K^2}\right). \tag{5}$$

Discussion. By comparing Theorem 4.2 with Theorem 4.5, we observe that the sample complexity of the ICRL solver exceeds that of the IRC solver by a factor of $1/(1-\gamma)^2$. This increase arises from the need to estimate the advantage function under the expert policy, i.e., $A^{r,\pi^E}_{\mathcal{M}\cup c}$, which dictates whether to impose additional costs on the current state-action pair. As we explained above, $A^{r,\pi^E}_{\mathcal{M}\cup c}$ is closely related to the update of Lagrange multiplier λ (i.e., λ should increase when $A^{r,\pi^E}_{\mathcal{M}\cup c} > 0$). The additional complexity is also reflected in the distinction between tri-level optimization (1) and bi-level optimization (2), where the Lagrange multiplier λ must converge to its optimal λ^* so that feasible cost set can be established.

5 Cross-Environment Transferability: Generalizing the Safety and Optimality of Constraints

Beyond cloning the behaviors of expert agents, a critical application of the inverse optimization methods (including IRL, ICRL, and IRC) is inferring the generalizable oracle signals (e.g., rewards or constraints) that can guide the behaviors of agents under similar environments. Denote the source CMDP as $\mathcal{M}_c = (\mathcal{S}, \mathcal{A}, P_\mathcal{T}, r, c, \epsilon, \mu_0, \gamma)$ and the target CMDP as $\mathcal{M}'_c = (\mathcal{S}, \mathcal{A}, P'_\mathcal{T}, r', c, \epsilon, \mu_0, \gamma)$, where they can be different in the reward function and the transition model (i.e., $r' \neq r$, $P'_\mathcal{T} \neq P_\mathcal{T}$). Under this setting, unlike previous studies (Cao et al., 2021; Rolland et al., 2022) where the underlying oracle signals (e.g., rewards) can be inferred from multiple environments and expert demonstrations, our study adheres to the common ICI problem setting, assuming access to only a *single* expert within a specific environment. Building upon this, we study the transferability of learned constraints across both different *transition dynamics* and varying *reward functions*.

5.1 Generalizing Safety Guarantees Across Diverse Environments

Transferring the recovered reward correction term Δr or the cost function c introduces new challenges that remain largely unexplored in the ICI literature since there is no guarantee that Δr or c with the known new reward function r' and new transition model $P'_\mathcal{T}$ will make the new expert policy $(\pi')^E$ *constraint satisfying* in the target CMDP $\mathcal{M}'_c$. That is to say, transferred Δr or c may lead to unsafe policies excluded from the feasible region in the target environment.

Challenges in Ensuring Safety with IRC Solutions. Although the IRC solver is more sample-efficient in training within the source environment, it struggles to guarantee safety in the target environment, as illustrated in Figure 1. Consider a hard constraint scenario; soft constraints will be discussed later. In the source environment, trajectory τ_1 (with a reward $r_S(\tau_1) = 2$) has a larger reward than τ_2 ($r_S(\tau_2) = 1$), but the expert agent prefers τ_2 since τ_1 passes through an unsafe state s^c. To align

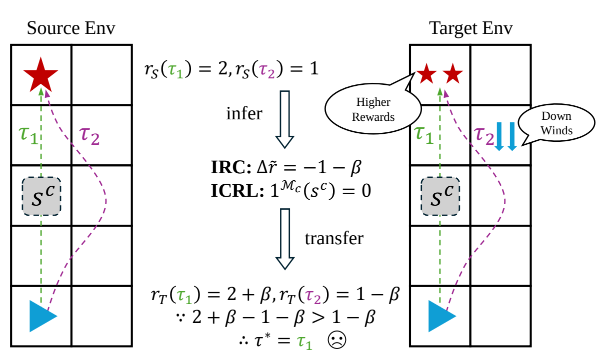

Figure 1: An example showing that transferring Δr and constraint $\mathbb{1}^{\mathcal{M}_c}(s^c) = 0$ learned in the source environment (left) to the target environment (right) induces different optimal policies (τ_1 denotes the optimal path and τ_2 denotes the second-best path).

with the expert's demonstration, the IRC solver learns a reward correction term $\Delta r(s^c) = -1 - \beta$ (where $\beta > 0$), ensuring that $r_S(\tau_1) + \Delta r(\tau_1) = 1 - \beta < 1 = r_S(\tau_2) + \Delta r(\tau_2)$ (note that $\Delta r(\tau_2) = 0$ since τ_2 does not pass through s^c). However, when this reward correction term is transferred to a similar but not identical target environment—where the reward functions and transition dynamics slightly differ ($r_T(\tau_1) = 2 + \beta$, $r_T(\tau_2) = 1 - \beta$)—the correction term becomes inapplicable. The reward correction term renders the unsafe path τ_1 as seemingly safe, with $r_T(\tau_1) + \Delta r(\tau_1) = 1 \geq 1 - \beta = r_T(\tau_2) + \Delta r(\tau_2)$, misleading the agent to choose the hazardous τ_1.

In essence, the reward correction terms reflect the extent of penalization applied to constraint-violating actions in the source environment. These penalties can be easily *offset* by increasing the gains on penalized actions or decreasing the gains on other actions in the target environment. Comparably, the ICRL solvers do not have such difficulty since they learn a hard constraint that invalidates any visit to s^c (e.g., the feasibility $1^{\mathcal{M}_c}(s^c) = 0$). Such a constraint guarantees that the agent always selects the feasible state-action pairs under both the source and target environments. This is because the cost function directly decides the boundaries of the feasible region, a modification that cannot be compensated for in the same manner as correction terms.

A Formal Study of Transferability in Safety. Transferability with single expert knowledge requires similarity restrictions between source and target environments (Metelli et al., 2021, Assumption 4.1) (Schlaginhaufen & Kamgarpour, 2024, Theorem 3.10). In the context of safety, we expect the learned constraint knowledge remains at least partially active in the target environment, i.e., the two constrained critical regions overlap so that the learned constraint information can be effectively reused. This property is characterized as follows:

Assumption 5.1. (Similarity). Let $(\mathcal{M}, \pi^E, r)$ and $(\mathcal{M}', (\pi')^E, r')$ represent the source and target input of both solvers. The intersection of their constraint violating state-action pairs is not empty: $\mathcal{G} = \{(s,a) \mid A^{r,\pi^E}_{\mathcal{M}\cup c}(s,a) > 0\} \cap \{(s,a) \mid A^{r,(\pi')^E}_{\mathcal{M}'\cup c}(s,a) > 0\} \neq \varnothing$.

Under this assumption, we begin by addressing the hard constraint scenario, in which the ICRL solver produces the cost function $\hat{c}$ that can be safely transferred to the target CMDP, ensuring safety within $\mathcal{G}$. This property is illustrated in the following lemma:

Lemma 5.2. *Suppose a hard constraint scenario. For any $(s', a') \in \mathcal{G}$, the feasible cost function $\hat{c}$ inferred by the ICRL solver can prevent the visitation to (s', a') under target CMDP.*

However, the reward correction term learned by the IRC solver fail to guarantee safety in the target environment. In the following, we identify the condition under which Δr is not transferable.

Theorem 5.3. *Suppose a hard constraint scenario. At state s, let a^E denote the expert action, a^C denote the action that satisfies $(s, a^C) \in \mathcal{G}$ and a^O denote the other actions. $\forall r' \in [0, R_{\max}]^{\mathcal{S}\times\mathcal{A}}$ and $\forall P'_{\mathcal{T}} \in \Delta^{\mathcal{S}}_{\mathcal{S}\times\mathcal{A}}$, if $\exists s \in \mathcal{S}$, $\forall\, a^E, a^O \in \mathcal{A}$, $\exists\, a^C \in \mathcal{A}$ that satisfies the following conditions, then the reward correction term Δr constructed by such Q-functions leads to unsafety in the target CMDP,*

$$Q^{r+\Delta r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}(\mathbf{e}_{(s,a^E)} - \mathbf{e}_{(s,a^C)}) \geq 0, Q^{r+\Delta r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}(\mathbf{e}_{(s,a^E)} - \mathbf{e}_{(s,a^O)}) \geq 0,$$
$$Q^{r+\Delta r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}(\mathbf{e}_{(s,a^E)} - \mathbf{e}_{(s,a^C)}) < (Y')^{-1}\left[r' - r + (Y - Y')Q^{r+\Delta r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}\right](\mathbf{e}_{(s,a^C)} - \mathbf{e}_{(s,a^E)}),$$
$$Q^{r+\Delta r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}(\mathbf{e}_{(s,a^O)} - \mathbf{e}_{(s,a^C)}) < (Y')^{-1}\left[r' - r + (Y - Y')Q^{r+\Delta r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}\right](\mathbf{e}_{(s,a^C)} - \mathbf{e}_{(s,a^O)}),$$

where $Y = (I_{\mathcal{S}\times\mathcal{A}} - \gamma P_{\mathcal{T}}\pi^E)$, $Y' = (I_{\mathcal{S}\times\mathcal{A}} - \gamma P'_{\mathcal{T}}(\pi')^E)$ and $\mathbf{e}_{(s,a)}$ denotes a vector with value of 1 at index (s,a) and 0 elsewhere.

The first two inequalities, obtained from Lemma 4.1, ensure the optimality of the source expert policy π^E. The last two inequalities state that unsafe actions will be chosen if increments in the Q function of constraint-violating actions *larger* than that of other actions after transfer. To numerically validate the above theorem, we conduct a detailed analysis of the example shown in Figure 1 in Appendix B.7.

Extension to Soft Constraint. The above results pertain to hard constraint scenarios. In cases of the soft constraint, i.e., threshold $\epsilon > 0$ in (PI), any (s,a) can be visited by feasible policies because the cost of visiting (s, a) can always be mitigated via the power of discount factor $\gamma < 1$. In this sense, cost functions, like reward correction terms, reflect the degree of penalization for constraint-violating actions that can be compensated by new transition dynamics and expert policies. Although the recovered cost function no longer guarantees safety in $\mathcal{G}$, it still outperforms inferred reward correction terms in the sense that it resists the variation between source and target reward functions. Detailed analyses and additional evaluations are presented in Appendix B.8.2.

5.2 Generalizing Optimality Guarantees Across Diverse Environments

The analysis mentioned above primarily focuses on ensuring safety (satisfying constraints) in the target environment. However, it does not guarantee that the underlying policy is optimal in terms of reward maximization. In this context, a trivial solution could involve blocking many state-action pairs deviating from expert trajectories. While this approach ensures safety, it compromises the optimality of the solution. In this section, moving a step further from safety guarantees, we establish theoretical foundations for the optimality of policies trained under the estimated *hard* constraint. As safety takes precedence over optimality, the IRC solver is not discussed in this context.

Transferability in Optimality for ICRL Solver. Guaranteeing the optimality of transferred cost functions to target environments requires sufficient knowledge of the environment landscape. To achieve this goal, we recognize the difficulty of obtaining optimality guarantees under unregularized settings (Schlaginhaufen & Kamgarpour, 2024, Remark 3.12) and emphasize that exploration is crucial for obtaining theoretical guarantees of transferability.

Assumption 5.4. (Exploration). Let $\mathcal{F} = \{s \in \mathcal{S} \mid \exists a \in \mathcal{A} \colon c(s, a) = 0\}$ be the state feasible region, and $\mathcal{Q} = \left\{\mu \in \mathbb{R}_+^{\mathcal{S}\times\mathcal{A}} \colon (E - \gamma P)^\top \mu = (1-\gamma)\mu_0\right\} \subseteq \Delta^{\mathcal{S}\times\mathcal{A}}$ be the set of occupancy measures that satisfies the Bellman flow constraints. We assume that for any $s \in \mathcal{F}$ and $\mu \in \mathcal{Q}$, the state occupancy measure $\mu(s) := \sum_a \mu(s, a)$ is lower bounded by a positive constant, i.e., $\mu(s) \geq \mu_{\min} > 0$.

To ensure the policy can exhibit exploratory behavior, we adjust the CRL objective (PI) by assuming the optimal policy maximizes the regularized objective in the following:

$$\pi^* = \arg\max_{\pi} \mathbb{E}_{\mu_0, \pi, P_{\mathcal{T}}} \left[\sum_{t=0}^{\infty} \gamma^t \Big(r(s_t, a_t) - \lambda^* \cdot c(s_t, a_t) + h(\pi(\cdot|s_t)) \Big) \right], \tag{6}$$

where the Shannon entropy regularizer $h(\pi(\cdot|s)) := -\alpha \mathbb{E}_\pi[\log \pi(a|s)]$ and the weighting term $\alpha > 0$. Additionally, we model the influence of policy by deriving the occupancy measure μ. As a result, the dual representation of the CRL problem (DI) can be recast as a convex optimization problem (Altman, 2021; Puterman, 2014):

$$\mathsf{CRL}_{P_{\mathcal{T}}}(r, c) := \arg\max_{\mu \in \mathcal{Q}} J(r, \lambda^*, c, \mu), \text{with } J(r, \lambda^*, c, \mu) = \langle r - \lambda^* c, \mu \rangle - \mathbb{E}_\mu[h(\pi^\mu)], \tag{7}$$

where $\mu(s, a) = (1-\gamma)\mathbb{E}_{\pi, P_{\mathcal{T}}}\left[\sum_{t=0}^{\infty} \gamma^t \mathbb{1}(s_t = s, a_t = a)\right]$ and $\pi^\mu(a|s) = \mu(s, a)/\mu(s)$.

Under this formulation, to quantify the performance of a given occupancy measure μ under an inferred cost function c, we define the sub-optimality gap as follows:

$$\ell_{P_{\mathcal{T}}}^{r, \lambda^*}(c, \mu) := \max_{\mu' \in \mathcal{Q}} J(r, \lambda^*, c, \mu') - J(r, \lambda^*, c, \mu). \tag{8}$$

Based on this definition, we are ready to define the transferability of optimality across the source environment $\mathcal{M}_c$ and the target environment $\mathcal{M}'_c$ with different reward and transition functions.

Definition 5.5. (ε-transferability). For some fixed $\varepsilon > 0$, we say the inferred cost function $\hat{c}$ is ε-transferable to some new reward function $r' \in [0, R_{\max}]^{\mathcal{S}\times\mathcal{A}}$ and transition law $P'_{\mathcal{T}} \in \Delta_{\mathcal{S}}^{\mathcal{S}\times\mathcal{A}}$ if $\ell_{P'_{\mathcal{T}}}^{r', (\lambda')^*}(\hat{c}, \mathsf{CRL}(c^E)) \leq \varepsilon$.

Intuitively, ε-transferability quantifies how well the inferred cost function $\hat{c}$ captures the preference of expert behaviors in a different environment. Specifically, the expert policy is ε-optimal relative to the optimal policy derived under $\hat{c}$. When the gap is large, it indicates that certain feasible state-action pairs are overly penalized, potentially leading to suboptimal policy decisions. Conversely, when the gap is small, the estimated cost function $\hat{c}$ accurately explains the expert's behaviors,

A key element in bounding this suboptimality gap in target environments is to constrain the similarity between source and target transition laws (Schlaginhaufen & Kamgarpour, 2024). To reflect corresponding feasible cost sets $\mathcal{C}_{\mathbb{S}_{\text{ICRL}}}$ while solving the unidentifiability issues in inverse problems, we map a transition law $P_{\mathcal{T}}$ to a subspace $\mathcal{U}_{P_{\mathcal{T}}} := \text{im}(E - \gamma P_{\mathcal{T}})$ via potential shaping transformation (Ng et al., 1999). The expert's optimality is ensured by this transformation via the cost affine subspace $c + \mathcal{U}_{P_{\mathcal{T}}} \subseteq \mathcal{C}_{\mathbb{S}_{\text{ICRL}}}$. A detailed discussion of this transition subspace for cost equivalence is provided in Appendix B.29. Given the high dimensionality of this subspace, we utilize principal angles to provide a more refined measure of similarity and dissimilarity between them.

Definition 5.6. (Principal angles (Galántai, 2013)) Let $\mathcal{V}, \mathcal{W} \subseteq \mathbb{R}^n$ be two subspaces of dimension $m \leq n$. The principal angles $0 \leq \theta_1(\mathcal{V}, \mathcal{W}) \leq \ldots \leq \theta_m(\mathcal{V}, \mathcal{W}) =: \theta_{\max(\mathcal{V},\mathcal{W})} \leq \pi/2$ between $\mathcal{V}$ and $\mathcal{W}$ are defined recursively via

$$\cos(\theta_i(\mathcal{V}, \mathcal{W})) = \max_{v \in \mathcal{V}, w \in \mathcal{W}} \langle v, w \rangle \text{ s.t. } \|v\|_2 = \|w\|_2 = \mathbf{1},\ \langle v, v_j \rangle = \langle w, w_j \rangle = 0,\ j = 1, \ldots, i-1,$$

where v_j, w_j are the maximizers corresponding to the angle θ_j. For two transition laws $P_{\mathcal{T}1}, P_{\mathcal{T}2}$, we define $\theta_i(P_{\mathcal{T}1}, P_{\mathcal{T}2}) := \theta_i(\mathcal{U}_{P_{\mathcal{T}1}}, \mathcal{U}_{P_{\mathcal{T}2}})$ and $\theta_{\max}(P_{\mathcal{T}1}, P_{\mathcal{T}2}) := \theta_{\max}(\mathcal{U}_{P_{\mathcal{T}1}}, \mathcal{U}_{P_{\mathcal{T}2}})$.

Theorem 5.7. *Let $P'_{\mathcal{T}}$ be the transition law in the target environment and $d_1 = \|[c^E - \hat{c}]_{\mathcal{U}_{P'_{\mathcal{T}}}}\|_2$. Suppose that Assumption 5.4 holds. If $\ell_{P_{\mathcal{T}}}^{r',(\lambda')^*}(\hat{c}, \mathsf{CRL}(c^E)) \leq \varepsilon_1$, $\hat{c}$ is ε-transferable to $P'_{\mathcal{T}}$ with*

$$\varepsilon = 2 \max \left\{ d_1^2 \sin\left(\theta_{\max}(P_{\mathcal{T}}', P_{\mathcal{T}})\right)^2 / 2, 2\varepsilon_1 / \sigma_{\mathcal{R}} \right\} / \eta, \tag{9}$$

where $\sigma_{\mathcal{R}}$ and η are regularity constants, given in Appendix B.26.

The above theorem establishes that a small margin between two transition laws, combined with strong explainability of the recovered cost under the target reward function in the source environment, ensures that the recovered cost is ε-optimal in the target environment. We observe that the transferability in optimality of cost is affected by two key factors. The first factor is the discrepancy between the source and target CMDP, which encompasses: 1) the difference in rewards and Lagrange multipliers, captured by ε_1, and 2) the discrepancy in transition dynamics, indicated by $\theta_{\max}$. The second factor is the estimation error associated with the recovered cost $\hat{c}$, denoted by d_1.

6 Empirical Evaluation

We empirically evaluate the ICRL solver against the IRC solver in four different constrained gridworld environments. For each gridworld environment, source and target environments differ in reward functions and transition dynamics. Also, we present a series of visualizations to offer deeper insights into the discrepancy between the two solvers in constraint inference.

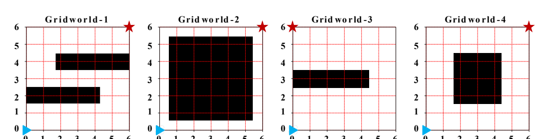

Figure 2: Four different Gridworld environments.

Experiment Setting. The experimental settings are primarily based on a public ICRL benchmark (Liu et al., 2023). In our experiments, we focus on evaluating the *training efficiency* and *transferability* of the ICRL and IRC solvers. The results are assessed using two key metrics:1) *discounted cumulative rewards*, which quantify the total rewards achieved by the learned policy. 2) *discounted cumulative costs*, which calculate the total costs incurred by the policy. We compare the uniform sampling strategy (Appendix Algorithm 1) of the ICRL and IRC solvers.

In Figure 2, we design four distinct Gridworld environments for both source and target environments, each characterized by unique constraints. The agent's goal is to navigate from a starting location (blue) to a target location (red) while avoiding constraints (black). The expert policy is trained under ground-truth constraints. Four source environments exhibit a stochasticity of $p = 0.05$, i.e., the agent takes a uniformly randomized action with that probability, while four target environments feature a higher stochasticity of $p = 0.1$. All rewards are assigned at the target location, with identical values of 1 in four source environments and $2, 7, 7, 15$ in four respective target environments.

Figure 3 demonstrates the learned cost functions by the ICRL solver at each state and the learned reward correction terms by the IRC solver at each state, in the source environments in four Gridworld settings (from left to right, Gridworld-1,2,3,4). Figure 4 demonstrates the accumulated rewards and costs of resulting policies learned under inferred reward correction terms by the IRC solver and cost functions by the ICRL solver at each iteration. In the source environments, we observe that the IRC solver converges more quickly than the ICRL solver, indicating higher training efficiency. However, in the target environments, we find that inferred reward correction terms lead to unsafe policies (costs exceeding 0), whereas recovered cost functions ensure both safety (costs converging to 0) and optimality (rewards converging to the expert). For continuous environments, we leverage the Maximum Entropy framework of ICRL (Malik et al., 2021) and IRC solvers (Hugessen et al., 2024) where the two solvers are designed to recover constraint knowledge that best explains the expert demonstrations from an offline dataset. Check Appendix C and D for more experimental results and implementation details.

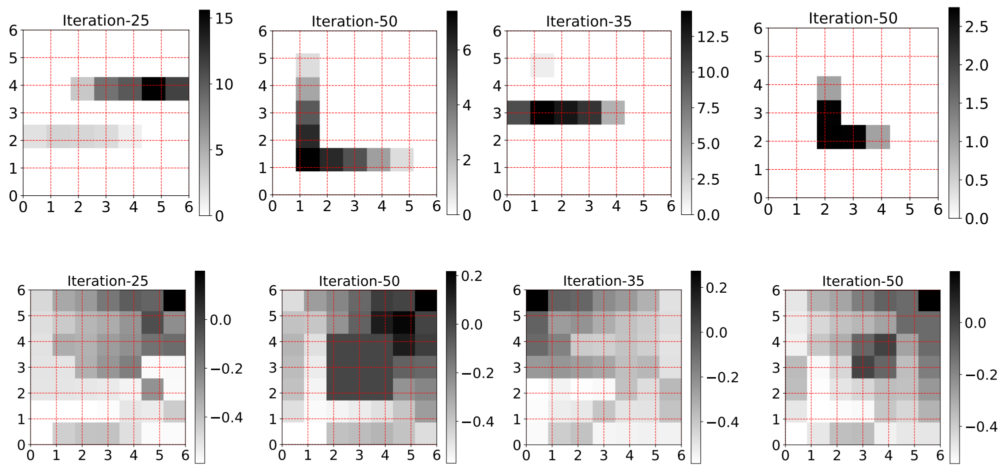

Figure 3: Learned constraint information by the ICRL solver (top) and the IRC solver (bottom).

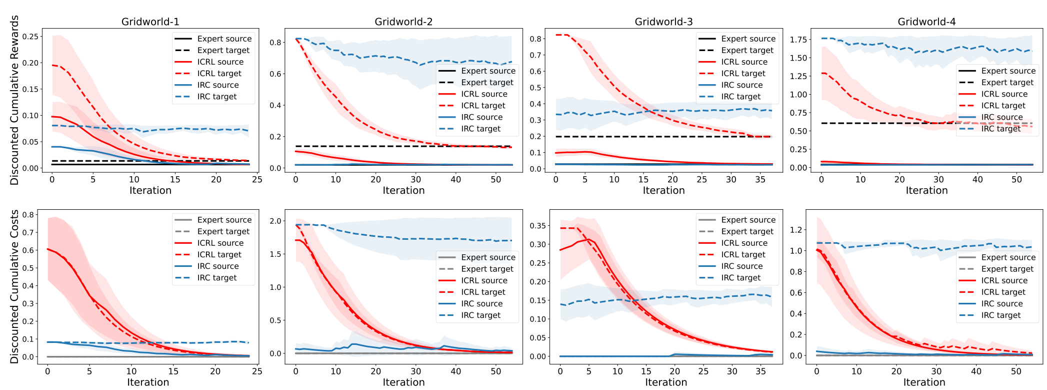

Figure 4: Training curves of discounted cumulative rewards (top), costs (bottom) for the ICRL (red) and IRC (blue) solvers across four Gridworld environments. The expert's rewards and costs are represented in grey. Solid and dashed lines correspond to source and target environments, respectively.

7 Conclusion

Summary. In this paper, we study the novel challenge of transferring learned constraint information from source to target environments. Constraint information can be either implicit reward correction terms by IRC solvers or explicit cost functions by ICRL solvers. We compare both solvers regarding training efficiency and transferability. First, we evaluate their training efficiencies. While the IRC solver is guaranteed faster convergence, the additional sample complexity required by the ICRL solver proves essential for ensuring both safety and optimality in new environments. Under hard constraints, the recovered cost functions strictly prevent the agent in target environments from entering overlapping critical regions, whereas the inferred reward correction terms can be easily offset by variations in transition laws and reward functions, leading to insufficient constraint representation. We also derive conditions that limit the similarity between source and target environments to ensure the optimality for the ICRL solver. Empirical studies across various environments validate our findings.

Limitations and Future Work. Our research initiates intriguing avenues for future studies. First, extending our approach to incorporate demonstrations from multiple experts, including sub-optimal ones would be a valuable contribution to the field. Second, deriving sufficient and necessary conditions to guarantee the transferability of constraint information under arbitrary reward functions and transition dynamics presents an intriguing avenue for exploration. Lastly, we believe that extending to more complex and scalable tasks, particularly those in real-world applications, could offer valuable insights into the practical challenges and opportunities involved in transferring constraint information.

REFERENCES

Eitan Altman. *Constrained Markov decision processes*. Routledge, 2021.

Kareem Amin, Nan Jiang, and Satinder Singh. Repeated inverse reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

Mattijs Baert, Pietro Mazzaglia, Sam Leroux, and Pieter Simoens. Maximum causal entropy inverse constrained reinforcement learning. *arXiv preprint arXiv:2305.02857*, 2023.

Haoyang Cao, Samuel Cohen, and Lukasz Szpruch. Identifiability in inverse reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pp. 12362–12373, 2021.

Zeyu Feng, Bowen Zhang, Jianxin Bi, and Harold Soh. Safety-constrained policy transfer with successor features. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7219–7225, 2023.

Aurél Galántai. *Projectors and projection methods*, volume 6. Springer Science & Business Media, 2013.

Ashish Gaurav, Kasra Rezaee, Guiliang Liu, and Pascal Poupart. Learning soft constraints from constrained expert demonstrations. In *International Conference on Learning Representations (ICLR)*, 2023.

Shangding Gu, Long Yang, Yali Du, Guang Chen, Florian Walter, Jun Wang, Yaodong Yang, and Alois C. Knoll. A review of safe reinforcement learning: Methods, theory and applications. *arxiv preprint arxiv: 2205.10330*, 2022.

Adriana Hugessen, Harley Wiltzer, and Glen Berseth. Simplifying constraint inference with inverse reinforcement learning. In *the First Reinforcement Learning Safety Workshop at Reinforcement Learning Conference (RLC)*, 2024.

Konwoo Kim, Gokul Swamy, Zuxin Liu, Ding Zhao, Sanjiban Choudhury, and Zhiwei Steven Wu. Learning shared safety constraints from multi-task demonstrations. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Jianxiong Li, Xiao Hu, Haoran Xu, Jingjing Liu, Xianyuan Zhan, Qing-Shan Jia, and Ya-Qin Zhang. Mind the gap: Offline policy optimization for imperfect rewards. In *International Conference on Learning Representations (ICLR)*, 2023.

David Lindner, Andreas Krause, and Giorgia Ramponi. Active exploration for inverse reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 5843–5853, 2022.

Guiliang Liu, Yudong Luo, Ashish Gaurav, Kasra Rezaee, and Pascal Poupart. Benchmarking constraint inference in inverse reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2023.

Shicheng Liu and Minghui Zhu. Distributed inverse constrained reinforcement learning for multi-agent systems. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Yongshuai Liu, Avishai Halev, and Xin Liu. Policy learning with constraints in model-free reinforcement learning: A survey. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 4508–4515, 2021.

Shehryar Malik, Usman Anwar, Alireza Aghasi, and Ali Ahmed. Inverse constrained reinforcement learning. In *International Conference on Machine Learning (ICML)*, pp. 7390–7399, 2021.

Alberto Maria Metelli, Giorgia Ramponi, Alessandro Concetti, and Marcello Restelli. Provably efficient learning of transferable rewards. In *International Conference on Machine Learning (ICML)*, pp. 7665–7676, 2021.

A. Ng, Daishi Harada, and Stuart J. Russell. Policy invariance under reward transformations: Theory and application to reward shaping. In *International Conference on Machine Learning (ICML)*, 1999.

Andrew Y Ng, Stuart Russell, et al. Algorithms for inverse reinforcement learning. In *International Conference on Machine Learning (ICML)*, pp. 663–670, 2000.

Santiago Paternain, Luiz F. O. Chamon, Miguel Calvo-Fullana, and Alejandro Ribeiro. Constrained reinforcement learning has zero duality gap. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 7553–7563, 2019.

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

Guanren Qiao, Guiliang Liu, Pascal Poupart, and ZhiQiang Xu. Multi-modal inverse constrained reinforcement learning from a mixture of demonstrations. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Guorui Quan, Zhiqiang Xu, and Guiliang Liu. Learning constraints from offline demonstrations via superior distribution correction estimation. In *International Conference on Machine Learning (ICML)*, 2024.

R Tyrrell Rockafellar and Roger J-B Wets. *Variational analysis*, volume 317. Springer Science & Business Media, 2009.

Paul Rolland, Luca Viano, Norman Schürhoff, Boris Nikolov, and Volkan Cevher. Identifiability and generalizability from multiple experts in inverse reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 550–564, 2022.

Andreas Schlaginhaufen and Maryam Kamgarpour. Towards the transferability of rewards recovered via regularized inverse reinforcement learning. *arxiv preprint arxiv: 2406.01793*, 2024.

Dexter R. R. Scobee and S. Shankar Sastry. Maximum likelihood constraint inference for inverse reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2020.

Joar Max Viktor Skalse, Matthew Farrugia-Roberts, Stuart Russell, Alessandro Abate, and Adam Gleave. Invariance in policy optimisation and partial identifiability in reward learning. In *International Conference on Machine Learning (ICML)*, pp. 32033–32058, 2023.

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

Sheng Xu and Guiliang Liu. Robust inverse constrained reinforcement learning under model misspecification. In *International Conference on Machine Learning (ICML)*, 2024a.

Sheng Xu and Guiliang Liu. Uncertainty-aware constraint inference in inverse constrained reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2024b.

Bo Yue, Jian Li, and Guiliang Liu. Provably efficient exploration in inverse constrained reinforcement learning. *arXiv preprint arXiv:2409.15963*, 2024.

Quanqi Zhang, Chengwei Wu, Haoyu Tian, Yabin Gao, Weiran Yao, and Ligang Wu. Safety reinforcement learning control via transfer learning. *Automatica*, 166:111714, 2024. ISSN 0005-1098.

Table of Contents

A General Notations and Nomenclature

In Table 1, we report the explicit definition of notations and nomenclature applied in our paper.

B Definitions, Theorems, and Proofs

B.1 Additional Definitions

Definition B.1. (Hausdorff distance). (Rockafellar & Wets, 2009). Let (M, d) be a metric space. The Hausdorff distance d_H between two non-empty subsets $A \subseteq M$ and $B \subseteq M$ with distance function d is defined as:

$$d_H(A, B) = \max\left\{\sup_{a\in A}\inf_{b\in B} d(a, b), \sup_{b\in B}\inf_{a\in A} d(b, a)\right\}$$

where:

- $\inf_{b\in B} d(a, b)$ denotes the shortest distance from a point $a \in A$ to all points in set B.
- $\sup_{a\in A}\inf_{b\in B} d(a, b)$ is the maximum of these shortest distances for all points in A.
- Similarly, $\sup_{b\in B}\inf_{a\in A} d(b, a)$ represents the maximum shortest distance from all points in B to set A.

Definition B.2. (Affine space). An affine subspace A of a vector space V is defined as:

$$A = v_0 + W = \{v_0 + w \mid w \in W\},$$

where v_0 is a fixed vector in V and W is a linear subspace of V.

Definition B.3. (Operator). Let $f \in \mathbb{R}^{\mathcal{S}}$ and $g \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$. We denote by P and π the operators induced by the transition model p and by the policy π, i.e.,$(Pf)(s, a) = \sum_{s'\in\mathcal{S}} p(s'|s, a)f(s')$ and $(\pi g)(s) = \sum_{a\in\mathcal{A}} \pi(a|s)g(s, a)$. Moreover, the expansion operator $(Ef)(s, a) = f(s)$. Given $\pi \in$

Table 1: General Notations and Nomenclature

Symbol	Name	Signature
$\mathcal{S}$	State space	/
$\mathcal{A}$	Action space	/
r/r'	Source/Target reward function	$\mathbb{R}^{\mathcal{S}\times\mathcal{A}}$
$P_{\mathcal{T}}/P'_{\mathcal{T}}$	Source/Target transition model	$\Delta^{\mathcal{S}}_{\mathcal{S}\times\mathcal{A}}$
$\pi^E/(\pi')^E$	Source/Target expert policy	$\Delta^{\mathcal{A}}_{\mathcal{S}}$
c^E	Underlying expert cost function	$\mathbb{R}^{\mathcal{S}\times\mathcal{A}}$
c	Cost function	$\mathbb{R}^{\mathcal{S}\times\mathcal{A}}$
Δr	Reward correction term	$\mathbb{R}^{\mathcal{S}\times\mathcal{A}}$
$V^{r_1,\pi}_{\mathcal{M}\cup r}$	Reward state-value function for r_1 of π in $\mathcal{M}\cup r$	$\mathbb{R}^{\mathcal{S}}$
$Q^{r_1,\pi}_{\mathcal{M}\cup r}$	Reward action-value function for r_1 of π in $\mathcal{M}\cup r$	$\mathbb{R}^{\mathcal{S}\times\mathcal{A}}$
$A^{r_1,\pi}_{\mathcal{M}\cup r}$	Reward advantage function for r_1 of π in $\mathcal{M}\cup r$	$\mathbb{R}^{\mathcal{S}\times\mathcal{A}}$
$V^{c,\pi}_{\mathcal{M}\cup c}$	Cost state-value function for c of π in $\mathcal{M}\cup c$	$\mathbb{R}^{\mathcal{S}}$
$Q^{c,\pi}_{\mathcal{M}\cup c}$	Cost action-value function for c of π in $\mathcal{M}\cup c$	$\mathbb{R}^{\mathcal{S}\times\mathcal{A}}$
$V^{r,\pi}_{\mathcal{M}\cup c}$	Cost state-value function for r of π in $\mathcal{M}\cup c$	$\mathbb{R}^{\mathcal{S}}$
$Q^{r,\pi}_{\mathcal{M}\cup c}$	Cost action-value function for r of π in $\mathcal{M}\cup c$	$\mathbb{R}^{\mathcal{S}\times\mathcal{A}}$
$\mathbb{S}_{\text{IRC}}$	IRC solver	/
$\mathcal{R}_{\mathbb{S}_{\text{IRC}}}$	Set of feasible reward correction terms	$\{\mathbb{R}^{\mathcal{S}\times\mathcal{A}}\}$
$\mathbb{S}_{\text{ICRL}}$	ICRL solver	/
$\mathcal{C}_{\mathbb{S}_{\text{ICRL}}}$	Set of feasible cost functions	$\{\mathbb{R}^{\mathcal{S}\times\mathcal{A}}\}$
ε	Target accuracy	$\mathbb{R}^+$
δ	Significancy	$(0,1)$
θ	Principal angle	$[0,\pi/2]$
$\|\cdot\|_\infty$	Infinity norm	/
E	Expansion operator	$\mathbb{R}^{\mathcal{S}}\to\mathbb{R}^{\mathcal{S}\times\mathcal{A}}$

$\Delta^{\mathcal{A}}_{\mathcal{S}}$, we denote with $(B^\pi g)(s,a)=g(s,a)1\{\pi(a|s)>0\}$ and $(B^\pi g)(s,a)=g(s,a)1\{\pi(a|s)>0\}$ and $(\overline{B}^\pi g)(s,a)=g(s,a)1\{\pi(a|s)=0\}$.

Definition B.4. (Inifinity norm). For a vector a, we define the vector infinity norm as $||a||_\infty = \max_i|a_i|$. For a matrix A, we define the matrix infinity norm as $||A||_\infty=\max_i\sum_j|A_{ij}|$.

B.2 Estimating the Transition Model and Expert Policy

We define how we estimate the transition model and the expert policy in Algorithm 1.

We record the returns of a state-action pair (s,a) by observing a next state $s'\sim P(\cdot|s,a)$, and the preference of expert agents $a_E\sim\pi^E(\cdot|s)$ in each visited state. For iteration k, we denote by $n_k(s,a,s')$ the number of times we observe the transition tuple (s,a,s'). Denote $n_k(s,a)=\sum_{s'\in\mathcal{S}}n_k(s,a,s')$ and $n_k(s)=\sum_{a\in\mathcal{A}}n_k(s,a)$. For the expert policy and the transition model estimation, we define the cumulative counts $N_k(s,a,s')=\sum_{j=1}^k n_j(s,a,s')$, $N_k(s,a)=\sum_{j=1}^k n_j(s,a)$ and $N_k(s)=\sum_{j=1}^k n_j(s)$. Finally, we represent the estimated transition model and expert policy as follows:

$$\widehat{P_{\mathcal{T}}}_k(s'|s,a)=\frac{N_k(s,a,s')}{N_k^+(s,a)},\quad \widehat{\pi}_k^E(a|s)=\frac{N_k(s,a)}{N_k^+(s)},\tag{10}$$

where $x^+=\max\{1,x\}$.

B.3 Uniform Sampling Strategy for Each Solver

To acquire desired information, the agent utilizes the uniform sampling strategy to query a generative model. Specifically, the agent can always query a generative model about a state-action pair (s,a) to receive a next state $s'\sim P(\cdot|s,a)$ and about a state s to receive an expert action $a_E\sim\pi^E(\cdot|s)$.

Algorithm 1 Uniform Sampling Strategy With a Generative Model

Input: significance $\delta \in (0,1)$, target accuracy ε, maximum number of samples per iteration $n_{\max}$
Initialize $k \leftarrow 0$, $\varepsilon_0 = \frac{1}{1-\gamma}$
while $\varepsilon_k > \varepsilon$ **do**
  Collect $\lceil \frac{n_{\max}}{SA} \rceil$ samples from each $(s,a) \in \mathcal{S} \times \mathcal{A}$
  For IRC solver, update accuracy $\varepsilon_{k+1} = \frac{1}{1-\gamma} \max_{(s,a)\in\mathcal{S}\times\mathcal{A}} \mathcal{I}_{k+1}^{\Delta r}(s,a)$
  For ICRL solver, update accuracy $\varepsilon_{k+1} = \frac{1}{1-\gamma} \max_{(s,a)\in\mathcal{S}\times\mathcal{A}} \mathcal{I}_{k+1}^{c}(s,a)$
  Update $\widehat{\pi}_{k+1}^E$ and $\widehat{P_{\mathcal{T}}}_{k+1}$ according to Appendix B.2.
  $k \leftarrow k+1$
end while

B.4 Theoretical Results of the IRL Solver

In this part, we discuss the IRL solver for constraint inference, which forms the theoretical foundation of the IRC solver.

Inverse Reinforcement Learning (IRL). Typically, IRL solvers recover the reward function from expert demonstrations, where the environment is always considered to be safe. To employ IRL for inferring constraints in environments with safety issues, we need to modify the original formalization of the IRL problem (Ng et al., 2000) as follows.

Definition B.5. (*IRL solver for constraint inference*). An IRL solver can be depicted as a function of a pair $\mathbb{S}_{\text{IRL}}(\mathcal{M}, \pi^E)$, where $\mathcal{M}$ is a $\mathcal{M}_c \backslash c$ (CMDP without knowing the cost) and $\pi^E \in \Delta_{\mathcal{S}}^{\mathcal{A}}$ is an expert's policy. A corrected reward function r^c is a *feasible* solution from $\mathbb{S}_{\text{IRL}}$, if and only if π^E is an optimal policy for $(\mathcal{M}\backslash r) \cup r^c$. We denote by $\mathcal{R}_{\mathbb{S}_{\text{IRL}}}$ the set of feasible reward functions.

Lemma B.6. *(Feasible reward set implicit) (Metelli et al., 2021, Lemma 3.1). Let $\mathbb{S}_{IRL}(\mathcal{M}, \pi^E)$ be an IRL solver. A reward function $r^c \in \mathcal{R}_{\mathbb{S}_{IRL}}$ if and only if for all $(s,a) \in \mathcal{S} \times \mathcal{A}$, the following holds:*

- *if $\pi^E(a|s) > 0$, $Q_{\mathcal{M}\cup r^c}^{r^c,\pi^E}(s,a) = V_{\mathcal{M}\cup r^c}^{r^c,\pi^E}(s)$,*
- *if $\pi^E(a|s) = 0$, $Q_{\mathcal{M}\cup r^c}^{r^c,\pi^E}(s,a) \leq V_{\mathcal{M}\cup r^c}^{r^c,\pi^E}(s)$.*

Lemma B.7. *(Metelli et al., 2021, Lemma B.1). Let $\mathbb{S}_{IRL}(\mathcal{M}, \pi^E)$ be an IRL solver. A Q-function satisfies the condition of Lemma B.6 if and only if there exist $\zeta \in \mathbb{R}_{\geq 0}^{\mathcal{S}\times\mathcal{A}}$ and $V^r \in \mathbb{R}^{\mathcal{S}}$ such that:*

$$Q_{\mathcal{M}\cup r^c}^{r^c,\pi^E} = -\overline{B}^{\pi^E}\zeta + EV^r. \tag{11}$$

Furthermore, $\|V^r\|_\infty \leq \|Q_{\mathcal{M}\cup r^c}^r\|_\infty$ and the the expansion operator E satisfies $(Ef)(s,a) = f(s)$.

Lemma B.8. *(Feasible reward set explicit) (Metelli et al., 2021, Lemma 3.2). Let $\mathbb{S}_{IRL}(\mathcal{M}, \pi^E)$ be an IRL solver. A reward function $r^c \in \mathcal{R}_{\mathbb{S}_{IRL}}$ if and only if there exist $\zeta \in \mathbb{R}_{\geq 0}^{\mathcal{S}\times\mathcal{A}}$ and $V^r \in \mathcal{R}^{\mathcal{S}}$ such that*

$$r^c = -\overline{B}^{\pi^E}\zeta + (E - \gamma P_{\mathcal{T}})V^r. \tag{12}$$

Lemma B.9. *(Error Propagation) (Metelli et al., 2021, Theorem 3.1). Let $\mathbb{S}_{IRL}(\mathcal{M}, \pi^E)$ and $\widehat{\mathbb{S}}_{IRL}(\widehat{\mathcal{M}}, \widehat{\pi}^E)$ be two instances of IRL solver. Then, for any $r^c \in \mathcal{R}_{\mathbb{S}_{IRL}}$ such that $r^c = -\overline{B}^{\pi^E}\zeta + (E - \gamma P_{\mathcal{T}})V^r$ and $\|r^c\|_\infty \leq R_{\max}$, there exists $\widehat{r}^c \in \mathcal{R}_{\widehat{\mathbb{S}}_{IRL}}$ such that element-wise it holds that:*

$$|r^c - \widehat{r}^c| \leq \overline{B}^{\pi^E} B^{\widehat{\pi}^E}\zeta + \gamma \left| \left(P_{\mathcal{T}} - \widehat{P_{\mathcal{T}}} \right) V^r \right|. \tag{13}$$

Furthermore, $\|\zeta\|_\infty \leq \frac{R_{\max}}{1-\gamma}$ and $\|V^r\|_\infty \leq \frac{R_{\max}}{1-\gamma}$.

Theorem B.10. *(Sample Complexity of the IRL Solver) (Metelli et al., 2021, Theorem 5.1). If an IRL solver stops at iteration K with accuracy ε_K, then with probability at least $1-\delta$ it converges, with a number of samples upper bounded by:*

$$n \leq \tilde{\mathcal{O}}\left(\frac{\gamma^2 R_{\max}^2 SA}{(1-\gamma)^4 \varepsilon_K^2} \right). \tag{14}$$

Proof. The above result can be obtained by setting $\mathcal{M}' = \mathcal{M}$ and $\gamma' = \gamma$ in (Metelli et al., 2021, Theorem 5.1). $\square$

Remark B.11. In IRL solvers, although r in the CMDP is independent of the cost function c, r^c indeed captures the information of underlying constraint signals for enabling the imitating agent to reproduce π^E. However, the IRL solver lacks a mechanism to leverage the known reward function r for constraint inference. Moreover, since r^c relies on the current reward function, directly transferring r^c to CMDPs with different reward functions is highly risky. Therefore, it is necessary to adjust the IRL solver's settings to accommodate changes in the reward function. Li et al. (2023) propose modifying imperfect reward functions to align them with expert behaviors, but this approach lacks tractable sample complexity. To address these limitations, we propose a modified version of the IRL solver, the Inverse Reward Correction (IRC) solver in the main paper.

B.5 Theoretical Results of the IRC Solver

In this part, we provide additional discussion regarding the IRC solver for constraint inference.

Inverse Reward Correction (IRC). Based on the known r, IRC solvers learn a reward correction term $\Delta r(s,a)$ to capture constraint signals. The goal of IRC solvers is to enable the imitating agent to match expert demonstrations by following the corrected rewards: $r^c(s,a) = r(s,a) + \Delta r(s,a), \forall (s,a) \in \mathcal{S} \times \mathcal{A}$.

Lemma 4.1. *(Feasible reward correction set implicit). Let $\mathbb{S}_{IRC}(\mathcal{M}, \pi^E, r)$ be an IRC solver. Δr is a feasible reward correction term, i.e., $\Delta r \in \mathcal{R}_{\mathbb{S}_{IRC}}$ if and only if for all $(s,a) \in \mathcal{S} \times \mathcal{A}$, it holds that:*

(i) if $\pi^E(a|s) > 0$, $Q^{r+\Delta r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s,a) = V^{r+\Delta r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s)$,

(ii) if $\pi^E(a|s) = 0$, $Q^{r+\Delta r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s,a) \leq V^{r+\Delta r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s)$.

Proof. From Lemma B.6 and the decomposition that $r^c(s,a) = r(s,a) + \Delta r(s,a), \forall (s,a)$, we have,

(i) if $\pi^E(a|s) > 0$, $Q^{r+\Delta r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s,a) = V^{r+\Delta r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s)$,

(ii) if $\pi^E(a|s) = 0$, $Q^{r+\Delta r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s,a) \leq V^{r+\Delta r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s)$.

$\square$

Corollary B.12. *(Feasible reward correction set implicit). Let $\mathbb{S}_{IRC}(\mathcal{M}, \pi^E, r)$ be an IRC solver. Let $\Delta r \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$, then Δr is a feasible reward correction term, i.e., $\Delta r \in \mathcal{R}_{\mathbb{S}_{IRC}}$ if and only if for all $(s,a) \in \mathcal{S} \times \mathcal{A}$, it holds that:*

(i) if $\pi^E(a|s) > 0$, $Q^{\Delta r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s,a) = -A^{r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s,a) + V^{\Delta r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s)$,

(ii) if $\pi^E(a|s) = 0$, $Q^{\Delta r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s,a) \leq -A^{r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s,a) + V^{\Delta r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s)$.

Proof. Note that, for any given policy π, Q-function and V-function are both linear towards reward function, i.e., $Q^{r_1+r_2,\pi}_{\mathcal{M}\cup(r_1+r_2)} = Q^{r_1,\pi}_{\mathcal{M}\cup(r_1+r_2)} + Q^{r_2,\pi}_{\mathcal{M}\cup(r_1+r_2)}$ and $V^{r_1+r_2,\pi}_{\mathcal{M}\cup(r_1+r_2)} = V^{r_1,\pi}_{\mathcal{M}\cup(r_1+r_2)} + V^{r_2,\pi}_{\mathcal{M}\cup(r_1+r_2)}$. Thus, inheriting from Lemma 4.1, we obtain,

(i) if $\pi^E(a|s) > 0$, $Q^{\Delta r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s,a) + Q^{r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s,a) = V^{r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s) + V^{\Delta r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s)$,

(ii) if $\pi^E(a|s) = 0$, $Q^{\Delta r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s,a) + Q^{r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s,a) \leq V^{r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s) + V^{\Delta r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s)$.

By simple transposition, we derive,

(i) if $\pi^E(a|s) > 0$, $Q^{\Delta r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s,a) = -Q^{r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s,a) + V^{r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s) + V^{\Delta r, \pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s)$,

(ii) if $\pi^E(a|s) = 0$, $Q^{\Delta r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s,a) \leq -Q^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s,a) + V^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s) + V^{\Delta r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s)$.

By $A^{r,\pi}_{\mathcal{M}\cup r_1}(s,a) = Q^{r,\pi}_{\mathcal{M}\cup r_1}(s,a) - V^{r,\pi}_{\mathcal{M}\cup r_1}(s)$, we have,

(i) if $\pi^E(a|s) > 0$, $Q^{\Delta r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s,a) = -A^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s,a) + V^{\Delta r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s)$,

(ii) if $\pi^E(a|s) = 0$, $Q^{\Delta r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s,a) \leq -A^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s,a) + V^{\Delta r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}(s)$.

□

Lemma B.13. *Let $\mathbb{S}_{IRC}(\mathcal{M},\pi^E,r)$ be an IRC solver. A Q-function satisfies the condition of Lemma 4.1 if and only if there exist $\zeta \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}_{\geqslant 0}$ and $V^r \in \mathbb{R}^{\mathcal{S}}$ such that:*

$$Q^{\Delta r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)} = -\overline{B}^{\pi^E}\zeta - A^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)} + EV^r. \tag{15}$$

Proof. The proof can be easily derived from from (Metelli et al., 2021, Lemma B.1.). □

Lemma B.14. *(Feasible reward correction set explicit). Let $\mathbb{S}_{IRC}(\mathcal{M},\pi^E,r)$ be an IRC solver. Δr is a feasible reward correction term, i.e., $\Delta r \in \mathcal{R}_{\mathbb{S}_{IRC}}$ if and only if there exist $\zeta \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}_{\geq 0}$ and $V^r \in \mathbb{R}^{\mathcal{S}}$ such that:*

$$\Delta r = -\overline{B}^{\pi^E}\zeta - r + (E-\gamma P_{\mathcal{T}})\,V^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)} + (E-\gamma P_{\mathcal{T}})\,V^r. \tag{16}$$

Proof. From Bellman equation (Sutton & Barto, 2018), $\Delta r = \left(I_{\mathcal{S}\times\mathcal{A}} - \gamma P_{\mathcal{T}}\pi^E\right) Q^{\Delta r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}$ and $r = \left(I_{\mathcal{S}\times\mathcal{A}} - \gamma P_{\mathcal{T}}\pi^E\right) Q^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}$.

$$\begin{aligned}
\Delta r &= \left(I_{\mathcal{S}\times\mathcal{A}} - \gamma P_{\mathcal{T}}\pi^E\right) Q^{\Delta r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)} \\
&= \left(I_{\mathcal{S}\times\mathcal{A}} - \gamma P_{\mathcal{T}}\pi^E\right)\left(-\overline{B}^{\pi^E}\zeta - A^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)} + EV^r\right) \\
&= -\overline{B}^{\pi^E}\zeta - A^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)} + EV^r + \gamma P_{\mathcal{T}}\pi^E A^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)} - \gamma P_{\mathcal{T}} V^r \\
&= -\overline{B}^{\pi^E}\zeta - \left(I_{\mathcal{S}\times\mathcal{A}} - \gamma P_{\mathcal{T}}\pi^E\right) A^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)} + (E-\gamma P_{\mathcal{T}})\,V^r \\
&= -\overline{B}^{\pi^E}\zeta - \left(I_{\mathcal{S}\times\mathcal{A}} - \gamma P_{\mathcal{T}}\pi^E\right)\left(Q^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)} - EV^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}\right) + (E-\gamma P_{\mathcal{T}})\,V^r \\
&= -\overline{B}^{\pi^E}\zeta - \left(I_{\mathcal{S}\times\mathcal{A}} - \gamma P_{\mathcal{T}}\pi^E\right) Q^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)} + \left(I_{\mathcal{S}\times\mathcal{A}} - \gamma P_{\mathcal{T}}\pi^E\right) EV^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)} + (E-\gamma P_{\mathcal{T}})\,V^r \\
&= -\overline{B}^{\pi^E}\zeta - r + (E-\gamma P_{\mathcal{T}})\,V^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)} + (E-\gamma P_{\mathcal{T}})\,V^r,
\end{aligned} \tag{17}$$

where the last equality utilizes $\pi^E E = I_{\mathcal{S}}$. □

Lemma B.15. *(Error Propagation). Let $\mathbb{S}_{IRC}(\mathcal{M},\pi^E,r)$ and $\widehat{\mathbb{S}}_{IRC}(\widehat{\mathcal{M}},\widehat{\pi}^E,\widehat{r})$ be two instances of IRC solver. Then, for any $\Delta r \in \mathcal{R}_{\mathbb{S}_{IRC}}$ such that $\Delta r = -\overline{B}^{\pi^E}\zeta - r + (E-\gamma P_{\mathcal{T}})\,V^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)} + (E-\gamma P_{\mathcal{T}})\,V^r$, there exists $\widehat{\Delta r} \in \mathcal{R}_{\widehat{\mathbb{S}}_{IRC}}$ such that element-wise it holds that:*

$$\left|\Delta r - \widehat{\Delta r}\right| \leq \overline{B}^{\pi^E} B^{\widehat{\pi}^E}\zeta + \gamma\left|\left(P_{\mathcal{T}} - \widehat{P_{\mathcal{T}}}\right)\left(V^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)} + V^r\right)\right|. \tag{18}$$

Furthermore, $\|\zeta\|_\infty \leq \frac{2R_{\max}}{1-\gamma}$, $\|V^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}\|_\infty \leq \frac{R_{\max}}{1-\gamma}$ and $\|V^r\|_\infty \leq \frac{R_{\max}}{1-\gamma}$.

Proof. From Lemma B.14, we can express Δr and $\widehat{\Delta r}$ as,

$$\Delta r = -\overline{B}^{\pi^E}\zeta - r + (E-\gamma P_{\mathcal{T}})\,V^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)} + (E-\gamma P_{\mathcal{T}})\,V^r \tag{19}$$

$$\widehat{\Delta r} = -\overline{B}^{\widehat{\pi}^E}\widehat{\zeta} - r + (E-\gamma\widehat{P_{\mathcal{T}}})V^{r,\widehat{\pi}^E}_{\widehat{\mathcal{M}}\cup(r+\Delta r)} + (E-\gamma\widehat{P_{\mathcal{T}}})\widehat{V}^r \tag{20}$$

Since we look for the existence of $\widehat{\Delta r}$, we provide a specific choice of $\widehat{\zeta}$ and $\widehat{V}^r$: $\widehat{\zeta} = \overline{B}^{\pi^E}\zeta$ and $\widehat{V}^r = V^r + V^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)} - V^{r,\widehat{\pi}^E}_{\widehat{\mathcal{M}}\cup(r+\Delta r)}$

$$\Delta r - \widehat{\Delta r} = -\overline{B}^{\pi^E} B^{\widehat{\pi}^E}\zeta - \gamma\left(P_{\mathcal{T}} - \widehat{P_{\mathcal{T}}}\right)\left(V^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)} + V^r\right) \tag{21}$$

By taking the absolute value and applying the triangular inequality, we obtain:

$$\left|\Delta r - \widehat{\Delta r}\right| \leq \overline{B}^{\pi^E} B^{\widehat{\pi}^E}\zeta + \gamma\left|\left(P_{\mathcal{T}} - \widehat{P_{\mathcal{T}}}\right)\left(V^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)} + V^r\right)\right|. \tag{22}$$

Note that the L_∞-norms of value function $\|V^{r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}\|_\infty \leq \frac{R_{\max}}{1-\gamma}$ and $\|V^r\|_\infty \leq \frac{R_{\max}}{1-\gamma}$. Then, by Lemma B.13, $\|\zeta\|_\infty \leq \frac{2R_{\max}}{1-\gamma}$. ☐

Theorem 4.2. *(Sample Complexity of the IRC Solver). If an IRC solver stops at iteration K with accuracy ε_K, then with probability at least $1-\delta$ it converges, with a number of samples upper bounded by:*

$$n \leq \tilde{\mathcal{O}}\left(\frac{4\gamma^2 R_{\max}^2 SA}{(1-\gamma)^4\varepsilon_K^2}\right). \tag{23}$$

Proof. Compare Lemma B.15 for the IRC solver with Lemma B.9 for the IRL solver. Using the proof techniques from (Metelli et al., 2021, Theorem 5.1), we derive the following sample complexity for the IRC solver. ☐

Remark B.16. Compared to the IRL solver, the IRC solver utilizes the known reward signals for constraint inference. This method treats r as "imperfect" rewards and learns a correction term $\Delta r(s,a)$ to incorporate constraint signals. Note that the IRC solver does not explicitly model the constraints during inference. Instead, the solver considers an unconstrained RL problem instead of CRL (Liu et al., 2021) during policy update.

B.6 Theoretical Results of the ICRL Solver

In this part, we provide additional discussion regarding the ICRL solver for constraint inference.

Inverse Constrained Reinforcement Learning (ICRL). ICRL solvers infer the constraint respected by the expert agents from their demonstration data. An ICRL solver admits the following assumptions.

Assumption B.17. *Either of the following two statements holds:*
(i) The constraint in Eq. (PI) is a hard constraint such that $\epsilon = 0$;
(ii) The constraint in Eq. (PI) is a soft constraint such that $\epsilon > 0$, and the expert policy is deterministic.

Lemma 4.3. *(Feasible cost set implicit (Yue et al., 2024, Lemma 4.3)).* Under Assumption B.17, and let $\mathbb{S}_{\text{ICRL}}(\mathcal{M}, \pi^E, r)$ be an ICRL solver, then c is a feasible cost, i.e., $c \in \mathcal{C}_{\mathbb{S}_{\text{ICRL}}}$ if and only if $\forall(s,a) \in \mathcal{S}\times\mathcal{A}$:

(i) if $\pi^E(a|s) > 0$, i.e., (s,a) follows the expert policy:

$$Q^{c,\pi^E}_{\mathcal{M}\cup c}(s,a) - V^{c,\pi^E}_{\mathcal{M}\cup c}(s) = 0. \tag{24}$$

(ii) if $\pi^E(a|s) = 0$ and $A^{r,\pi^E}_{\mathcal{M}\cup c}(s,a) > 0$, i.e., (s,a) violates the constraint:

$$Q^{c,\pi^E}_{\mathcal{M}\cup c}(s,a) - V^{c,\pi^E}_{\mathcal{M}\cup c}(s) > 0. \tag{25}$$

(iii) if $\pi^E(a|s) = 0$ and $A^{r,\pi^E}_{\mathcal{M}\cup c}(s,a) \leq 0$, i.e., (s,a) is in the non-critical region:

$$Q^{c,\pi^E}_{\mathcal{M}\cup c}(s,a) - V^{c,\pi^E}_{\mathcal{M}\cup c}(s) \leq 0. \tag{26}$$

Lemma B.18. *Let $\mathbb{S}_{ICRL}(\mathcal{M}, \pi^E, r)$ be an ICRL solver. A Q-function satisfies the condition of Lemma 4.3 if and only if there exists $\zeta \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}_{\geq 0}$ and $V^c \in \mathbb{R}^{\mathcal{S}}$ such that:*

$$Q^{c,\pi^E}_{\mathcal{M}\cup c} = A^{r,\pi^E}_{\mathcal{M}\cup c}\zeta + EV^c, \tag{27}$$

where the expansion operator E satisfies $(Ef)(s) = f(s,a)$.

Lemma B.19. *(Feasible cost set explicit (Yue et al., 2024, Lemma 4.4)). Let $\mathbb{S}_{ICRL}(\mathcal{M}, \pi^E, r)$ be an ICRL solver. c is a feasible cost, i.e., $c \in \mathcal{C}_{\mathbb{S}_{ICRL}}$ if and only if there exist $\zeta \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}_{\geq 0}$ and $V^c \in \mathbb{R}^{\mathcal{S}}$ such that:*

$$c = A^{r,\pi^E}_{\mathcal{M}\cup c}\zeta + (E - \gamma P_{\mathcal{T}})V^c. \tag{28}$$

Similarly, we show how the estimation error on the environmental dynamic and on the expert policy propagates to the cost function.

Lemma B.20. *(Error Propagation)(Yue et al., 2024, Lemma 4.5). Let $\mathbb{S}_{ICRL}(\mathcal{M}, \pi^E, r)$ and $\widehat{\mathbb{S}}_{ICRL}(\widehat{\mathcal{M}}, \widehat{\pi}^E, r)$ be two instances of ICRL solvers. For any $c \in \mathcal{C}_{\mathbb{S}_{ICRL}}$ satisfying $c = A^{r,\pi^E}_{\mathcal{M}\cup c}\zeta + (E - \gamma P_{\mathcal{T}})V^c$ and $\|c\|_\infty \leq C_{\max}$, there exists $\widehat{c} \in \mathcal{C}_{\widehat{\mathbb{S}}_{ICRL}}$ such that element-wise it holds that:*

$$|c - \widehat{c}| \leq \gamma \left|(P_{\mathcal{T}} - \widehat{P_{\mathcal{T}}})V^c\right| + \left|A^{r,\pi^E}_{\mathcal{M}\cup c} - A^{r,\widehat{\pi}^E}_{\widehat{\mathcal{M}}\cup r}\right|\zeta. \tag{29}$$

Furthermore, $\|V^c(s)\|_\infty \leq C_{\max}/(1-\gamma)$ and $\|\zeta\|_\infty \leq C_{\max}/\min^+_{(s,a)}|A^{r,\pi^E}_{\mathcal{M}\cup c}|$.

Lemma B.21. *(Yue et al., 2024, Lemma 4.6) For a given policy π, let $A^{r,\pi}_{\mathcal{M}\cup r}$ denote the reward advantage function based on the original MDP $\mathcal{M}$. For an estimated policy $\widehat{\pi}$, let $A^{r,\widehat{\pi}}_{\widehat{\mathcal{M}}\cup c}$ denote the reward advantage function based on the estimated MDP $\widehat{\mathcal{M}}$. Then, we have*

$$\left|A^{r,\pi}_{\mathcal{M}\cup c} - A^{r,\widehat{\pi}}_{\widehat{\mathcal{M}}\cup c}\right| \leq \frac{2\gamma}{1-\gamma}\left|(\widehat{P_{\mathcal{T}}} - P_{\mathcal{T}})V^{r,\widehat{\pi}}_{\widehat{\mathcal{M}}}\right| + \frac{\gamma(1+\gamma)}{1-\gamma}\left|(\pi - \widehat{\pi})P_{\mathcal{T}}V^{r,\pi}_{\mathcal{M}}\right|. \tag{30}$$

Theorem 4.5. *(Sample Complexity of ICRL solver(Yue et al., 2024, Theorem C.9)). If an ICRL solver terminates at iteration K with the updated accuracy ε_K, then with probability at least $1-\delta$, it converges with a number of samples upper bounded by*

$$n \leq \widetilde{\mathcal{O}}\left(\frac{\gamma^2\sigma^2 SA}{(1-\gamma)^6\varepsilon_K^2}\right). \tag{31}$$

where $\sigma = \sqrt{3}\gamma C_{\max}\left(R_{\max}(3+\gamma)/\min^+\left|A^{r,\pi^E}_{\mathcal{M}\cup c}\right| + (1-\gamma)\right)$.

Remark B.22. Compared with the IRC solver, the ICRL solver considers a CMDP environment instead of an MDP one, so it explicitly models constraints during policy updates.

B.7 Numerical Analysis of the Example in Figure 1

As illustrated in Figure 1, the basic environment contains 2×5 grids, where the agent navigates from the starting location $(0, 0)$ (the left bottom corner) to the target location $(0, 4)$ (the left top corner) with four possible actions, i.e., going up, down, left and right. In the source environment, $(0, 4)$ (colored blue) is assigned with a *reward* of 10, $(0, 2)$ (colored orange) contains an absolute constraint that must not be accessed, while all other locations are assigned with 0 rewards and 0 costs. Discount factor $\gamma = 0.7$ and the agent has a probability of 0.1 to execute a random action. Figure 5 illustrates the source environment, optimal value function, and optimal policy at each state.

Under the current reward function $r((0, 4)) = 10$, the agent chooses to go upward at $(0, 1)$, which violates the hard constraint and deviates from the expert policy. In order to guide the agent to go right at $(0, 1)$ instead of going up, a feasible reward correction is $\Delta r((0, 2)) = -3$, which, together with reward function $r((0, 4)) = 10$, ensures the optimality of the expert policy.

Suppose we have a target environment, with $r'((0, 4)) = 20$, and a stochasticity of 0.4 to the right column of the environment (the stochasticity of the left column remains 0.1), under which the reward value function is shown in Figure 6 (middle). r' with $\Delta r((0, 2)) = -3$ generate the policy in Figure 6 (right). We observe that $\Delta r((0, 2)) = -3$ is not enough to penalize the going up action at $(0, 1)$.

Next, we validate Theorem 5.3. At state $(0, 1)$, a^E = Right, a^C = Up, a^O_1 = Down, a^O_2 = Left. Through numerical calculation, we derive the following results which match the results of Theorem 5.3. Here, we simplify $Q^{r+\Delta r,\pi^E}_{\mathcal{M}\cup c}$ as $Q^{r+\Delta r}$.

- $Q^{r+\Delta r}(a^E) - Q^{r+\Delta r}(a^C) = 0.556 \geq 0$;

- $Q^{r+\Delta r}(a^E) - Q^{r+\Delta r}(a_1^O) = 1.011 \geq 0$; $Q^{r+\Delta r}(a^E) - Q^{r+\Delta r}(a_2^O) = 0.609 \geq 0$;
- $Q^{r+\Delta r}(a^E) - Q^{r+\Delta r}(a^C) = 0.556 < (Y')^{-1}(Y - Y')Q^{r+\Delta r}[a^C - a^E] = 3.446$;
- $Q^{r+\Delta r}(a_1^O) - Q^{r+\Delta r}(a^C) = -0.455 < (Y')^{-1}(Y - Y')Q^{r+\Delta r}[a^C - a_1^O] = 3.711$;

  $Q^{r+\Delta r}(a_2^O) - Q^{r+\Delta r}(a^C) = -0.053 < (Y')^{-1}(Y - Y')Q^{r+\Delta r}[a^C - a_2^O] = 3.565$.

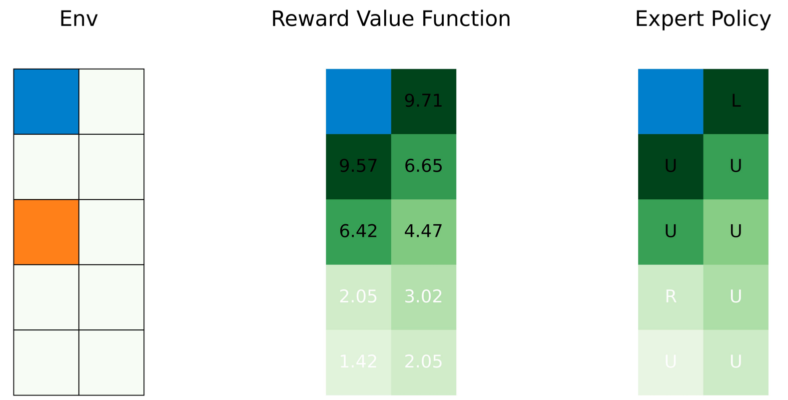

Figure 5: The basic environment (left), the source reward value function under r and $P_{\mathcal{T}}$ (middle), and the expert policy of the example in Figure 1 (right).

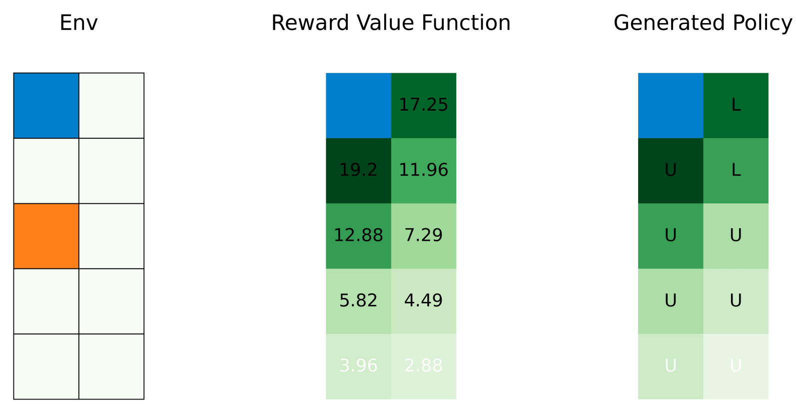

Figure 6: The basic environment (left), the target reward value function under r' and $P'_{\mathcal{T}}$ (middle), and the generated policy (right).

B.8 Theoretical Results of Transferability in Safety

B.8.1 The Hard Constraint Scenario

Lemma 5.2. *Suppose a hard constraint scenario. For any $(s', a') \in \mathcal{G}$, the feasible cost function $\hat{c}$ inferred by the ICRL solver (3.1) can prevent the visitation to (s', a') in the target CMDP.*

Proof. For any $(s', a') \in \mathcal{G}$ and efficiently small cost estimation error ε, there must be $\hat{c}(s', a') > 0$ by the ICRL solver, which bans action a' at state s'. When transferred in a hard constraint scenario to a target CMDP, this $\hat{c}(s', a')$ will also ban this action a' at state s', since any policy with $\pi(a'|s') > 0$ definitely leads to a violation of the hard constraint. $\square$

Theorem 5.3. *Suppose a hard constraint scenario. At state s, let a^E denote the expert action , a^C denote the action that satisfies $(s, a^C) \in \mathcal{G}$ and a^O denote the other actions. $\forall r' \in [0, R_{\max}]^{\mathcal{S}\times\mathcal{A}}$ and $\forall P'_{\mathcal{T}} \in \Delta^{\mathcal{S}}_{\mathcal{S}\times\mathcal{A}}$, if $\exists s \in \mathcal{S}$, $\forall\, a^E, a^O \in \mathcal{A}$, $\exists\, a^C \in \mathcal{A}$ that satisfies the following condition, then the reward correction term Δr constructed by such Q-functions violates safety in the target CMDP,*

$$Q^{r+\Delta r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}(\mathbf{e}_{(s,a^E)} - \mathbf{e}_{(s,a^C)}) \geq 0, Q^{r+\Delta r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}(\mathbf{e}_{(s,a^E)} - \mathbf{e}_{(s,a^O)}) \geq 0,$$
$$Q^{r+\Delta r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}(\mathbf{e}_{(s,a^E)} - \mathbf{e}_{(s,a^C)}) < (Y')^{-1}\left[r' - r + (Y - Y')Q^{r+\Delta r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}\right](\mathbf{e}_{(s,a^C)} - \mathbf{e}_{(s,a^E)}),$$
$$Q^{r+\Delta r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}(\mathbf{e}_{(s,a^O)} - \mathbf{e}_{(s,a^C)}) < (Y')^{-1}\left[r' - r + (Y - Y')Q^{r+\Delta r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}\right](\mathbf{e}_{(s,a^C)} - \mathbf{e}_{(s,a^O)}),$$

where $Y = (I_{\mathcal{S}\times\mathcal{A}} - \gamma P_{\mathcal{T}}\pi^E)$, $Y' = (I_{\mathcal{S}\times\mathcal{A}} - \gamma P'_{\mathcal{T}}(\pi')^E)$ and $\mathbf{e}_{(s,a)}$ denotes a vector with value of 1 at the index (s, a) and 0 elsewhere.

Proof. For better readability, we denote $Q^{r'} = Q^{r'+\Delta r,(\pi')^E}_{\mathcal{M}'\cup(r'+\Delta r)}$ and $Q^r = Q^{r+\Delta r,\pi^E}_{\mathcal{M}\cup(r+\Delta r)}$. Since, at each state $s \in \mathcal{S}$, the agent chooses its action based on the maximum Q-function, we study the Q-function in the source and target CMDP.

$$Q^r(s,a) = (I_{\mathcal{S}\times\mathcal{A}} - \gamma P_{\mathcal{T}}\pi^E)^{-1} r^c(s,a) = (I_{\mathcal{S}\times\mathcal{A}} - \gamma P_{\mathcal{T}}\pi^E)^{-1}[r(s,a) + \Delta r(s,a)]$$
$$Q^{r'}(s,a) = (I_{\mathcal{S}\times\mathcal{A}} - \gamma P'_{\mathcal{T}}(\pi')^E)^{-1}(r^c)'(s,a) = (I_{\mathcal{S}\times\mathcal{A}} - \gamma P'_{\mathcal{T}}(\pi')^E)^{-1}[r'(s,a) + \Delta r(s,a)]$$

Due to the optimality of the expert policy, the Q-function of the expert action is the maximum in the source CMDP. We next show how the Q-function in the target CMDP is shifted.

$$Q^r(s,a) = (I_{\mathcal{S}\times\mathcal{A}} - \gamma P_{\mathcal{T}}\pi^E)^{-1}\left[r + \Delta r\right](s,a) \tag{32}$$
$$\begin{aligned} Q^{r'}(s,a), &= (I_{\mathcal{S}\times\mathcal{A}} - \gamma P'_{\mathcal{T}}(\pi')^E)^{-1}\left[r' + \Delta r\right](s,a) \\ &= (I_{\mathcal{S}\times\mathcal{A}} - \gamma P'_{\mathcal{T}}(\pi')^E)^{-1}\left[r' - r + (I_{\mathcal{S}\times\mathcal{A}} - \gamma P_{\mathcal{T}}\pi^E)Q^r\right](s,a) \\ &= (Y')^{-1}\left[(r' - r) + YQ^r\right](s,a) \\ &= (Y')^{-1}\left[(r' - r) + (Y - Y' + Y')Q^r\right](s,a) \\ &= (Y')^{-1}\left[(r' - r) + (Y - Y')Q^r\right](s,a) + Q^r(s,a). \end{aligned} \tag{33}$$

Based on Eq. (33), we have

$$\begin{aligned} Q^{r'}(s,a^E) &= (Y')^{-1}\left[r' - r + (Y - Y')Q^r\right](s,a^E) + Q^r(s,a^E) \\ Q^{r'}(s,a^C) &= (Y')^{-1}\left[r' - r + (Y - Y')Q^r\right](s,a^C) + Q^r(s,a^C) \\ Q^{r'}(s,a^O) &= (Y')^{-1}\left[r' - r + (Y - Y')Q^r\right](s,a^O) + Q^r(s,a^O). \end{aligned} \tag{34}$$

The first two inequalities result from Lemma 4.1, aiming to ensure the optimality of expert policy. The third and fourth inequalities are derived by substituting Eqs. (34) into $Q^{r'}(s,a^E) < Q^{r'}(s,a^C)$ and $Q^{r'}(s,a^O) < Q^{r'}(s,a^C)$ to construct target Q-functions that drive the agent choose constraint-violating actions a^C.

Given $Y = (I_{\mathcal{S}\times\mathcal{A}} - \gamma P_{\mathcal{T}}\pi^E)$ and $Y' = (I_{\mathcal{S}\times\mathcal{A}} - \gamma P'_{\mathcal{T}}(\pi')^E)$, we have

$$\begin{aligned} Y - Y' &= \gamma(P'_{\mathcal{T}}(\pi')^E - P_{\mathcal{T}}\pi^E) \\ &= \gamma[P'_{\mathcal{T}}(\pi')^E - P_{\mathcal{T}}(\pi')^E + P_{\mathcal{T}}(\pi')^E - P_{\mathcal{T}}\pi^E] \\ &= \underbrace{\gamma[(P'_{\mathcal{T}} - P_{\mathcal{T}})(\pi')^E]}_{\text{A: transition difference term}} + \underbrace{\gamma[P_{\mathcal{T}}((\pi')^E - \pi^E)]}_{\text{B: expert policy difference term}}. \end{aligned} \tag{35}$$

Based on (35), we can further split the increment from $Q^r(s,a)$ to $Q^{r'}(s,a)$ in (33) into:

$$Q^{r'}(s,a) - Q^r(s,a) = \left[\underbrace{(Y')^{-1}(r' - r)}_{\text{reward transfer shift}} + \underbrace{(Y')^{-1}AQ^r}_{\text{transition transfer shift}} + \underbrace{(Y')^{-1}BQ^r}_{\text{expert policy transfer shift}}\right](s,a) \tag{36}$$

□

B.8.2 EXTENSION TO THE SOFT CONSTRAINT SCENARIO

In this part, we extend our theory from the hard constraint scenario to the soft constraint scenario. Unlike the hard constraint scenario, where we construct a constraint set that a sensible agent should never visit, the soft constraint scenario is more complex. Given a threshold $\epsilon > 0$, no matter how large the constraint on a state-action pair (s, a) is, it is always possible to visit (s, a) after a sufficiently large number of steps, because the cost of visiting (s, a) can be reduced via the power of discount factor $\gamma < 1$.

For IRC solvers, in the soft constraint scenario, safety is also not guaranteed by the transferred reward correction term in the target environment. Lemma 5.3 still applies, as long as $\mathcal{G} \neq \varnothing$. Choosing a^C at state s and follows the expert policy $(\pi')^E$ in any other state definitely violates the constraint, since $A^{r,(\pi')^E}_{\mathcal{M}' \cup r'}(s, a) > 0$ and the target expert policy $(\pi')^E$ is optimal.

The primary difference between the hard and soft constraint scenarios is that the cost functions inferred by the ICRL solver may not be transferable. However, we demonstrate that the ICRL solver inherently offers better transferability in safety, because it is not affected by reward transfer shifts, compared with the IRC solver.

In the soft constraint scenario, the cost function $c(s, a)$ constructed by such Q-functions may not be transferable in the target CMDP, since for $(s, a) \in \mathcal{G}$, $Q^{c,(\pi')^E}_{\mathcal{M}' \cup c}(s, a) - V^{c,(\pi')^E}_{\mathcal{M}' \cup c}(s) > 0$ (situation (ii) in Lemma 4.3) is not necessarily satisfied.

Since at each state $s \in \mathcal{S}$, the agent examines whether this action is executable based on the cost Q-function, we study the cost Q-function in the source and target CMDP.

$$Q^c(s, a) = Q^{c,(\pi)^E}_{\mathcal{M} \cup c} = \left[(I_{\mathcal{S} \times \mathcal{A}} - \gamma P_{\mathcal{T}} \pi^E)^{-1} c\right](s, a) \tag{37}$$

$$Q^{c'}(s, a) = Q^{c,(\pi')^E}_{\mathcal{M}' \cup c} = \left[(I_{\mathcal{S} \times \mathcal{A}} - \gamma P'_{\mathcal{T}} (\pi')^E)^{-1} c\right](s, a) \tag{38}$$

Similar to the transferability of reward correction terms, we further discuss the influence of different reward functions, transition models, and expert policies on the transferability of cost functions. Since cost Q-functions of constraint-violating actions are larger than that of the expert action to ensure the optimality of expert policy, we next show how the cost Q-function in the target CMDP is shifted.

From (37) and (38), we obtain

$$\begin{aligned} Q^{c'}(s, a) &= [(Y')^{-1} Y Q^c](s, a) \\ &= [(Y')^{-1}(Y - Y' + Y') Q^c](s, a) \\ &= [(Y')^{-1}(Y - Y') Q^c](s, a) + Q^c(s, a) \end{aligned} \tag{39}$$

Given (35), the increment in cost Q-functions can be split into:

$$Q^{c'}(s, a) - Q^c(s, a) = \left[\underbrace{(Y')^{-1} A Q^c}_{\text{transition transfer shift}} + \underbrace{(Y')^{-1} B Q^c}_{\text{expert policy transfer shift}} \right](s, a) \tag{40}$$

By comparing (36) with (40), we can clearly see the difference in transferring reward correction terms and cost functions. Transferring the cost function by the ICRL solver is influenced by: 1) transition shift, and 2) expert policy shift. Transferring the reward correction term by IRC is influenced by: 1) reward shift, 2) transition shift, and 3) expert policy shift.

At state s, for $a^C \in \mathcal{G}$, we further distinguish two cases. Let a^{C_1} denote the action that satisfies $(s, a^{C_1}) \in \{(s, a) | A^{r,\pi^E}_{\mathcal{M} \cup c}(s, a) > 0\} \cap \{(s, a) | A^{r,(\pi')^E}_{\mathcal{M}' \cup c}(s, a) > 0\}$ and a^{C_2} denote the action that satisfies $(s, a^{C_2}) \in \{(s, a) | A^{r,\pi^E}_{\mathcal{M} \cup c}(s, a) > 0\} \backslash \{(s, a) | A^{r,(\pi')^E}_{\mathcal{M}' \cup c}(s, a) > 0\}$.

Based on Eq. (39), we have

$$Q^{c'}(s,a^E) = [(Y')^{-1}(Y-Y')Q^c](s,a^E) + Q^c(s,a^E)$$
$$Q^{c'}(s,a^O) = [Y')^{-1}(Y-Y')Q^c](s,a^O) + Q^c(s,a^O)$$
$$Q^{c'}(s,a^{C_1}) = [(Y')^{-1}(Y-Y')Q^c](s,a^{C_1}) + Q^c(s,a^{C_1})$$
$$Q^{c'}(s,a^{C_2}) = [(Y')^{-1}(Y-Y')Q^c](s,a^{C_2}) + Q^c(s,a^{C_2})$$
$$Q^{c'}(s,(a')^E) = [(Y')^{-1}(Y-Y')Q^c](s,(a')^E) + Q^c(s,(a')^E)$$

A sufficient condition that the ICRL solver fails would be choosing a^{C_1} in the target CMDP. Specifically, we should show the condition under which with new reward function r' and new transition $P'_{\mathcal{T}}$, $\exists s \in \mathcal{S}$, $\forall\, a^E \in \mathcal{A}$ and $\forall\, a^O \in \mathcal{A}$, $\exists\, a^{C_1} \in \mathcal{A}$: $Q^{c'}(s,(a')^E) > Q^{c'}(s,a^{C_1})$ (a^{C_1} does not violate the constraint but achieves higher rewards). We formally state this condition in the following lemma.

Lemma B.23. *$\forall r' \in [0, R_{\max}]^{\mathcal{S}\times\mathcal{A}}$ and $\forall P'_{\mathcal{T}} \in \Delta^{\mathcal{S}}_{\mathcal{S}\times\mathcal{A}}$, if $\exists s \in \mathcal{S}$, $\forall\, a^E \in \mathcal{A}$ and $\forall\, a^O \in \mathcal{A}$, $\exists\, a^{C_1} \in \mathcal{A}$ that satisfies the following condition, then the cost function c constructed by such cost Q-functions is non-transferable in the target CMDP,*

$$\begin{aligned}&Q^c(s,a^E) < Q^c(s,a^{C_1}), Q^c(s,a^E) < Q^c(s,a^{C_2}), Q^r(s,a^E) \geq Q^r(s,a^O)\\ &[(Y')^{-1}(Y-Y')Q^c](\mathbf{e}_{(s,a^{C_1})} - \mathbf{e}_{(s,(a')^E)}) < 0\end{aligned} \tag{41}$$

where $\mathbf{e}_{(s,a)}$ denotes a vector with value of 1 at the index (s,a) and 0 elsewhere.

Proof. The first three inequalities result from Lemma 4.3, aiming to ensure the optimality of expert policy. For the fourth inequality, since we do not know $(a')^E$, we need to distinguish four situations, namely $(a')^E$ is a^E, a^{C_1}, a^{C_2} or a^O.

Distinguish four cases:

1. If $(a')^E = a^E$, $Q^{c'}(s,(a')^E) = Q^{c'}(s,a^E) > Q^{c'}(s,a^{C_1})$;
2. If $(a')^E = a^{C_1}$, this can not happen, because $A^{r,(\pi')^E}_{\mathcal{M}'\cup c}(s,a^{C_1}) > 0$ and $(\pi')^E$ is deterministic;
3. If $(a')^E = a^{C_2}$, $Q^{c'}(s,(a')^E) = Q^{c'}(s,a^{C_2}) > Q^{c'}(s,a^{C_1})$, there will always be such $Q^c(s,a^{C_2})$ and $Q^c(s,a^{C_1})$ that satisfy this because there's no requirements on the relative value between $Q^c(s,a^{C_2})$ and $Q^c(s,a^{C_1})$;
4. If $(a')^E = a^O$, $Q^{c'}(s,(a')^E) = Q^{c'}(s,a^O) > Q^{c'}(s,a^{C_1})$.

Hence, we only need to discuss case 1 and case 4.

Case 1. $Q^{c'}(s,a^E) > Q^{c'}(s,a^{C_1})$

equivalent to

$$[(Y')^{-1}(Y-Y')Q^c](s,a^E) + Q^c(s,a^E) > [(Y')^{-1}(Y-Y')Q^c](s,a^{C_1}) + Q^c(s,a^{C_1}) \tag{42}$$

equivalent to

$$\begin{aligned}Q^c(s,a^E) - Q^c(s,a^{C_1}) &> [(Y')^{-1}(Y-Y')Q^c](s,a^{C_1}) - [(Y')^{-1}(Y-Y')Q^c](s,a^E)\\ &= [(Y')^{-1}(Y-Y')Q^c](\mathbf{e}_{(s,a^{C_1})} - \mathbf{e}_{(s,a^E)})\end{aligned} \tag{43}$$

equivalent to (considering $Q^c(s,a^E) - Q^c(s,a^{C_1}) < 0$)

$$[(Y')^{-1}(Y-Y')Q^c](\mathbf{e}_{(s,a^{C_1})} - \mathbf{e}_{(s,a^E)}) < 0 \tag{44}$$

Case 4. $Q^{c'}(s,a^O) > Q^{c'}(s,a^{C_1})$

equivalent to

$$[(Y')^{-1}(Y-Y')Q^c](s,a^O) + Q^c(s,a^O) > [(Y')^{-1}(Y-Y')Q^c](s,a^{C_1}) + Q^c(s,a^{C_1}) \tag{45}$$

equivalent to

$$Q^c(s,a^O) - Q^c(s,a^{C_1}) > [(Y')^{-1}(Y-Y')Q^c](s,a^{C_1}) - [(Y')^{-1}(Y-Y')Q^c](s,a^O)$$
$$= [(Y')^{-1}(Y-Y')Q^c](\mathbf{e}_{(s,a^{C_1})} - \mathbf{e}_{(s,a^O)}) \tag{46}$$

equivalent to (considering $Q^c(s,a^O) - Q^c(s,a^{C_1}) < 0$)

$$[(Y')^{-1}(Y-Y')Q^c](\mathbf{e}_{(s,a^{C_1})} - \mathbf{e}_{(s,a^O)}) < 0 \tag{47}$$

Bring together case 1 and case 4, we obtain

$$[(Y')^{-1}(Y-Y')Q^c](\mathbf{e}_{(s,a^{C_1})} - \mathbf{e}_{(s,(a')^E)}) < 0 \tag{48}$$

□

Lemma B.24. *If two additional assumptions are satisfied: 1) the transition model does not change, i.e., $P'_{\mathcal{T}} = P_{\mathcal{T}}$; 2) the optimal policy of source CMDP is the optimal policy of target CMDP, i.e., $\Pi^*_{\mathcal{M}_c} \subseteq \Pi^*_{\mathcal{M}'_c}$, then the feasible cost function $\hat{c}$ inferred by the ICRL solver (3.1) can prevent any visitation to $(s',a') \in \mathcal{G} \neq \varnothing$ in the target CMDP.*

Proof. Suppose $(s',a') \in \mathcal{G} \neq \varnothing$, at state s', since a' can bring more rewards both in the source CMDP and in the target CMDP, a' should be abandoned in both CMDPs. This means, in a soft constraint scenario, the expert policy at state s' must reach the threshold (Yue et al., 2024, Lemma 4.2). Since the constraint condition in case two is the same between the source and target CMDP, for $\forall\theta \in (0,1]$, policy $\pi'(a|s') = \begin{cases} \theta & , a = a' \\ 1-\theta, & a \sim \pi^E = (\pi')^E \end{cases}$ violates the soft constraint in target CMDP. As a result, action a' is abandoned in the target CMDP after transferring. □

Remark B.25. We demonstrate the performance of the ICRL solver in soft constraint scenarios in two aspects. In aspect one, only reward functions are different between source and target. In aspect two, only transition dynamics are different between source and target.

Aspect One: In soft constraint scenarios, the ICRL solver still outperforms the IRC solver in the sense that it resists the variation between source and target reward functions. Table 2 illustrates whether the inferred reward correction terms by the IRC solver violate the constraint in the target environment (safe ✓ or not ✗). We can see that with the increase in reward functions, the inferred reward correction terms tend to become unsafe while the inferred cost functions by ICRL solvers are safe still. Threshold $\epsilon = 0.015$ and ground-truth costs are 1.

Table 2: Safety (safe ✓ or not ✗) of inferred reward correction terms by the IRC solver under different rewards in the target environments of Gridworld-1. T represents the terminal location (6,6) in Gridworld-1.

different $r'(T)$ ↑	$r'(T)=1$	$r'(T)=1.2$	$r'(T)=1.4$	$r'(T)=1.6$	$r'(T)=1.8$	$r'(T)=2$
IRC solver	✓	✓	✓	✗	✗	✗
ICRL solver	✓	✓	✓	✓	✓	✓

Aspect Two: In soft constraint scenarios, ICRL solver can violate constraints due to compensated penalization by new transition dynamics. We provide a case study to help the readers better understand this aspect. Reconsider the example in Figure 1 in a soft constraint scenario. We only set $c((0,2)) = 1$ and $\epsilon = 0.8 \times (0.7)^{-2} > 0$. Suppose in the source environment, there is no randomness for any chosen actions. The true cost function the expert follows is $c^E((0,2)) = 1$. In this case, the shortest path, i.e., going straight upward from (0,0) to (0,5), is forbidden by the expert because the path induces a cumulative cost of $1 \times (0.7)^{-2} > \epsilon$. One choice of the feasible cost function is $c((0,2)) = 0.85$, since $0.85 \times (0.7)^{-2} > \epsilon$ is sufficient to ban the shortest path. Now we transfer $c((0,2)) = 0.85$ from the source to the target environment. Suppose the target environment only differs from the source environment in the transition model. In the target environment, if the location (0,1) alters its transition model to be $P_{\mathcal{T}}((0,2)|(0,1),UP) = 0.9$ and $P_{\mathcal{T}}((1,1)|(0,1),UP) = 0.1$.

The inferred optimal policy based on $c((0,2)) = 0.85$ visits (0,2) because the shortest path has a cumulative cost of $0.85 \times 0.9 \times (0.7)^{-2} = 0.765 \times (0.7)^{-2} < \epsilon$ but should be forbidden by the expert policy because the shortest path has a cumulative ground-truth cost of $0.9 \times (0.7)^{-2} > \epsilon$.

B.9 Theoretical Results of Transferability in Optimality

Proposition B.26. *Let $H_\gamma := 1/(1-\gamma), R := \max_{r\in\mathbb{R}} \|\breve{r}\|_\infty$,and $D = \max_{\breve{r},r'\in\mathbb{R}} \|r - r'\|_2$. Suppose that $\alpha < 2D$,then for the Shannon entropy regularization, η and $\sigma_\mathcal{R}$ can be obtained from (Schlaginhaufen & Kamgarpour, 2024, Proposition D.8):*

$$\eta = \alpha\mu_{min}/H_\gamma^2 \quad and \quad \sigma_\mathcal{R} = \frac{\mu_{min}\left(1-\frac{\alpha}{2D}\right)\exp\left(\frac{-2RH_\gamma}{\alpha}\right)}{D|\mathcal{S}||\mathcal{A}|^{2+H_\gamma}}. \tag{49}$$

Lemma B.27. *Suppose Assumptions 5.4 hold, and let $r, r' \in \mathbb{R}$. Then, we have*

$$\frac{\sigma_\mathcal{R}}{2}\|[r]_\mathcal{U} - [r']_\mathcal{U}\|_2^2 \le \ell(r', \mathsf{RL}(r)) = D_h(\mathsf{RL}(r), \mathsf{RL}(r')) \le \frac{1}{2\eta}\|[r]_\mathcal{U} - [r']_\mathcal{U}\|_2^2,$$

for some problem-dependent constant $\sigma_\mathcal{R} > 0$.

Proof. This lemma directly comes from (Schlaginhaufen & Kamgarpour, 2024, Lemma 3.5). □

Lemma B.28. *Consider $x, y \in \mathbb{R}^n$ and two subspaces $\mathcal{V}, \mathcal{W} \subset \mathbb{R}^n$ of dimension $m < n$.Then,*

$$\|[x]_\mathcal{W} - [y]_\mathcal{W}\|_2 \le \|\Pi_\mathcal{W} - \Pi_\mathcal{W}\| \cdot \|x - y\|_2 + \|[x]_\mathcal{V} - [y]_\mathcal{V}\|_2,$$

where $\|\Pi_\mathcal{W} - \Pi_\mathcal{V}\| = \sin(\theta_{\max}(\mathcal{V}, \mathcal{W}))$.

Proof. This lemma directly comes from (Schlaginhaufen & Kamgarpour, 2024, Proposition D.10). □

Definition B.29. *(Cost equivalence).* We extend the results of reward equivalence (feasible reward set) in IRL settings (Schlaginhaufen & Kamgarpour, 2024) to cost equivalence (feasible cost set) in ICRL settings. Given a linear subspace $\mathcal{V} \subseteq \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$, the quotient space $\mathbb{R}^{\mathcal{S}\times\mathcal{A}}/\mathcal{V}$ is the set of all equivalence classes $[r - \lambda^* c]_\mathcal{V} := \{r - \lambda^* c' \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}} : \lambda^*(c' - c) \in \mathcal{V}\}$. On quotient space $\mathbb{R}^{\mathcal{S}\times\mathcal{A}}/\mathcal{V}$,the quotient norm $\|[x]_\mathcal{V}\|_2 := \min_{v\in\mathcal{V}}\|x + v\|_2 = \|\Pi_{\mathcal{V}^\perp} x\|_2$ and we say that c and $\hat{c}$ are close in $\mathbb{R}^{\mathcal{S}\times\mathcal{A}}/\mathcal{V}$ given λ^*, if $\|[r - \lambda^* c]_\mathcal{V} - [r - \lambda^* c']_\mathcal{V}\|_2 = \|[(r - \lambda^* c) - (r - \lambda^* c')]_\mathcal{V}\|_2 = \|[\lambda^*(c' - c)]_\mathcal{V}\|_2$ is small. Moreover, the expert's cost is said to be recovered by $\hat{c}$ up to some equivalence class $[\cdot]_\mathcal{V}$ if $r - \lambda^*\hat{c} \subseteq [r - \lambda^* c^E]_\mathcal{V}$. In this paper, we consider the equivalence relations induced by the subspace of potential shaping transformations (Ng et al., 1999), i.e., $\mathcal{V} = \mathcal{U}_{P_\mathcal{T}} := \mathrm{im}(E - \gamma P_\mathcal{T})$. This subspace represents the feasible cost set defined in (28).

Theorem 5.7. *Let $P'_\mathcal{T}$ be a transition law in the target environment and $d_1 = \|[c^E - \hat{c}]_{\mathcal{U}_{P'_\mathcal{T}}}\|_2$. Suppose that Assumption 5.4 holds. If $\ell_{P_\mathcal{T}}^{r',(\lambda')^*}(\hat{c}, \mathsf{CRL}(c^E)) \le \varepsilon_1$, $\hat{c}$ is ε-transferable to $P'_\mathcal{T}$ with*

$$\varepsilon = 2\max\left\{d_1^2 \sin\left(\theta_{\max}({P_\mathcal{T}}', P_\mathcal{T})\right)^2/2, 2\varepsilon_1/\sigma_\mathcal{R}\right\}/\eta, \tag{50}$$

where $\sigma_\mathcal{R}$ and η are regularity constants, given in Appendix B.26.

Proof. For better readability, we denote $(\hat{r})' = r' - (\lambda')^*\hat{c}$ and $(\breve{r}')^E = r' - (\lambda')^* c^E$. It follows from Lemma B.27 that $\left\|[(\breve{r}')^E]_{\mathcal{U}_{P_\mathcal{T}}} - [(\hat{r})']_{\mathcal{U}_{P_\mathcal{T}}}\right\|_2 \le \sqrt{2\varepsilon_1/\sigma_\mathcal{R}}$. By choosing the closest $(\breve{r}')^E$ and $(\hat{r})'$ in subspace $\mathcal{U}_{P'_\mathcal{T}}$ in Lemma B.28, we then have that

$$\begin{aligned}\left\|[(\breve{r}')^E]_{\mathcal{U}_{P'_\mathcal{T}}} - [(\hat{r})']_{\mathcal{U}_{P'_\mathcal{T}}}\right\|_2 &\le \sin\left(\theta_{\max}(P'_\mathcal{T}, P_\mathcal{T})\right)\left\|[(\breve{r}')^E - (\hat{r})']_{\mathcal{U}_{P'_\mathcal{T}}}\right\|_2 + \left\|[(\breve{r}')^E]_{\mathcal{U}_{P_\mathcal{T}}} - [(\hat{r})']_{\mathcal{U}_{P_\mathcal{T}}}\right\|_2 \\ &\le d_1 \sin\left(\theta_{\max}(P'_\mathcal{T}, P_\mathcal{T})\right) + \sqrt{2\varepsilon_1/\sigma_\mathcal{R}}\end{aligned} \tag{51}$$

Hence, applying Lemma B.27 again yields

$$
\begin{aligned}
\ell^{r',(\lambda')^*}_{P'_{\mathcal{T}}}\left(\hat{c}, \mathsf{CRL}_{P'_{\mathcal{T}}}(c^E)\right) &\leq \frac{1}{2\eta}\left\|[(\breve{r}')^E]_{\mathcal{U}_{P'_{\mathcal{T}}}} - [(\hat{r})']_{\mathcal{U}_{P'_{\mathcal{T}}}}\right\|_2^2 \\
&\leq \frac{\left(d_1 \sin\left(\theta_{\max}(P_{\mathcal{T}}', P_{\mathcal{T}})\right) + \sqrt{2\varepsilon_1/\sigma_{\mathcal{R}}}\right)^2}{2\eta} \\
&\leq \frac{2\max\left\{d_1^2 \sin\left(\theta_{\max}(P_{\mathcal{T}}', P_{\mathcal{T}})\right)^2/2, 2\varepsilon_1/\sigma_{\mathcal{R}}\right\}}{\eta}. \qquad (52)
\end{aligned}
$$

□

C Comparison on Continuous Environments

For continuous environments, we apply an offline setting to compare the transferability of constraint knowledge inferred by IRC or ICRL solvers. The offline setting is different from the online setting in discrete environments. The expert policy for online estimation is replaced by expert demonstrations in a given dataset. For ICRL, The goal is to recover the minimum constraint set that best explains the expert data. Existing ICRL works commonly follow the Maximum Entropy framework (Malik et al., 2021). IRC solvers in this setting follow the same framework but solve a bi-level optimization problem (Hugessen et al., 2024).

We build on the codebases from Hugessen et al. (2024) and Liu et al. (2023) to compare the transferability performance of the IRC and ICRL solvers. We adapt the code to enable both solvers to infer constraint knowledge—such as reward correction terms or cost functions—within the source environment while evaluating the feasibility of this knowledge in the target environment. Using the blocked half-cheetah environment as a testbed, we report the results with mean ± standard deviation in Figure 7 with three random seeds. The definition of metrics and detailed source and target environment specifications are explained in Section D.

We find that IRC solver has better training efficiency in the source environment, i.e., achieving zeo violation rate with considerate feasible rewards more quickly than the ICRL solver. However, after transferring constraint knowledge into the target environment, inferred correction terms by the IRC solver fail to ensure safety (avoid constraint violation) while the cost function inferred by the ICRL solver has better generalizability.

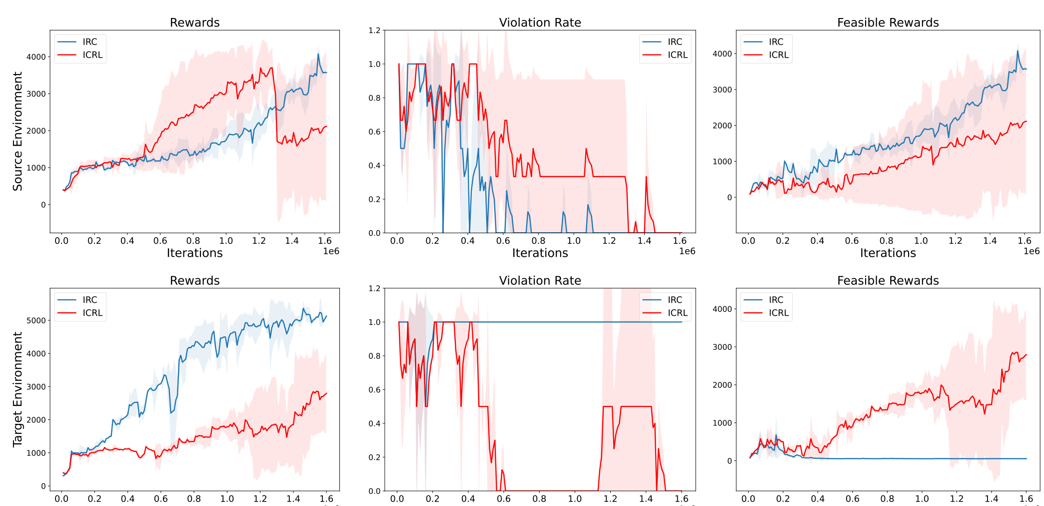

Figure 7: Training curves of rewards (left), violation rate (middle), and feasible rewards (right) for the ICRL (red) and IRC (blue) solvers. The top row shows the results for the source environment, and the bottom row shows the results for the target environment.

D EXPERIMENTAL DETAILS

We ran experiments on a desktop computer with Intel(R) Core(TM) i5-14400F and NVIDIA GeForce RTX 4060 Ti.

Details about Gridworld. In this paper, we construct a map with dimensions of 7×7 units and define four distinct settings, as shown in Figure 2. Locations are represented by two coordinates, with the first corresponding to the vertical axis and the second to the horizontal axis. The agent's objective is to navigate from a starting point to a target location while avoiding specified constraints. The agent begins in the bottom-left corner at position $(0, 0)$ and has eight possible actions: four cardinal directions (up, down, left, right) and four diagonal directions (upper-left, lower-left, upper-right, lower-right). The target location and reward are positioned in the upper-right cell $(6, 6)$ in the first, second, and fourth Gridworld environments, while in the third environment, the target is located in the upper-left cell $(6, 0)$. If the environment has a stochasticity of p, the agent has a probability of p to move randomly in any feasible direction, with each direction having a probability of $p/$num_of_actions. The reward is only provided at the target cell, with all other cells yielding zero reward. A cost of 1 is incurred if the agent enters a constrained location. Each policy rollout continues for a maximum of 50 time steps. In Figure 4, we present the mean and the 68% confidence interval (1-sigma error bar), calculated using three random seeds. Table 3 presents utilized hyperparameters in Gridworld experiments.

Table 3: List of the utilized hyperparameters in Gridworld environment.

Parameters	Gridworld
Max Episode Length	50
Discount Factor	0.7
Stopping Threshold	0.001
Stochasticity	0.05
Nu Max Clamp	1
Penalty Initial Value	0.1
Penalty Learning Rate	0.1
Source Terminal Rewards	1,1,1,1
Target Terminal Rewards	2,7,7,15
Ground-truth Costs	1

Details about Half-Cheetah The Blocked Half-Cheetah task is built on Mujoco, where the agent controls a two-legged robot. The reward is determined by the distance the robot travels between consecutive time steps, penalized by the magnitude of the input action. Each episode ends after a maximum of 5000 time steps. To impose a constraint, we block the region where the X-coordinate should satisfy $x\_pos \leq -3$, restricting the robot's movement to the region where the X-coordinate is between -3 and $+\infty$. The source environment follows the setup described above. In the target environment, rewards are scaled by a factor of 1.1, and Gaussian noise with a mean of 0 and a standard deviation of 0.1 is added to each observation. We utilize three metrics for evaluations: 1) rewards are defined as the total returns for an episode, regardless of constraints; 2) feasible rewards are the aggregated returns for an episode up to the first constraint violation; 3) the violation rate is calculated as the percentage of episodes in which one or more constraint violations occur. Table 4 presents utilized hyperparameters in Gridworld experiments. Other hyperparameters follow previous codebases (Hugessen et al., 2024; Liu et al., 2023).

Table 4: List of the utilized hyperparameters in Half-Cheetah environment.

Parameters	Half-Cheetah
Training Epoch	320
Testing Epoch	320
Max Episode Length	5000
IRC Solver	IRL-base