Learning True Objectives: Linear Algebraic Characterizations of Identifiability in Inverse Reinforcement Learning

Mohamad Louai Shehab MLSHEHAB@UMICH.EDU
Antoine Aspeel ANTOINAS@UMICH.EDU
Nikos Aréchiga NIKOS.ARECHIGA@TRI.GLOBAL
Andrew Best ANDREW.BEST@TRI.GLOBAL
Necmiye Ozay NECMIYE@UMICH.EDU

Editors: A. Abate, K. Margellos, A. Papachristodoulou

Abstract

Inverse reinforcement Learning (IRL) has emerged as a powerful paradigm for extracting expert skills from observed behavior, with applications ranging from autonomous systems to human-robot interaction. However, the identifiability issue within IRL poses a significant challenge, as multiple reward functions can explain the same observed behavior. This paper provides a linear algebraic characterization of several identifiability notions for an entropy-regularized finite horizon Markov decision process (MDP). Moreover, our approach allows for the seamless integration of prior knowledge, in the form of featurized reward functions, to enhance the identifiability of IRL problems. The results are demonstrated with experiments on a grid world environment.

Keywords: Markov decision process, inverse reinforcement learning, identifiability

1. Introduction

Inverse reinforcement learning (IRL) is the problem of finding the reward function of an agent from its behavior Ng and Russell (2000). IRL has gained significant attention in the research community since having access to expert demonstrations can alleviate the burden of manually specifying a reward function Abbeel and Ng (2004) and improve generalizability. A primary problem with IRL is that it is fundamentally ill-posed. Indeed, there are multiple reward functions leading to any observed behavior. Prior work has generally dealt with this ambiguity in reward learning by using heuristics, e.g., Max Margin IRL Ratliff et al. (2006), Bayesian IRL Ramachandran and Amir (2007), Max Entropy IRL Ziebart et al. (2008), Relative Entropy IRL Boularias et al. (2011), and Deep Max Entropy IRL Wulfmeier et al. (2015) (see Arora and Doshi (2021) for a comprehensive overview). These approaches are well-suited for learning an imitation policy since the learned reward is guaranteed to induce a learned policy at least as good as the expert one. However, when IRL is used for behavior modeling Li et al. (2022); Ashwood et al. (2022); Babes et al. (2011); Ramponi et al. (2020); Jenner and Gleave (2021), or for policy transfer to novel environments Cao et al. (2021); Rolland et al. (2022); Fu et al. (2018), it becomes crucial to address the reward ambiguity problem. In such settings, finding one reward function that explains the agent's behavior is not enough since different reward functions can lead to different interpretations of the agent's preferences or completely different behaviors on a modified environment. Instead, it is necessary to find the set of all possible rewards Metelli et al. (2021) that can explain the behavior.

This leads to different notions of reward equivalence. For example, two rewards are said to be trajectory equivalent if they lead to the same distribution of trajectories under the optimal policy.

© 2024 M.L. Shehab, A. Aspeel, N. Aréchiga, A. Best & N. Ozay.

Equivalence classes of rewards allow us to formalize the concept of identifiability of an MDP as follows: identifiability holds when a reward can be identified *up to the corresponding equivalence class*. In this context, our contribution is two-fold. First, we derive linear algebraic characterizations of weak, almost-strong, and strong trajectory equivalence classes of a reward function. This leads to necessary and sufficient conditions for the corresponding notions of identifiability. Then, we show how incorporating prior knowledge —in the form of featurized reward functions— can be seamlessly integrated into the framework to enhance the identifiability of rewards in certain environments.

2. Preliminaries

2.1. Notation

We denote by $\mathbb{N}$ and $\mathbb{R}$ the sets of natural and real numbers, respectively. The identity matrix in $\mathbb{R}^{n\times n}$ is denoted by I_n, the zero matrix in $\mathbb{R}^{m\times n}$ is denoted by $0_{m\times n}$, and the vector of ones in $\mathbb{R}^n$ is denoted by $\mathbf{1}_n$. For matrices A and B, $\begin{bmatrix} A & B \end{bmatrix}$ is the horizontal concatenation of A and B. We denote by $\texttt{ker}(A)$ and $\texttt{ran}(A)$ the null space and the column span of the matrix A respectively. For a matrix A and a set X, AX is the set $\{Ax | x \in X\}$. For any two sets X and Y, $X \times Y$ is their Cartesian product and $X \oplus Y$ is their Minkowski sum. For a vector x, $x \oplus Y$ denotes $\{x\} \oplus Y$. We denote by $\texttt{dim}(V)$ the dimension of a vector space V. The cardinality of a set Ω is denoted by $|\Omega|$, and $\Delta(\Omega)$ denotes the set of probability measures over the set Ω. The support of a measure $\mu \in \Delta(\Omega)$ is the set $\texttt{support}(\mu) = \{x \in \Omega \mid \mu(x) > 0\}$. The "Dirac" distribution that sets a point mass at state $s \in \Omega$ is denoted $\delta_{s,\Omega} \in \Delta(\Omega)$. It will be denoted δ_s when Ω is clear from the context. The indicator function $\mathbb{1}(\cdot)$ is $\mathbb{1}(a = b) = 1$ if $a = b$, and 0 otherwise. Given a function $f : X \to Y$, and a set $A \subseteq X$, we denote by $f|_A$ the restriction of f to A.

2.2. Markov Decision Processes

A Markov Decision Process (MDP) is a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{T}, \mu_0, r, \gamma, T)$, where $\mathcal{S} = \{s^{(1)}, \ldots, s^{(n)}\}$ is a finite set of states with cardinality $|\mathcal{S}| = n$; $\mathcal{A} = \{a^{(1)}, \ldots, a^{(m)}\}$ is a finite set of actions with cardinality $|\mathcal{A}| = m$; $\mathcal{T} : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ is a Markov transition kernel; $\mu_0 \in \Delta(S)$ is an initial distribution over the set of states; $r : \mathcal{S}\times\mathcal{A} \to \mathbb{R}$ is a reward function (or reward for short); $\gamma \in [0, 1]$ is a discount factor; and $T \in \mathbb{N}$ is the non-negative time horizon. A policy $\pi_t : \mathcal{S} \to \Delta(\mathcal{A})$ is a function that describes an agent's behavior at time step t by specifying an action distribution at each state. We denote by $\pi = (\pi_t)_{t=0}^{T-1}$ the *time-varying* stochastic policy throughout the entire horizon. A trajectory τ (of length T) is an alternating sequence of states and actions (ending with a state), i.e., $\tau = (s_0, a_0, s_1, a_1, \ldots, s_{T-1}, a_{T-1}, s_T)$ with $s_t \in \mathcal{S}$ and $a_t \in \mathcal{A}$. Under a policy π, a trajectory τ occurs with probability

$$\mathbb{P}^{\pi}_{\mu_0}(\tau) = \mu_0(s_0) \prod_{t=0}^{T-1} \pi_t(a_t|s_t) \prod_{t=0}^{T-1} \mathcal{T}(s_{t+1}|s_t, a_t),$$

which depends on the distribution of initial states, the policy, and the Markov transition kernel. We consider the Maximum Entropy Reinforcement Learning (MaxEntRL) objective given by:

$$J_{\text{MaxEnt}}(\pi; r) = \mathbb{E}^{\pi}_{\mu_0}[\sum_{t=0}^{T-1} \gamma^t \Big(r(s_t, a_t) + \lambda \mathcal{H}(\pi_t(.|s_t)) \Big)], \tag{1}$$

where $\lambda > 0$ is a regularization parameter, and $\mathcal{H}(\pi_t(.|s_t)) = -\sum_{a \in \mathcal{A}} \pi_t(a|s_t) \log(\pi_t(a|s_t))$ is the entropy of the policy π_t. The expectation is with respect to the probability measure $\mathbb{P}^{\pi}_{\mu_0}$. We denote by Ω the support of $\mathbb{P}^{\pi}_{\mu_0}$. Similarly, we denote by $\Omega(s_0)$ the support of $\mathbb{P}^{\pi}_{\delta_{s_0}}$, for some $s_0 \in \texttt{support}(\mu_0)$. The reward of a trajectory τ is given by overloading the reward function $r(\tau) = \sum_{t=0}^{T-1} \gamma^t r(s_t, a_t)$. We define the optimal policy set Π^*_r, corresponding to a reward function r, as the set of maximizers of (1), i.e.,

$$\Pi^*_r = \arg\max_{\pi} J_{\text{MaxEnt}}(\pi; r). \tag{2}$$

The non-uniqueness of the optimal policy stems from the fact that the policy can be arbitrarily specified for the non-accessible states without changing the objective value. However, the policy is unique over the accessible state-action pairs Kim et al. (2021). To formalize this, we define the accessible states at time step t and those throughout the horizon T as:

$$\texttt{Access}_t = \{s \in \mathcal{S} \mid \mathbb{P}^{\pi}_{\mu_0}(s_t = s) > 0 \text{ for some policy } \pi\},$$
$$\texttt{Access} = \{(t, s) \in [0, T-1] \times S \mid s \in \texttt{Access}_t\},$$

respectively. When we restrict the policies in Π^*_r to the accessible states, we obtain a unique policy, denoted by $\pi^*_r|_{\texttt{Access}}$[1]. Since the trajectory distribution for a given policy depends only on the accessible states, we define the optimal trajectory distribution for a reward r as $p_r = \mathbb{P}^{\pi^*_r}_{\mu_0}$, where $\pi^*_r \in \Pi^*_r$ is arbitrary. In particular, p_r is the distribution of trajectories when using an optimal policy corresponding to r and starting from the support of μ_0. Finally, we define an *MDP Model* as a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{T}, \mu_0, R, \gamma, T)$ where R is a set of reward functions, and $\mathcal{S}$, $\mathcal{A}$, $\mathcal{T}$, μ_0, γ, and T are defined as for an MDP.

2.3. Reward identifiability and Equivalence Classes

As in many identification problems, rewards can only be identified up to an equivalence class. Roughly speaking, an MDP model is more identifiable when the equivalence class is smaller. In what follows, we define a set of equivalence classes and use them to define different notions of identifiability. Let $R \subseteq \mathbb{R}^{nm}$ be the set of reward functions for the given MDP model. Let $\sim \subseteq R \times R$ denote an equivalence relation on R. For a given reward $r \in R$, the equivalence class of r with respect to the relation $\sim$ is defined as $[r]_\sim = \{\hat{r} \in R \mid \ \hat{r} \sim r\}$, where we use the shorthand $\hat{r} \sim r$ for $(\hat{r}, r) \in \sim$. Some of the equivalence relations of interest are as follows.

Definition 1 (Distribution Equivalence $\sim_d$) *Given an MDP model, two rewards r and $\hat{r}$ in R are distribution equivalent, denoted by $r \sim_d \hat{r}$, if $p_r = p_{\hat{r}}$.*

In words, two rewards are distribution equivalent when they induce the same optimal trajectory distribution.

Definition 2 (Policy Equivalence $\sim_\pi$) *Given an MDP model, two rewards in R are policy equivalent, denoted by $r \sim_\pi \hat{r}$, if $\pi^*_r|_{\texttt{Access}} = \pi^*_{\hat{r}}|_{\texttt{Access}}$.*

1. The notation π_t, the policy at time step t, is overloaded with π_r, the policy throughout the horizon $[0, T-1]$ corresponding to r.

Two rewards are policy equivalent if they induce the same optimal time-varying policy over the accessible states. Since $p_r = p_{\hat{r}} \iff \pi^*_r|_{\texttt{Access}} = \pi^*_{\hat{r}}|_{\texttt{Access}}$, distribution equivalence class and policy equivalence class are the same, hence we use them interchangeably.

Definition 3 (Weak Trajectory Equivalence $\sim_\tau$ Kim et al. (2021)) *Given an MDP model, two rewards in R are weak trajectory equivalent, denoted by $r \sim_\tau \hat{r}$, if for all $s_0 \in \texttt{support}(\mu_0)$, there exists $c_{s_0} \in \mathbb{R}$ such that $r(\tau) = \hat{r}(\tau) + c_{s_0}$, for all $\tau \in \Omega(s_0)$.*

Weak trajectory equivalence means that the two rewards are equivalent if their discounted sums along trajectories starting from the same initial state are a unique constant apart.

Definition 4 (Strong Trajectory Equivalence $\sim_\omega$) *Given an MDP model, two rewards in R are strong trajectory equivalent, denoted by $r \sim_\omega \hat{r}$, if there exists some $c \in \mathbb{R}$ such that $r(\tau) = \hat{r}(\tau) + c$, for all $\tau \in \Omega$.*

Strong trajectory equivalence is similar to weak trajectory equivalence but requires the discounted sums of rewards along all possible trajectories to be a unique constant apart independent of the initial state.

Definition 5 (State-Action Equivalence $\sim_{s,a}$ Kim et al. (2021)) *Given an MDP model, two rewards in R are state-action equivalent, denoted by $r \sim_{s,a} \hat{r}$, if there exists $c \in \mathbb{R}$ s.t. $r(s,a) = \hat{r}(s,a) + c$, for all $(s,a) \in \mathcal{S} \times \mathcal{A}$.*

State-action equivalence means that the two rewards are equivalent if they are a unique constant c apart at all state-action pairs. When the reward set is $R = \mathbb{R}^{nm}$, state-action equivalence class is the smallest equivalence class up to which it is possible to identify a reward. Indeed, from the definitions, it is easy to see that:

$$r \sim_{s,a} \hat{r} \implies r \sim_\omega \hat{r} \implies r \sim_\tau \hat{r} \implies r \sim_d \hat{r}. \tag{3}$$

Different notions of identifiability of MDP models in the literature deal with the question of when the reverse implications hold. In particular, we have the following definitions.

Definition 6 (Identifiability) *An MDP model is said to be:*

i. weakly identifiable *if for all $r, \hat{r} \in R$, $r \sim_\pi \hat{r} \iff r \sim_\tau \hat{r}$.*

ii. almost-strongly identifiable *if for all $r, \hat{r} \in R$, $r \sim_\pi \hat{r} \iff r \sim_\omega \hat{r}$.*

iii. strongly identifiable *if for all $r, \hat{r} \in R$, $r \sim_\pi \hat{r} \iff r \sim_{s,a} \hat{r}$.*

The definitions of weak and strong identifiability were introduced in Kim et al. (2021). It follows from Equation (3) that strong identifiability implies almost-strong identifiability, which implies weak identifiability.

3. Linear Algebraic Characterizations of Identifiability

In this section, we derive linear algebraic characterizations for the different notions of reward equivalence defined in Section 2.3. The different notions of identifiability are characterized by comparing the corresponding equivalence classes. Throughout this section, we assume that $R = \mathbb{R}^{mn}$.

3.1. Policy-Preserving Equivalence

We first recall that the solutions of finite horizon MaxEntRL problems are usually time-varying policies. However, in general not every time-varying policy is a solution to Problem (2) for some reward. Therefore, we first characterize the conditions a time-varying policy should satisfy to be a solution. Given a policy $\pi = (\pi_t)_{t=0}^{T-1}$, we vectorize it as follows:

$$\pi_t^{\log} = \lambda \begin{bmatrix} \log(\pi_t(a_1|s_1)) & \log(\pi_t(a_1|s_2)) & \cdots & \log(\pi_t(a_m|s_n)) \end{bmatrix}^\intercal \in \mathbb{R}^{mn}, \quad t = 0, 1, \ldots, T-1.$$

Furthermore, we define the matrices $\Gamma \in \mathbb{R}^{Tmn \times (Tn+mn)}$ and $\Xi \in \mathbb{R}^{Tmn}$ as:

$$\Gamma = \begin{bmatrix} \mathbf{I} & -\mathbf{E} & \gamma\mathbf{P} & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mathbf{I} & \mathbf{0} & -\mathbf{E} & \gamma\mathbf{P} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{I} & \vdots & \vdots & \ddots & \mathbf{0} & -\mathbf{E} & \gamma\mathbf{P} \\ \mathbf{I} & \mathbf{0} & \cdots & \cdots & \cdots & \mathbf{0} & -\mathbf{E} \end{bmatrix}, \quad \Xi = \begin{bmatrix} \pi_0^{\log} \\ \pi_1^{\log} \\ \vdots \\ \pi_{T-1}^{\log} \end{bmatrix},$$

with $\mathbf{I} = I_{mn}, \mathbf{E} = \begin{bmatrix} I_n & \cdots & I_n \end{bmatrix}^\intercal \in \mathbb{R}^{nm \times n}$ and $\mathbf{P} = \begin{bmatrix} P_{a^{(1)}}^\intercal & \cdots & P_{a^{(m)}}^\intercal \end{bmatrix}^\intercal \in \mathbb{R}^{nm \times n}$, where $P_{a^{(k)}} \in \mathbb{R}^{n \times n}$ is such that its ij-th entry is given by $\mathcal{T}(s^{(j)}|s^{(i)}, a^{(k)})$, $k \in \{1, \ldots, m\}$. Given Γ, we construct $\Gamma_{\texttt{Access}}$ by only keeping the rows in Γ corresponding to accessible states. Similarly, we construct $\Xi_{\texttt{Access}}$. Details of this construction is given in Appendix A.1. Observe that $\Gamma_{\texttt{Access}}$ and $\Xi_{\texttt{Access}}$ have $\sum_{t=0}^{T-1} m|\texttt{Access}_t|$ rows, which simplifies to Tnm when all states are accessible at all times. Then, we have the following necessary and sufficient condition for a time-varying policy π to be a solution of Problem (2) for some reward.

Proposition 7 *A time-varying policy $\pi = (\pi_t)_{t=0}^{T-1}$ solves Problem (2) for some reward if and only if $\Xi_{\texttt{Access}} \in \texttt{ran}(\Gamma_{\texttt{Access}})$.*

Proof See Appendix A.1. ■

We use this result to first characterize the set of rewards that can induce π then derive the finite-horizon policy-preserving equivalence class. To this end, we define the following affine subspace:

$$\mathcal{X} = \{x \in \mathbb{R}^{mn+Tn} \mid \Gamma_{\texttt{Access}} x = \Xi_{\texttt{Access}}\}. \tag{4}$$

Then, the set of rewards r such that $\pi \in \arg\max_\pi J_{\text{MaxEnt}}(\pi; r)$, denoted by $\mathcal{R}$, is given by:

$$\mathcal{R} = \mathcal{P}\mathcal{X}, \tag{5}$$

where $\mathcal{P} = \begin{bmatrix} \mathbf{I}_{mn} & \mathbf{0}_{mn \times Tn} \end{bmatrix}$ is the projection operator of a $mn + Tn$ dimensional vector onto its first mn components. By defining the following subspace:

$$\mathcal{K}_\Gamma = \mathcal{P}\texttt{ker}(\Gamma_{\texttt{Access}}), \tag{6}$$

we arrive at the following result.

Corollary 8 *Given a time-varying policy $\pi = (\pi_t)_{t=0}^{T-1}$ and an MDP model, let r be a reward that induces π. Then, the policy-preserving equivalence class of r is*

$$[r]_{\sim_\pi} = r \oplus \mathcal{K}_\Gamma.$$

Proof See Appendix A.2. ■

3.2. Weak Trajectory Equivalence and Weak Identifiability

Let $K = |\texttt{support}(\mu_0)|$, which denotes the number of initial states in the MDP. We denote these states by $\{s_0^{(k)}\}_{k=1}^K$. Consider $\{\Omega(s_0^{(k)})\}_{k=1}^K$, where each $\Omega(s_0^{(k)})$ corresponds to the set of all trajectories starting from $s_0^{(k)}$. For each $s_0^{(k)}$, we construct the matrix $M_{s_0^{(k)}} \in \mathbb{R}^{|\Omega(s_0^{(k)})| \times mn}$ as:

$$[M_{s_0^{(k)}}]_{ij} = \sum_{t=0}^{T-1} \gamma^t \mathbb{1}(\tau_i^{(k)}(t) = (s^{(j)}, a^{(j)})), \quad 1 \leq i \leq |\Omega(s_0^{(k)})| \text{ and } 1 \leq j \leq mn, \tag{7}$$

where $\tau_i^{(k)}(t)$ denotes the state action pair at time step t of the i-th trajectory of $\Omega(s_0^{(k)})$, for some arbitrary ordering of trajectories. Using the definition above, we can characterize the weak-trajectory equivalence class of a reward function.

Theorem 9 *The weak-trajectory equivalence class for a reward r is given by:*

$$[r]_{\sim_\tau} = r \oplus \bigcap_{k=1,\ldots,K} (\texttt{ran}(\mathbf{1}_{mn}) \oplus \texttt{ker}(M_{s_0^{(k)}})).$$

Proof See Appendix A.3. ■

The following characterization of weak identifiability follows directly from Theorem 9.

Corollary 10 *An MDP model with $R = \mathbb{R}^{mn}$ is weakly identifiable if and only if*

$$\mathcal{K}_\Gamma \subseteq \bigcap_{i=1,\ldots,K} \left(\texttt{ran}(\mathbf{1}_{mn}) \oplus \texttt{ker}(M_{s_0^{(i)}}) \right).$$

3.3. Strong Trajectory Equivalence and Almost-Strong Identifiability

The strong trajectory equivalence class of a reward can be characterized using a similar derivation to that of Section 3.2. To this end, define the matrix $M = \begin{bmatrix} M_{s_0^{(1)}}^\intercal & M_{s_0^{(2)}}^\intercal & \cdots & M_{s_0^{(K)}}^\intercal \end{bmatrix}^\intercal$.

Theorem 11 *The strong-trajectory equivalence class for a reward r is given by:*

$$[r]_{\sim_\omega} = r \oplus \texttt{ran}(\mathbf{1}_{mn}) \oplus \texttt{ker}(M).$$

Proof See Appendix A.4 ■

Using Theorem 11, we can directly characterize almost-strong identifiability as follows.

Corollary 12 *An MDP model with $R = \mathbb{R}^{mn}$ is almost-strongly identifiable if and only if*

$$\mathcal{K}_\Gamma \subseteq \texttt{ran}(\mathbf{1}_{mn}) \oplus \texttt{ker}(M).$$

The conditions given in Corollaries 11 and 12 can be computationally expensive to verify since the number of trajectories in a stochastic MDP typically grows exponentially with the horizon length. Hence, storing the matrices $(M_{s_0^{(i)}})_{i=1}^K$ and computing their null-space can quickly become computationally infeasible, even for moderately sized MDPs. This means that verifying weak- and almost-strong identifiability can be prohibitive. However, given our linear algebraic characterizations, we can design an incremental algorithm to mitigate the aforementioned problem. The algorithm is based on the following result for almost-strong identifiability.

Proposition 13 *Given an MDP model, let $\{k_1, \ldots, k_r\}$ be a basis for $\mathcal{K}_\Gamma$. Then the MDP model is almost-strongly identifiable if and only if*

$$\forall j \in \{1, \ldots, r\}\ \exists \xi_j \in \mathbb{R}\ \textit{s.t.}\ \forall i \in \{1, \ldots, |\Omega|\}\ M_i k_j = \xi_j,$$

where M_i is the i-th row of M corresponding to the i-th trajectory in Ω.

Proof See Appendix A.5. ■

Proposition 13 says that we can check for almost-strong identifiability by checking a property for individual trajectories instead of storing a large matrix of trajectories and computing its null-space. The procedure is summarized in Algorithm 1 of Appendix B.1. The same algorithm can be directly adapted to test weak identifiability by running it for each starting state $\{s_0^{(k)}\}_{k=1}^K$.

3.4. State-Action Equivalence and Strong Identifiability

For state-action equivalence, the following result follows directly from its definition.

Theorem 14 *The state-action equivalence class for a reward r is given by:*

$$[r]_{\sim_{s,a}} = r \oplus \mathtt{ran}(\mathbf{1}_{mn}).$$

Strong identifiability can be characterized using this theorem as follows.

Corollary 15 *An MDP model with $R = \mathbb{R}^{mn}$ is strongly identifiable if and only if*

$$\mathcal{K}_\Gamma \subseteq \mathtt{ran}(\mathbf{1}_{mn}).$$

Corollary 15 gives an efficient way to check if an MDP model is strongly identifiable. Indeed, we can (i) compute the accessible states $\mathtt{Access}$, (ii) compute the matrix $\Gamma_{\mathtt{Access}}$, (iii) compute a basis of its kernel, and (iv) compute the dimension of $\mathcal{K}_\Gamma$. This dimension is one if and only if the MDP model is strongly identifiable. This consists of a polynomial time algorithm to check the strong identifiability of an MDP model. In fact, the computational complexity can be further improved in the fully accessible case as detailed in Appendix C. We note that this is in contrast to the strong identifiability condition in Cao et al. (2021), which is exponential in the horizon T.

4. Feature-Based Identifiability

So far, we have studied identifiability of rewards in inverse reinforcement learning for the reward set $R = \mathbb{R}^{mn}$. However, a common assumption in reinforcement learning is that the agent is trying to optimize a reward function that can be expressed as a linear combination of known features. This means that the conditions in Corollaries 10, 12 and 15 can be made tighter, since not every reward function in R can be written as a linear combination of the pre-determined features. Given that features describe a subspace in the reward space, incorporating feature-based rewards into our framework becomes just a matter of intersecting these subspaces with our previous results. In particular, consider a feature function $f : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^k$. Define the $mn \times k$ matrix describing the feature function as $F = \begin{bmatrix} f_1(\cdot) & f_2(\cdot) & \cdots & f_k(\cdot) \end{bmatrix}$, where $f_i(\cdot)$ is the i-th feature evaluated at all the state-action pairs. Let $R_f = \{r \in \mathbb{R}^{nm} | \exists\, \omega \in \mathbb{R}^k \text{ s.t. } r(s,a) = \omega^\intercal f(s,a),\ \forall (s,a) \in \mathcal{S} \times \mathcal{A}\}$ be the space of featurized reward functions. We can directly see that $r \in R_f \iff r \in \mathtt{ran}(F)$.

Moreover, to distinguish the equivalence classes when using $R = R_f$ from the ones when $R = \mathbb{R}^{mn}$, we use $[r]_{\sim_{\pi,f}}, [r]_{\sim_{\tau,f}}, [r]_{\sim_{\omega,f}}$ and $[r]_{\sim_{(s,a),f}}$. As in Section 3.1, where it is stated that not every time-varying policy is induced by a reward, clearly not every time-varying policy is induced by a featurized reward.

Theorem 16 *Given a time-varying policy $\pi = (\pi_t)_{t=0}^{T-1}$ and an MDP Model, the set of featurized rewards r such that $\pi \in \arg\max_{\pi} J_{MaxEnt}(\pi; r)$, denoted by $\mathcal{R}_f$, is given by:*

$$\mathcal{R}_f = \mathcal{R} \bigcap \mathtt{ran}(F). \tag{8}$$

Proof Follows from the construction of $\mathcal{R}$ with the added constraint that $r \in \mathtt{ran}(F)$. ■

We note that in Theorem 16, if π is not induced by a featurized reward, then Equation (8) gives the empty set. As in the unconstrained reward case, we can show that the featurized equivalence classes can be derived simply by taking the intersection between the equivalence classes studied in Section 3 with $\mathtt{ran}(F)$:

Theorem 17

$$[r]_{\sim_{\bullet,f}} = [r]_{\sim_{\bullet}} \cap \mathtt{ran}(F), \quad \textit{for } \bullet \in \{\pi, \tau, \omega, (s,a)\}. \tag{9}$$

Proof Similar to the proofs of Section 3, while noting the new structure of R_f. ■

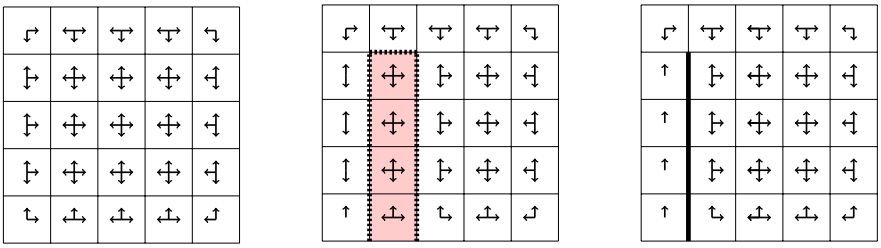
Equation (9) reveals that if $\mathtt{ran}(\mathbf{1}_{mn}) \subseteq \mathtt{ran}(F)$, then $[r]_{\sim_{(s,a),f}} = [r]_{\sim_{s,a}}$. Otherwise, $[r]_{\sim_{(s,a),f}} = \{r\}$. That is, if the vector of ones is not in the range of the feature matrix, it might be possible to exactly identify a unique reward in the featurized setting. Moreover, the results in Theorem 17 are not restricted to rewards constrained to subspaces via features but can easily be generalized to arbitrary reward sets R by taking the intersection with R instead of $\mathtt{ran}(F)$.

5. Numerical Experiments

In this section, we test our framework on different grid world examples with different dynamics. The code to generate the results is available at https://github.com/mlshehab/learning_true_objectives.

5.1. Unconstrained Reward Functions

We demonstrate our framework on three versions of a 5 by 5 grid world shown in Figure 1. The four possible actions available for the agent are: `UP`, `DOWN`, `LEFT`, `RIGHT`. Each action succeeds with a probability 0.9, and with probability 0.1 the agent moves randomly to one of the 4 neighboring cells or stays in the same cell. The first grid world, shown in Figure 1(*a*), is the original grid world where all transitions are admissible. In the second grid world, shown in Figure 1(*b*), we introduce a strip blocking (denoted by the dashed line and red area) that the agent can not enter from outside, but can escape if started inside. Note that all actions are still available at all states, but the outcome of a blocked action is uniformly distributed over the available neighboring cells. Lastly, we introduce a wall in the grid world of Figure 1(*c*) which forces the only possible transition on the left column to be upward. For example, if the agent starts at the lower left corner, then the only way they can reach the right side of the grid world is by first traveling along the left border until the blocking is



(*a*) Original (*b*) Blocking 1 (*c*) Blocking 2

Figure 1: Three grid worlds considered in this section: (a) the original grid world with no blocking, (b) the red strip is blocked from outside, and (c) the thick line only blocks transitions from the left side.

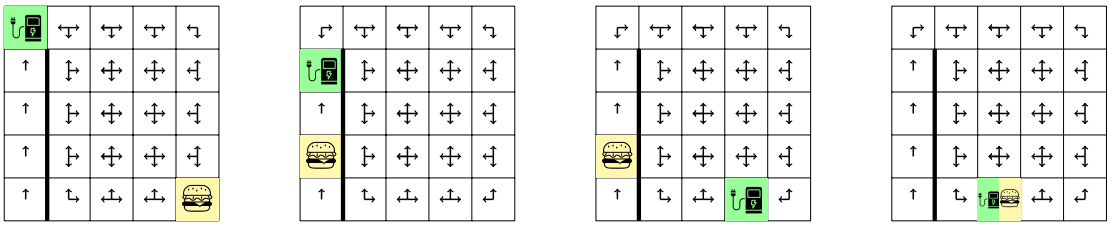
cleared. We take the horizon length to be 15 and the initial distribution to be a single starting state; results with varying horizon lengths and starting states are given in Appendix B.2. For the MDP described by Figure 1(*a*), with any starting state, we find that $\mathcal{K}_\Gamma = \texttt{ran}(\mathbf{1}_{mn})$, which means that the MDP model is strongly identifiable. We note that if we remove self-transitions, the MDP model is not strongly identifiable anymore. On the other hand, we get that $\dim(\mathcal{K}_\Gamma) > 1$ for the MDPs of Figures 1(*b*) and 1(*c*), with a starting state inside the blocking and on the bottom left corner respectively, and hence both are not strongly identifiable. We observe that the subspace $\mathcal{K}_\Gamma$ is along the states in the red strip in Figure 1(*b*) and along the states on the left most wall of Figure 1(*c*), meaning that we can arbitrarily change the reward at these states and still induce the same optimal policy. Additional results with weak and almost-strong identifiability are given in Appendices B.3 and B.4.

5.2. Featurized Reward Functions

In this section, we show how prior information, in the form of featurized rewards, can improve identifiability. Consider a scenario where the rewards depend on landmarks in a grid world and we want to place the landmarks in a way to understand how much agents value different landmarks. In particular, we present four such cases in Figures 2(*a*), 2(*b*), 2(*c*) and 2(*d*), where the important landmarks are a burger joint and a vehicle charging station. We denote the two landmarks by l_1 and l_2. The feature function $f : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^2$ is given by $f_i(s, a) = -\texttt{manhattan\_distance}(s, l_i),\ \forall a \in \mathcal{A}$. F is constructed by stacking the feature function values for all state-action pairs. We report the results with a horizon of 15 and the varying horizon results are given in Appendix B.2. The starting state is the lower left corner. Our framework shows that the any placement of the landmarks, e.g. Figures 2(*a*), 2(*b*), 2(*c*) and 2(*d*), makes the MDP model *strongly identifiable*. In particular, we find that $\mathcal{K}_\Gamma \cap \texttt{ran}(F) = \texttt{ran}(\mathbf{1}_{mn})$ for Figure 2(*a*). For Figures 2(*b*), 2(*c*) and 2(*d*), we find that $\mathcal{K}_\Gamma \cap \texttt{ran}(F) = 0$. Since $\texttt{ran}(\mathbf{1}_{mn}) \not\subseteq \texttt{ran}(F)$ for all these placements, we conclude that the true reward function can be exactly recoverable. Sparse feature results are given in Appendix B.5.

6. Related Works

Here we compare our results with some recent work on the reward ambiguity problem of IRL. In their work, Cao et al. (2021) derive necessary and sufficient conditions for strong-identifiability in infinite and finite horizons. For finite horizon, they characterize strong identifiability in terms of



(*a*) Features 1 (*b*) Features 2 (*c*) Features 3 (*d*) Features 4

Figure 2: The blocked grid world of Figure 1(*c*) with features. The colored cells denote the position of important landmarks.

the properties of "full-action rank" and "full access". Our work builds on Cao et al. (2021) by first deriving explicitly the set of rewards inducing a policy. Additionally, we demonstrate how a linear algebraic characterization enables a polynomial-complexity test for strong identifiability and extends to different notions of identifiability. Rolland et al. (2022) extend the work of Cao et al. (2021) to find linear algebraic characterizations for strong-identifiability in infinite horizon settings. Amin et al. (2017) studied how access to sequential tasks could enhance identifiability and reduce the mismatch between the demonstrator's objective and the learned reward function. However, these previous works assume access either to demonstrations of the agents in multiple sufficiently distinct environments, or multiple tasks. Instead, our work presents unified necessary and sufficient conditions for weak and strong identifiability (with and without features) using the policy in one single environment. Schlaginhaufen and Kamgarpour (2023) also derive a linear algebraic characterization of strong identifiability in the infinite horizon constrained MDP setting. The major commonality between these prior works is assuming an infinite horizon setting, for which the optimal policy is known to be stationary and thus simplifies the analysis. Kim et al. (2021) studied identifiability using the notions of weak and strong identifiability. However, their necessary and sufficient conditions for strong identifiability requires the MDP model to be weakly identifiable, for which a means of verification was not presented except for deterministic MDPs. Our results allow verifying weak identifiability for any MDP. Finally, Skalse et al. (2023) generalize most of the previous works by characterizing transformations on the rewards that preserve optimality under different RL objectives. Our work is complementary to theirs by focusing on MaxEntRL objective and extracting computable linear algebraic characterizations for different equivalence classes.

7. Conclusion

In this work, we established linear algebraic characterizations of weak-, almost-strong, and strong-identifiability of MDPs. Our numerical examples illustrate how these new theoretical results can be leveraged to choose features making the underlying MDP identifiable. In the future, we will build on this approach to design identifiability preserving abstractions. Finally, we will investigate the problem of reward identifiability from a finite set of expert trajectories, instead of knowing the exact expert policy.

Acknowledgments

Toyota Research Institute ("TRI") provided funds to assist the authors with their research but this article solely reflects the opinions and conclusions of its authors and not TRI or any other Toyota entity. MLS and NO were also supported in part by NSF grants CNS-1931982 and CNS-1918123.

References

Pieter Abbeel and Andrew Y Ng. Apprenticeship learning via inverse reinforcement learning. In *International Conference on Machine Learning*, 2004.

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

Saurabh Arora and Prashant Doshi. A survey of inverse reinforcement learning: Challenges, methods and progress. *Artificial Intelligence*, 297, 2021.

Zoe Ashwood, Aditi Jha, and Jonathan W Pillow. Dynamic inverse reinforcement learning for characterizing animal behavior. *Advances in Neural Information Processing Systems*, 35:29663–29676, 2022.

Monica Babes, Vukosi Marivate, Kaushik Subramanian, and Michael L Littman. Apprenticeship learning about multiple intentions. In *International Conference on Machine Learning*, pages 897–904, 2011.

Abdeslam Boularias, Jens Kober, and Jan Peters. Relative entropy inverse reinforcement learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 182–189, 2011.

Haoyang Cao, Samuel Cohen, and Lukasz Szpruch. Identifiability in inverse reinforcement learning. *Advances in Neural Information Processing Systems*, 34:12362–12373, 2021.

Justin Fu, Katie Luo, and Sergey Levine. Learning robust rewards with adverserial inverse reinforcement learning. In *International Conference on Learning Representations*, 2018.

Erik Jenner and Adam Gleave. Preprocessing reward functions for interpretability. *Advances in Neural Information Processing Systems Workshop on Cooperative AI*, 2021.

Kuno Kim, Shivam Garg, Kirankumar Shiragur, and Stefano Ermon. Reward identification in inverse reinforcement learning. In *International Conference on Machine Learning*, pages 5496–5505, 2021.

Dan Li, Mohamad Louai Shehab, Zexiang Liu, Nikos Aréchiga, Jonathan DeCastro, and Necmiye Ozay. Outlier-robust inverse reinforcement learning and reward-based detection of anomalous driving behaviors. In *25th International Conference on Intelligent Transportation Systems (ITSC)*, pages 4175–4182, 2022.

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

Andrew Y Ng and Stuart Russell. Algorithms for inverse reinforcement learning. In *International Conference on Machine Learning*, 2000.

Deepak Ramachandran and Eyal Amir. Bayesian inverse reinforcement learning. In *International Joint Conferences on Artificial Intelligence*, volume 7, pages 2586–2591, 2007.

Giorgia Ramponi, Amarildo Likmeta, Alberto Maria Metelli, Andrea Tirinzoni, and Marcello Restelli. Truly batch model-free inverse reinforcement learning about multiple intentions. In *International conference on artificial intelligence and statistics*, pages 2359–2369. PMLR, 2020.

Nathan D Ratliff, J Andrew Bagnell, and Martin A Zinkevich. Maximum margin planning. In *International Conference on Machine Learning*, pages 729–736, 2006.

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

Andreas Schlaginhaufen and Maryam Kamgarpour. Identifiability and generalizability in constrained inverse reinforcement learning. In *International Conference on Machine Learning*, 2023.

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*, pages 32033–32058, 2023.

Markus Wulfmeier, Peter Ondruska, and Ingmar Posner. Deep inverse reinforcement learning. *CoRR, abs/1507.04888*, 2015.

Brian D. Ziebart, Andrew Maas, J. Andrew Bagnell, and Anind K. Dey. Maximum entropy inverse reinforcement learning. In *AAAI Conference on Artificial Intelligence*, volume 8, pages 1433–1438, 2008.

Appendix A. Proofs

A.1. Proof of Proposition 7

We build on the following result adapted from (Cao et al., 2021) by setting the terminal reward to zero.

Lemma 18 *For any time-varying policy $\pi = (\pi_t)_{t=0}^{T-1}$, and for any function $\nu : \{0, \dots, T-1\} \times \mathcal{S} \to \mathbb{R}$, the reward function given by*

$$r(s,a) = \lambda \log \pi_t(a|s) - \gamma \mathbb{E}_{s' \sim P(.|s,a)}[\nu_{t+1}(s')] + \nu_t(s), \tag{10}$$

with $\nu_T = 0$, is the only reward function for which π is the optimal solution of (2) with value function ν.

Lemma 18 describes *implicitly* all the possible reward functions for which a given policy π is optimal. Since Equation (10) is linear in r and ν for all t in $[0, T-1]$, we can construct a linear system of equations which the reward and value function have to satisfy in order to induce a given policy π. This allows us to *explicitly* describe all the possible rewards inducing π. Proposition 7 is essentially doing this by also taking the ambiguities due to (in)accessibility into account.

Proof [of Proposition 7] We create vectorized versions of the reward and value function as:

$$\begin{aligned} r &= \begin{bmatrix} r(s_1,a_1) & r(s_2,a_1) & \cdots & r(s_n,a_m) \end{bmatrix}^\intercal, \\ \nu_t &= \begin{bmatrix} \nu_t(s_1) & \nu_t(s_2) & \cdots & \nu_t(s_n) \end{bmatrix}^\intercal, \quad t = 0, \cdots, T-1. \end{aligned}$$

Equation (10) gives necessary and sufficient conditions that a reward and value have to satisfy in order to induce a given time-varying policy π. Using the definitions of $\mathbf{I}, \mathbf{E}, \mathbf{P}$ and $\pi_t^{\log}$, we can write the equation as:

$$\begin{bmatrix} \mathbf{I} & -\mathbf{E} & \gamma \mathbf{P} \end{bmatrix} \begin{bmatrix} r \\ \nu_t \\ \nu_{t+1} \end{bmatrix} = \pi_t^{\log}. \tag{11}$$

If a state i is not accessible at time t, we delete all its corresponding rows from Equation (11). The indices of the deleted rows are $\mathcal{I} = \{nl + i \mid l = 0, \cdots, m-1\}$. This amounts to deleting m rows for each inaccessible state, corresponding to m state-action pairs that have no constraint at time step t due to the state not being accessible. Finally, a reward r and value function ν satisfying this equation exist if and only if $\Xi_{\texttt{Access}} \in \texttt{ran}(\Gamma_{\texttt{Access}})$, which concludes the proof. ■

A.2. Proof of Corollary 8

Proof We first prove the $\subseteq$ direction. Let $r, \hat{r}$ be two rewards that induce the same time-varying policy π. By Equation (5), we know that:

$$r = \mathcal{P}x, \qquad \hat{r} = \mathcal{P}\hat{x}, \qquad \text{where } x, \hat{x} \in \mathcal{X}.$$

Since $x, \hat{x} \in \mathcal{X}$, then $x - \hat{x} \in \texttt{ker}(\Gamma_{\texttt{Access}})$. Thus, $r - \hat{r} = \mathcal{P}(x - \hat{x}) \in \mathcal{K}_\Gamma$, hence we have $[r]_{\sim_\pi} \subseteq r \oplus \mathcal{K}_\Gamma$. For the $\supseteq$ direction, let r be a reward inducing π and let $\hat{r} = r + v, v \in \mathcal{K}_\Gamma$. Since

$r \in \mathcal{R}$, then there exists $x \in \mathcal{X}$ such that $\Gamma_{\texttt{Access}} x = \Xi_{\texttt{Access}}$ and $r = \mathcal{P}x$. Define $\hat{x}$ as:

$$\hat{x} = x + \underbrace{\begin{bmatrix} v^\intercal & \mathbf{0}_{Tn}^\intercal \end{bmatrix}^\intercal}_{=\eta}.$$

Then $\Gamma_{\texttt{Access}} \hat{x} = \Gamma_{\texttt{Access}} x + \Gamma_{\texttt{Access}} \eta = \Xi_{\texttt{Access}}$, hence $\hat{x} \in \mathcal{X}$ and $\hat{r} = \mathcal{P}\hat{x}$, so $\hat{r} \in \mathcal{R}$. Thus $\hat{r} \in [r]_{\sim_\pi}$, and thus $r \oplus \mathcal{K}_\Gamma \subseteq [r]_{\sim_\pi}$. ■

A.3. Proof of Theorem 9

We make use of the following lemma for general subspaces S_i and a vector r:

Lemma 19

$$\bigcap_{i=1,\dots,K} r \oplus S_i = r \oplus \bigcap_{i=1,\dots,K} S_i.$$

Proof We proceed by proving inclusion in both directions:
$\subseteq$: Let $v \in \bigcap_{i=1,\dots,K} r \oplus S_i$. Then, $\forall i, v \in r \oplus S_i$. It follows that for every i, there exists s_i such that $v = r + s_i$. Hence, $v - r = s_i$ and then $v - r \in S_i$ for all i. Consequently, $v - r \in \bigcap_{i=1,\dots,K} S_i$ and it follows that $v = r + s$, with $s \in \bigcap_{i=1,\dots,K} S_i$.
$\supseteq$: Let $v \in r \oplus \bigcap_{i=1,\dots,K} S_i$. Then, $v = r + s, s \in \bigcap_{i=1,\dots,K} S_i$. Hence, for all i, $s \in S_i$. Thus, for all i, $v \in r \oplus S_i$ which leads to $v \in \bigcap_{i=1,\dots,K} r \oplus S_i$, concluding the proof. ■

Now, we can prove Theorem 9.
Proof [of Theorem 9] Let r be a reward in R. Using Definition 3 and Equation (7), a reward $\hat{r}$ is weak-trajectory equivalent to r if and only if for all $k = 1, \dots, K$, there exists $c_k \in \mathbb{R}$ such that

$$M_{s_0^{(k)}}(r - \hat{r}) = c_k \mathbf{1}_{|\Omega(s_0^{(k)})|}. \tag{12}$$

Using $M_{s_0^{(k)}} \mathbf{1}_{mn} = (\sum_{t=0}^{T-1} \gamma^t) \mathbf{1}_{|\Omega(s_0^{(k)})|}$ and defining $\tilde{c}_k = c_k / \sum_{t=0}^{T-1} \gamma^t$, Equation (12) can be rewritten $M_{s_0^{(k)}}(r - \hat{r} - \tilde{c}_k \mathbf{1}_{mn}) = 0$. This holds for some $\tilde{c}_k$ if and only if $\hat{r} \in r \oplus \texttt{ran}(1_{mn}) \oplus \texttt{ker}(M_{s_0^{(k)}})$. Since this must hold for all $k = 1, \dots, K$, it gives

$$\hat{r} \in \bigcap_{k=1,\dots,K} (r \oplus \texttt{ran}(\mathbf{1}_{mn}) \oplus \texttt{ker}(M_{s_0^{(k)}})).$$

Using Lemma 19, this can be rewritten as

$$\hat{r} \in r \oplus \bigcap_{k=1,\dots,K} (\texttt{ran}(\mathbf{1}_{mn}) \oplus \texttt{ker}(M_{s_0^{(k)}})),$$

concluding the proof. ■

A.4. Proof of Theorem 11

Proof Let r be a reward. Using Definition 4 and the definition of the matrix M, a reward $\hat{r}$ is strong-trajectory equivalent to r if and only if it exists $c \in \mathbb{R}$ such that

$$M(r - \hat{r}) = c\mathbf{1}_{|\Omega|}. \tag{13}$$

Defining $\tilde{c} = c/\sum_{t=0}^{T-1} \gamma^t$, and using $M(\tilde{c}\mathbf{1}_{mn}) = c\mathbf{1}_{|\Omega|}$, Equation (13) can be rewritten $M(r - \hat{r} - \tilde{c}\mathbf{1}_{mn}) = 0$. Such a $\tilde{c}$ exists if and only if $\hat{r} \in r \oplus \texttt{ran}(\mathbf{1}_{mn}) \oplus \texttt{ker}(M)$. ■

A.5. Proof of Proposition 13

Proof Let $\{k_1, \dots, k_r\}$ be a basis for $\mathcal{K}_\Gamma$. Then:

$$\begin{aligned}
&\mathcal{K}_\Gamma \subseteq \texttt{ran}(\mathbf{1}_{mn}) \oplus \texttt{ker}(M) \quad \text{(from Corollary 12)} \\
&\iff k_j \in \texttt{ran}(\mathbf{1}_{mn}) \oplus \texttt{ker}(M), \quad \forall j = 1, \dots, r \\
&\iff \exists v_j \in \mathbb{R}^{mn},\ \bar{\xi}_j \in \mathbb{R}, \quad \text{s.t. } k_j = \bar{\xi}_j \mathbf{1}_{mn} + v_j, \quad Mv_j = 0, \quad \forall j = 1, \dots, r \\
&\iff Mk_j = \xi_j \mathbf{1}_{|\Omega|}, \quad \xi_j = (\sum_{t=0}^{T-1} \gamma^t)\bar{\xi}_j, \quad \forall j = 1, \dots, r \\
&\iff [Mk_j]_i = \xi_j, \quad \forall i = 1, \cdots, |\Omega|, \quad \forall j = 1, \dots, r
\end{aligned}$$

which concludes the proof. ■

Appendix B. Algorithmic Details and Additional Examples

B.1. Test of Almost-Strong Identifiability

In Algorithm 1, we present an incremental procedure for testing almost-strong identifiability. The same algorithm can be adapted to test weak-identifiability by running it for each starting state $\{s_0^{(k)}\}_{k=1}^K$ and making sure the output is 1 for all starting states. We only have to keep track of the variables $(\xi_j)_{j=1}^r$, and compute the state-visitation row of a trajectory at each time step.

B.2. Results with Varying Horizon Length and Starting States

In this section, we show the effect of horizon length and starting state on strong identifiability results. Changing the starting state and horizon essentially changes `Access`, yielding different identifiability results for different start state/horizon combinations. We generally expect longer horizons and starting states with larger accessible sets to result in better identifiability. In Figures 3(*a*), 3(*b*) and 3(*c*), we show these changes for the examples of Section 5.1. In particular, we plot the dimension of $\mathcal{K}_\Gamma$ with varying horizons for different starting states. Since $\mathbf{1}_{mn} \in \mathcal{K}_\Gamma$, we can equivalently say that an MDP model is strongly identifiable if, and only if, $\texttt{dim}(\mathcal{K}_\Gamma) = 1$. We notice that the MDP model is strongly identifiable for all starting states in Figure 1(*a*) beyond a horizon of 9. For Figures 1(*b*) and 1(*c*), starting states that are most covering (i.e., states 7 and 4) yield the best identifiability results beyond horizons 7 and 13. In Table 1, we show the results for those of Section 5.2.

Algorithm 1: Test of Almost-Strong Identifiability

Input: basis for $\mathcal{K}_\Gamma : \{k_i\}_{i=1}^r$
Output: 1, if MDP model is almost-strongly identifiable, 0 otherwise.

$\tau_1 \leftarrow$ any starting trajectory
$r_1 \leftarrow$ corresponding row of τ_1 in M, constructed using (7)
for $j \leftarrow 1$ **to** r **do**
  $\xi_j \leftarrow r_1^\intercal k_j$
end
for *each trajectory* τ_i **do**
  $r_i \leftarrow$ corresponding row of τ_i in M, constructed using (7)
  for $j \leftarrow 1$ **to** r **do**
    if $r_i^\intercal k_j \neq \xi_j$ **then**
      return 0
    end
  end
end
return 1

Table 1: Identifiability with **dense** features from different initial states and different horizon lengths for the grid world of Figure 1(*c*). strong: *strongly identifiable*, not strong: *not strongly identifiable*, exact: *exactly identifiable*.

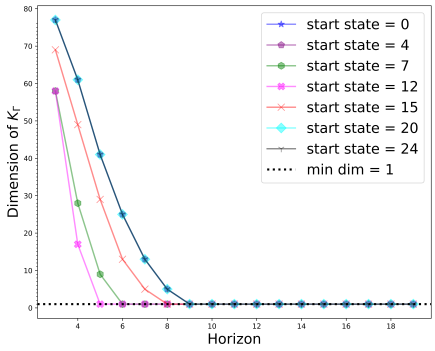
Landmarks Location	**Identifiability Status**			
	Starting State = 4			**Starting State = 15**
	$T \in [1, 5]$	$T \in [6, 7]$	$T \in [8, 20]$	$T \in [1, 20]$
(0, 24)	not strong	strong	strong	strong
(3, 1)	not strong	not strong	exact	exact
(3, 19)	not strong	exact	exact	exact
(14, 14)	not strong	exact	exact	exact

B.3. Example of Weakly Identifiable But Not Strongly Identifiable

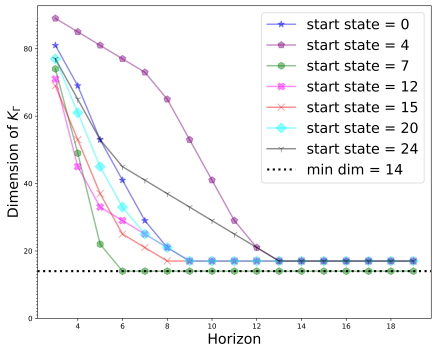
Before giving out an illustrative example of an MDP model that is weakly identifiable but not strongly identifiable, it is worth mentioning the following remark.

Remark 20 *Given Equation (11) and the fact that the reward at the last time step is given by* $r = \bar{\pi}_t^{\log} + \mathbf{E}\nu_{T-1}$*, we can show that* $\mathbf{E}\texttt{ker}(\mathbf{P}) \subseteq \mathcal{K}_\Gamma$*. Thus, if* $\texttt{ker}(\mathbf{P}) \not\subseteq \texttt{ran}(\mathbf{1}_n)$*, the MDP model is not strongly identifiable. Since* $\mathbf{P}\mathbf{1}_n = \mathbf{1}_{mn}$*, the previous condition is equivalent to* $\texttt{ker}(\mathbf{P}) \neq \{0\}$*. Hence, we can equivalently say that an MDP model is strongly identifiable only if* $\mathbf{P}$ *is full rank.*

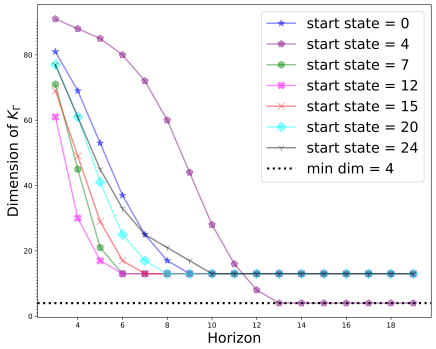
Given our linear algebraic characterizations, it is possible to come up with examples that are weakly identifiable, but not strongly identifiable. For example, consider an MDP with 3 states (s_1, s_2, s_3)



(*a*) Original



(*b*) Blocking 1



(*c*) Blocking 2

Figure 3: The three grid worlds of Figure 1 with varying horizons and varying starting states. The starting state numbering is such that the state on the top left is 0, and increases by 1 going south, and by 5 going east.

and 2 actions a_1, a_2. s_1 is the only starting state. Assume that the transition matrices are given by:

$$P_{a_1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}, P_{a_2} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}.$$

With a horizon of 2, the trajectory matrix M is given by:

$$M = \begin{bmatrix} 1+\gamma & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & \gamma & 0 & 0 \\ 0 & 0 & \gamma & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & \gamma \end{bmatrix}.$$

We can directly see that $\texttt{ker}(M) = \texttt{ran}(e_2, e_5)$, where e_i is the i-th euclidean vector in $\mathbb{R}^6$. By constructing Ψ, we get that $\mathcal{K}_\Gamma = \texttt{ran}(e_1{+}e_3{+}e_4{+}e_6, e_2{+}e_5)$. Thus, $\mathbf{E}\mathcal{K}_\Gamma \subseteq \texttt{ran}(\mathbf{1}_6) \oplus \texttt{ker}(M)$, meaning that the MDP model is weakly identifiable (also follows from Kim et al. (2021) since

deterministic MDPs are weakly identifiable). However, since $\mathbf{P}$ is not full-rank, the MDP model is not strongly identifiable using Remark 20.

B.4. Weak and Almost-Strong Identifiability Results

We run Algorithm 1 on the MDP models in Figures 1(*a*), 1(*b*) and 1(*c*). We take the horizon length to be 15. We get:

- The MDP model of Figure 1(*a*) is *strongly identifiable*, and thus it is trivially both *weakly identifiable* and *almost-strongly identifiable.*
- The MDP model of Figure 1(*b*) is *almost-strongly identifiable* if the set of starting states is completely outside the blocking, or a single state inside the blocking. It is *weakly identifiable* for any set of starting states. We note that the basis of $\mathcal{K}_\Gamma$ is exactly along the states in the red strip (meaning that we can arbitrarily change the reward at these states and still induce the same optimal policy). This means that if the starting state is outside the red strip, all trajectories never visit these blocked states, and thus the inner product in line 9 of Algorithm 1 stays the same (in fact, equals 0). Also, if a trajectory starts inside the red strip, it has to leave in 1 step and can not re-enter, and thus again the value of line 9 stays the same. If we allow transitions inside the strip blocking, then the MDP model becomes *strongly identifiable*.
- Similarly, the MDP model of Figure 1(*c*) is *almost-strongly identifiable* if the set of starting states is right of the wall, or a single state on the left column. It is *weakly identifiable* for any set of starting states. The same reasoning as Figure 1(*b*) applies, since the basis of $\mathcal{K}_\Gamma$ is exactly along the states on the left-most wall.

B.5. Sparse Feature Function Setting

To highlight the importance of the feature function, we consider a more sparse feature function given by $f_i(s,a) = 1, \forall a \in \mathcal{A}$ if $s = l_i$, and 0 otherwise. We consider the MDP model of Figure 1(*c*). We find that with this feature function, if we place any of the burger joint or the charging station on the left most column, the MDP model is not strongly identifiable. However, we are free to place them at any position on the right of the thick wall and obtain an exactly identifiable MDP model with a sufficiently long horizon. This means that if the underlying reward function of agents is a linear combination of these sparse features, then placing any of the burger joint or the charging station on the left-most column is not ideal since we can not disambiguate which landmark the agent prefers. Detailed results with varying horizons are given in Table 2.

Appendix C. Fully Accessible Case

In this section, we derive a closed form for $\mathcal{K}_\Gamma$ in the case where $\texttt{Access}_t = \mathcal{S}$ for $t = 0, \cdots, T-1$. This allows us to directly derive interpretable sufficient conditions that the MDP model has to satisfy in order to be strongly identifiable. We argue that full accessibility is a necessary condition for knowing the policy π everywhere, which is the assumption in Cao et al. (2021). The first result allows us to write $\mathcal{K}_\Gamma$ in a more compact form.

Lemma 21 *Let $\mathcal{K}_\Gamma$ be defined as in Equation (6). Then*

$$\mathcal{K}_\Gamma = \mathbf{E}\mathcal{S},$$

Table 2: Identifiability with **sparse** features from different initial states and different horizons lengths for the grid world of Figure 1(*c*). strong: *strongly identifiable*, not strong: *not strongly identifiable*, exact: *exactly identifiable*.

Landmarks Location	**Identifiability Status**				
	Starting State = 4			**Starting State = 15**	
	$T \in [1, 10]$	$T \in [11, 12]$	$T \in [13, 20]$	$T \in [1, 5]$	$T \in [6, 20]$
(0, 24)	not strong	not strong	exact	not strong	exact
(3, 1)	not strong	not strong	not strong	not strong	not strong
(3, 19)	not strong	not strong	not strong	not strong	not strong
(14, 14)	not strong	exact	exact	not strong	exact

where

$$\mathcal{S} = \{x \in \mathbb{R}^n \mid \mathbf{P}\mathbf{L}^t x \in \mathtt{ran}(\mathbf{E}), \quad t = 0, \cdots, T-1\}, \qquad \mathbf{L} = \frac{1}{m}\sum_{i=1}^{m} P_{a_i}. \tag{14}$$

Proof $\mathbf{E}\mathcal{S} \subseteq \mathcal{K}_\Gamma$: Let $x_{T-1} \in \mathcal{S}$. We want to show that $\mathbf{E}x_{T-1} \in \mathcal{K}_\Gamma$. Since $x_{T-1} \in \mathcal{S}$, then there exists $x_{T-2} \in \mathbb{R}^n$ such that $\mathbf{E}x_{T-2} = \gamma \mathbf{P} x_{T-1}$. Given that $\mathbf{E}^\dagger \mathbf{P} = \mathbf{L}$ (where $\mathbf{E}^\dagger$ denotes the pseudo-inverse of $\mathbf{E}$), we can write $x_{T-2} = \gamma \mathbf{L} x_{T-1}$. This gives that $\mathbf{P}x_{T-2} = \gamma \mathbf{P}\mathbf{L}x_{T-1}$, combined with $x_{T-1} \in \mathcal{S}$, means that there exists $x_{T-3} \in \mathbb{R}^n$ such that $\mathbf{E}x_{T-3} = \gamma \mathbf{P} x_{T-2}$, resulting in $x_{T-3} = \gamma^2 \mathbf{L}^2 x_{T-1}$. Repeating the same process, we can construct $(x_t)_{t=0}^{T-1}$ satisfying:

$$\mathbf{E}x_t = \gamma \mathbf{P} x_{t+1}, \quad t = 0, \cdots, T-2.$$

Now, construct the vector $k = \begin{bmatrix} r^\intercal & \nu_0^\intercal & \nu_1^\intercal & \cdots & \nu_{T-1}^\intercal \end{bmatrix}^\intercal$ where:

$$r = \mathbf{E}x_{T-1} \quad \text{and} \quad \nu_i = \sum_{t=i}^{T-1} x_t, \text{ for } i \in [0, T-1].$$

Then, we can verify that:

$$r - \mathbf{E}\nu_t + \gamma \mathbf{P}\nu_{t+1} = 0 \quad \forall t = [0, T-2], \qquad \text{and } r = \mathbf{E}\nu_{T-1}. \tag{15}$$

Then, $k \in \mathtt{ker}(\Gamma)$ and thus $r = \mathcal{P}k \in \mathcal{K}_\Gamma$. Since $r = \mathbf{E}\nu_{T-1} = \mathbf{E}x_{T-1}$, we conclude that $\mathbf{E}x_{T-1} \in \mathcal{K}_\Gamma$.
$\mathcal{K}_\Gamma \subseteq \mathbf{E}\mathcal{S}$: Let $r \in \mathcal{K}_\Gamma$. We want to prove that $r \in \mathbf{E}\mathcal{S}$, i.e., $r = \mathbf{E}x$ for some $x \in \mathcal{S}$. Since $r \in \mathcal{K}_\Gamma$, then there exists $k \in \mathtt{ker}(\Gamma)$ such that $r = \mathcal{P}k$. The vector k can be written as $\begin{bmatrix} r^\intercal & \nu_0^\intercal & \nu_1^\intercal & \cdots & \nu_{T-1}^\intercal \end{bmatrix}^\intercal$, with r and $(\nu_t)_{t=0}^{T-1}$ satisfying conditions (15). Define $x_{T-1} = \nu_{T-1}$ and $x_t = \nu_t - \sum_{i=t+1}^{T-1} x_i$, $t \in [0, T-2]$. Then $\mathbf{E}x_t = \gamma \mathbf{P} x_{t+1}$ for all $t \in [0, T-2]$, yielding $x_{T-1} \in \mathcal{S}$. Since $r = \mathbf{E}x_{T-1}$, we conclude that $r \in \mathbf{E}\mathcal{S}$. ■

We also make use of the following lemma.

Lemma 22 *Let $x \in \mathbb{R}^n$. Then*

$$\mathbf{P}x \in \mathtt{ran}(\mathbf{E}) \iff x \in \mathtt{ker}(D),$$

where

$$D = \begin{bmatrix} (P_{a_2} - P_{a_1})^\intercal & (P_{a_3} - P_{a_1})^\intercal & \cdots & (P_{a_m} - P_{a_1})^\intercal \end{bmatrix}^\intercal. \tag{16}$$

Proof Let $x \in \mathbb{R}^n$. Then:

$$\begin{aligned}
\mathbf{P}x \in \texttt{ran}(\mathbf{E}) &\iff \exists v \in \mathbb{R}^n \text{ such that } \begin{bmatrix} P_{a_1} \\ \vdots \\ P_{a_m} \end{bmatrix} x = \begin{bmatrix} I_n \\ \vdots \\ I_n \end{bmatrix} v, \\
&\iff \exists v \in \mathbb{R}^n \text{ such that } P_{a_i} x = v \quad i = 1, \cdots, m \\
&\iff (P_{a_1} - P_{a_i})x = 0 \quad i = 2, \cdots, m \\
&\iff x \in \texttt{ker}(D).
\end{aligned}$$

■

Finally, we can write $\mathcal{K}_\Gamma$ compactly as follows.

Proposition 23 *Let* $\mathbf{L}$ *and* D *be defined as in Equations (14) and (16) respectively. Then:*

$$\mathcal{K}_\Gamma = \mathbf{E}ker\left(\begin{bmatrix} D \\ D\mathbf{L} \\ D\mathbf{L}^2 \\ \vdots \\ D\mathbf{L}^{T-1} \end{bmatrix} \right).$$

Proof Follows directly from Lemmas 21 and 22 by noting that $\mathbf{P}\mathbf{L}^t x \in \texttt{ran}(\mathbf{E}) \iff M^t x \in \texttt{ker}(D) \iff x \in \texttt{ker}(DM^t)$. ■

The main implication of Proposition 23 is that checking the necessary and sufficient condition for strong identifiability in MDP models can be done by computing the kernel of a Tmn by n matrix as compared to Γ, which is Tmn by $mn + Tn$. We can directly arrive at the following results:

Corollary 24 *Assume* $\gamma \neq 0$. *If any of the following conditions is true:*

1. *There exists* $t \geq 0$ *such that* $\texttt{rank}(D\mathbf{L}^t) = n - 1$,
2. *There exists two actions* $a_i \in \mathcal{A}, a_j \in \mathcal{A}, i \neq j$ *such that* $\texttt{rank}(\begin{bmatrix} P_{a_i} - P_{a_j} \end{bmatrix}) = n - 1$,

Then the MDP model is strongly identifiable for all horizons $T \geq T^*$ *(where* $T^* = t + 1$ *for the first condition, and* $T^* = 1$ *for the second).*

Proof Follows directly from the closed form of $\mathcal{K}_\Gamma$ given by Proposition 23. ■

Remark 25 *A particular case of Corollary 24 is that a fully accessible MDP model is strongly identifiable for all horizons* $T \geq 1$ *if* $\texttt{rank}(D) = n - 1$. *Interestingly, the same condition on* D *is required in order to identify a reward function up to a constant by observing an expert act in two identical MDP models with only different discount factors* $\gamma_1 \neq \gamma_2$ *[Rolland et al. (2022), Corollary 5]. It is also equivalent to the condition for identification of an action-independent reward from a single expert [Cao et al. (2021), Corollary 3].*