Reward Identification in Inverse Reinforcement Learning

Kuno Kim [1] **Kirankumar Shiragur** [1] **Shivam Garg** [1] **Stefano Ermon** [1]

Abstract

We study the problem of reward identifiability in the context of Inverse Reinforcement Learning (IRL). The reward identifiability question is critical to answer when reasoning about the effectiveness of using Markov Decision Processes (MDPs) as computational models of real world decision makers in order to understand complex decision making behavior and perform counterfactual reasoning. While identifiability has been acknowledged as a fundamental theoretical question in IRL, little is known about the types of MDPs for which rewards are identifiable, or even if there exist such MDPs. In this work, we formalize the reward identification problem in IRL and study how identifiability relates to properties of the MDP model. For deterministic MDP models with the MaxEntRL objective, we prove necessary and sufficient conditions for identifiability. Building on these results, we present efficient algorithms for testing whether or not an MDP model is identifiable.

1. Introduction

Inverse Reinforcement Learning (IRL) is the process of estimating a reward function from demonstrations of optimal behavior. In general, the demonstrator is assumed to behave optimally with respect to an underlying Markov Decision Process (MDP). A key component of the MDP is the reward function which encapsulates the underlying incentives driving the behavior of the optimal demonstrator. A fundamental, yet unsolved question in the field of IRL is reward identifiability: given the optimal behavior, can the reward motivating the behavior be identified up to a reasonable equivalence class?

The reward identifiability question is heavily motivated by real world use cases of IRL. One such area is in applying MDPs to build computational models (Niv, 2009) of real-world, rational decision makers such as investors (Dixit et al., 1994; Rust, 1994), farmers (Nielsen and Kristensen, 2015), doctors (Heckman and Navarro, 2007), and animals (Montague et al., 1995; Doya and Sejnowski, 1998). Given demonstrations of how these decision makers behave, one can apply IRL to extract the underlying reward which can then be studied to better interpret and understand economic, healthcare, and ecological systems. In order for the MDP model to be an effective modeling choice, it should be sufficiently *flexible* so that there exists an MDP that induces complex, realistic behaviors and *identifiable* so that the extracted reward function can be interpreted. If the MDP model is unidentifiable, this fact implies that a large number of arbitrarily different rewards rationalize the demonstrations equally well under the MDP model. As a result, no meaningful understanding of the decision maker can be obtained by IRL.

Another important application of IRL is in counterfactual reasoning (Kalouptsidi et al., 2015; Christensen and Connault, 2019), such as a financial institution attempting to predict the change in behavior of its customer base in response to changes in the economic climate amongst other environmental factors (Rust, 1994). Assuming that the only factor that has changed is the environment and not the customer's incentives, IRL can be used to deduce a set of plausible reward functions describing the customer's incentives and the modeler can choose a reward to re-optimize in a different environment that simulates the change in economic climate. In this scenario, identifiability is a desirable property since often times the modeler's will examine the extracted rewards and choose the one which is most likely to transfer well to the new environment. If the MDP model is unidentifiable, it's unlikely that the modeler can effectively select a transfer reward as a large set of vastly different plausible rewards cannot be interpretted.

Despite the importance of the reward identifiability question, it is heavily under-explored in the Machine Learning (ML) literature. Little known about the types of MDP models for which the reward are identifiable, or even if there exist such MDP models. Many prior works raise concerns that IRL is an ill-posed problem due to identifiability issues (Ng et al., 2000; Ziebart et al., 2011; Ziebart, 2010; Dvijotham and Todorov, 2010; Fu et al., 2017; Geng et al., 2020), often pro-

[*]Equal contribution [1]Department of Computer Science, Stanford University, Palo Alto, USA. Correspondence to: Kuno Kim <khkim@cs.stanford.edu>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

viding the example that a constant reward rationalizes any optimal behavior. This example gives the wrongful impression that all MDP models are always unidentifiable as there are several RL frameworks such as the Maximum Entropy RL (MaxEntRL) framework where constant rewards cannot rationalize all behaviors. To the best of our knowledge, no prior works in the IRL literature have formally studied the reward identifiability problem.

In this work, we formalize the reward identification problem in IRL and study how identifiability relates to properties of the MDP model. For deterministic MDP models with the MaxEntRL objective, we prove necessary and sufficient conditions for identifiability. Building on these results, we present efficient algorithms for testing whether or not an MDP model is identifiable.

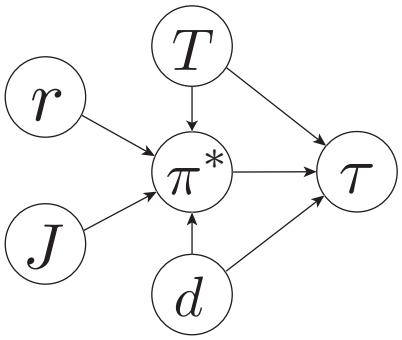

Figure 1. **Graphical Representation of MDP Models** where d is the domain, J is the learning objective, r is the reward, T is the time horizon, π^* is the optimal policy for (d, r, T, J) and τ is a trajectories sampled from π^*

2. Preliminaries

Let $\Delta(S)$ denote the set of probability measures over the set S. An MDP $\mathcal{M}$ is a finite sequence (tuple) $\mathcal{M} := (\mathcal{X}, \mathcal{A}, P, P_0, r, \gamma, T)$ where $\mathcal{X}$ is the discrete state space, $\mathcal{A}$ is the discrete action (decision) space, $P : \mathcal{X} \times \mathcal{A} \to \Delta(\mathcal{X})$ is the transition kernel, $P_0 \in \Delta(\mathcal{X})$ is the initial state distribution, $r : \mathcal{X} \times \mathcal{A} \to \mathbb{R} \in R$ is the reward function in a reward family R, and $T \geq 0$ is the time horizon.

Let $\oplus$ denote the concatenation operator between finite sequences $A_n = (a_0, ..., a_n), B_m = (b_0, ..., b_m)$ so that $A_n \oplus B_m = (a_1, ..., a_n, b_1, ..., b_m)$. For integers $i \leq j$, we use the indexing notation $A_i = (a_i), A_{i:j} = (a_i, a_{i+1}, ..., a_j)$. We decompose an MDP into its domain $d := (\mathcal{X}, \mathcal{A}, P, P_0, \gamma) \in D$, reward $r \in R$, and horizon $T \in \mathbb{N}$ so that $\mathcal{M} := d \oplus (r) \oplus (T)$. Intuitively, the domain d characterizes the physical embodiment of the decision making agent as well as the external environment dynamics, the reward r encapsulates the desired optimal behaviors for a task, and the time horizon T defines how many decisions the agent gets to make to accomplish a task.

A policy is a stochastic function $\pi : \mathcal{X} \to \Delta(\mathcal{A}) \in \Pi$ where Π denotes the set of considered policies. A trajectory of length k is a sequence of state-actions, i.e $\tau = (x_t, a_t)_{t=0}^k$ where for all $0 \leq t \leq k$, $x_t \in \mathcal{X}, a_t \in \mathcal{A}$. The trajectory distribution of a policy π executed in a domain d with time horizon T is denoted $p(\tau; \pi, d, T) = P_0(x_0)\pi(a_0|x_0) \prod_{t=1}^T \pi(a_t|x_t)P(x_t|x_{t-1}, a_{t-1})$. When there is no ambiguity we will omit d, T and simply write $p(\tau; \pi)$. We will denote by $\mathcal{X}^0$ the set of feasible initial states, i.e $x \in \mathcal{X}^0 \Rightarrow P_0(x) > 0$, and by $\Omega[x, d, T]$ the set of feasible trajectories of length T in domain d starting from initial state $x \in \mathcal{X}^0$, so $\tau' \in \Omega[x, d, T]$ if $\tau'_0 = x$ and there exists a policy π that can sample it, i.e $p(\tau'; \pi) > 0$. When x is omitted, $\Omega[d, T] = \bigcup_{x \in \mathcal{X}^0} \Omega[x, d, T]$ denotes the set of all feasible trajectories.

The learning objective $J : \Pi \times R \times D \times \mathbb{N} \to \mathbb{R} \in \mathcal{J}$ is a reward-dependent metric of policy performance that has as a unique global maximum with respect to π for any (d, r, T). An optimal policy π^* for an RL task (d, r, T, J) satisfies $\pi^* = \arg\max_{\pi \in \Pi} J(\pi; d, r, T)$. Note that we restrict ourselves to well-behaving learning objectives that induce a unique optimal policy so that identifiability is well-defined later. For example, the Maximum Entropy RL (MaxEntRL) objective $J_{\text{MaxEnt}}(\pi; d, r, T) = \mathbb{E}_\pi[\sum_{t=0}^T \gamma^t r(x_t, a_t) - \log \pi(a_t|x_t)]$ satisfies the uniqueness maximizer property while the standard RL objective, i.e $\mathbb{E}_\pi[\sum_{t=0}^T \gamma^t r(x_t, a_t)]$, does not. For compactness, we overload notation and write $r(\tau) = \sum_{t=0}^T \gamma^t r(x_t, a_t)$. We can then define *RL task* to be a sequence $\mathcal{M} \oplus (J) := (d, r, T, J)$, i.e an MDP with a corresponding learning objective, which fully defines the RL problem to be solved.

We define the corresponding optimal trajectory distribution for an RL task as $p_r(\tau; d, T, J) := p(\tau; \pi^*, d, T)$. Again, when there is no ambiguity we will omit d, T and simply write $p_r(\tau)$. For some RL tasks, p_r can be written down explicitly as a function of the reward r. For example, when J is the MaxEntRL objective from before and the transitions are deterministic, $p_r(\tau) = e^{r(\tau)} / \int_{\tau' \in \Omega[d,T]} e^{r(\tau')} d\tau'$.

An *MDP model* $\mathcal{P}_{\text{MDP}}[R; d, T, J] := \{p_r(\tau; d, T, J) : r \in R\}$ is a family of optimal trajectory distributions parametrized by the reward r. Note that the unknown parameter is the reward r and (d, T, J) are assumed to be known. The data generating process for an MDP model is solving an RL problem with respect to $\mathcal{M}$ then sampling trajectories τ from the optimal policy π^*. (see Figure 1)

Inverse Reinforcement Learning (IRL) seeks to invert the map $r \to p_r(\tau)$ given samples, or demonstrations, from p_r. IRL is a ill-posed problem when $r \to p_r$ is not injective with respect to r, i.e many underlying rewards rationalize the distribution of observable data. In this work, we are in-

terested in this question of identifiability: when is it possible to invert $r \to p_r(\tau)$ up to a reasonable equivalence class given knowledge of p_r (e.g by having an infinite number of demonstrations) and (d, T, J). In the following section, we will begin by formalizing reward identifiability.

3. Identifiability

In general, a statistical model $\mathcal{P}_\Theta = \{p_\theta \mid \theta \in \Theta\}$ for some parameter family Θ is said to be identifiable if the map $\theta \mapsto p_\theta$ is bijective, i.e $\theta_1 = \theta_2 \iff p_{\theta_1} = p_{\theta_2}$. However, this standard definition of identifiability is not directly applicable to MDP models since most RL objectives will yield the same optimal policy when the rewards are additively shifted by a constant. Thus, we first propose a definition of identifiability adapted to MDP models. We use $\cong$ to denote an equivalence relation on R, and the corresponding equivalence class of r by $[r] = \{\hat{r} \in R \mid \hat{r} \cong r\}$. All proofs of theorems, propositions, and examples will be deferred to the Appendix.

Definition 1. (Identifiability) *An MDP model* $\mathcal{P}_{\text{MDP}}[R; d, T, J] = \{p_r(\tau; d, T, J) \mid r \in R\}$ *is* ***identifiable*** *up to an equivalence relation* $\cong$ *if for all* $r, \hat{r} \in R$*,*

$$r \cong \hat{r} \iff p_r = p_{\hat{r}}$$

We will consider two specific equivalence relations and derive different levels of identifiability from them. The first is trajectory equivalence:

$$\begin{aligned} r \cong_\tau \hat{r} \iff &\forall x \in \mathcal{X}^0, \tau', \tau'' \in \Omega[x, d, T], \\ &\hat{r}(\tau') - r(\tau') = \hat{r}(\tau'') - r(\tau'') \end{aligned} \tag{1}$$

Two rewards are trajectory equivalent, i.e $\cong_\tau$, if they are equal up to a constant after discounted summing over state-action pairs in trajectories starting from the same initial vertex. In other words, the two rewards represent the same preferences over trajectories. The second is state-action equivalence:

$$\begin{aligned} r \cong_{x,a} \hat{r} \iff &\forall (x', a'), (x'', a'') \in \mathcal{X} \times \mathcal{A}, \\ &\hat{r}(x', a') - r(x', a') = \hat{r}(x'', a'') - r(x'', a'') \end{aligned} \tag{2}$$

Two rewards are state-action equivalent, i.e $\cong_{x,a}$, if they are equal up to a constant at the state-action level. In other words, the two rewards represent the same preferences over state-actions. We will use $[r]_\tau = \{\hat{r} \in R \mid \hat{r} \cong_\tau r\}$ and $[r]_{x,a} = \{\hat{r} \in R \mid \hat{r} \cong_{x,a} r\}$ to denote trajectory and state-action equivalence classes, respectively.

Before defining different levels of identifiability, we introduce a notion of *proper* MDP models which we will focus on for the remainder of this work.

Definition 2. (Proper Models) *An MDP model* $\mathcal{P}_{\text{MDP}}[R; d, T, J] = \{p_r(\tau; d, T, J) \mid r \in R\}$ *is* ***proper*** *if for all* $r, \hat{r} \in R$*,*

$$r \cong_\tau \hat{r} \Rightarrow p_r = p_{\hat{r}}$$

A proper MDP model is one that yields the same optimal behavior, i.e $p_r = p_{\hat{r}}$, when the rewards are trajectory equivalent. An MDP model with most RL objectives J will be proper since J generally takes the form $\mathbb{E}_{\tau\sim\pi}[r(\tau)] + f(\pi)$ for some regularization function $f : \Pi \to \mathbb{R}$. For example, MaxEnt MDP models are proper. (see Example 1)

Example 1. *Let* J_{MaxEnt} *be the MaxEntRL objective. Then, the MaxEnt MDP model* $\mathcal{P}_{\text{MDP}}[R; d, T, J_{\text{MaxEnt}}]$ *is proper.*

We are now ready to define different levels of identifiability starting with weak identifiability. (Definition 3)

Definition 3. (Weak Identifiability) *An MDP model* $\mathcal{P}_{\text{MDP}}[R; d, T, J]$ *is* ***weakly identifiable*** *if it is identifiable up to* $\cong_\tau$*, i.e trajectory equivalence.*

If an MDP model is weakly identifiable, then one can identify rewards at the trajectory level given the optimal behavior. Put differently, a modeler can deduce the demonstrator's true preferences over trajectories starting from the same initial state from the demonstrator's behavior. We now define a stronger notion of identifiability in Definition 4.

Definition 4. (Strong Identifiability) *An MDP model is* ***strongly identifiable*** *if it is identifiable up to rewards shifted by a constant, i.e* $\cong_{x,a}$*.*

If an MDP model is strongly identifiable, then one can identify rewards at the most granular level, i.e state-actions, given the optimal behavior. Intuitively, the best we could hope for is to have MDP models be identifiable up to state-action equivalences since the optimal behavior should not change when state-action rewards are shifted by a constant. For example,

Proposition 1 shows that, for proper RL models, strong identfiability is a strictly stronger notion of identifiability than weak identifiability.

Proposition 1. *A proper MDP model is strongly identifiable only if it is weakly identifiable*

In the following sections, we will characterize MDP models that satisfy different levels of identifiability starting with weak identifiability, and eventually moving up to strong identifiability.

4. Weak Identifiability

4.1. Deterministic MaxEnt MDP models

As shown by Proposition 1, an MDP model must first be weakly identifiable in order to be strongly identifiable. In

this section, we first show that under the widely employed MaxEntRL learning objective, J_{MaxEnt}, MDP models with a domain d containing deterministic dynamics is weakly identifiable.

Theorem 1. *Let $\mathcal{P}_{\text{MDP}}[R; d, T, J_{\text{MaxEnt}}]$ be a MaxEnt MDP model and $R \subseteq \{r \mid r : \mathcal{X} \times \mathcal{A} \to \mathbb{R}\}$ be any set of rewards. Then, for all domains $d := (\mathcal{X}, \mathcal{A}, P, P_0, \gamma)$ consisting of deterministic transition dynamics, i.e $\forall(x,a), |\text{supp}(P(\cdot|x,a))| = 1$, a deterministic initial state, i.e $|\text{supp}(P_0)| = 1$, and $T \geq 0$, $\mathcal{P}_{\text{MDP}}[R; d, T, J_{\text{MaxEnt}}]$ is weakly identifiable.*

For such deterministic MaxEnt MDP models, the optimal trajectory distribution is given analytically by (Ziebart et al., 2011)

$$p_r(\tau) = \frac{e^{r(\tau)}}{Z} \text{ where } Z = \sum_{\tau' \in \Omega[d,T]} e^{r(\tau')} \tag{3}$$

Eq. 3 provides the intuition behind Theorem 1. $p_r(\tau)$ is an energy-based model (EBM) with the energy function set to the additive inverse of trajectory rewards $-r(\tau)$. We may than apply the result that two EBMs are equal if and only if the energy functions differ by a constant. (see Appendix for formal proof)

Intuitively, the demonstrator has a strong degree of control over it's future when the dynamics are deterministic, as it can choose to sample a feasible trajectory at its will without interference from the randomness of the dynamics. Thus, the optimal trajectory distribution can be viewed as a noiseless manifestation of the demonstrator's trajectory preferences, which makes it possible to uniquely identify the trajectory level rewards. On the contrary, the following section will provide examples of stochastic MDPs for which the corresponding MDP model is not identifiable as a result of the randomness of the environment interfering with the agent realizing trajectories according to its true preferences.

4.2. Common Misconceptions about Stochastic MDP Models and Weak Identifiability

A common misconception in the IRL literature is that the trajectory distribution of the MaxEnt optimal policy for stochastic dynamics is always equal to

$$p_r(\tau) \propto e^{r(\tau)} P_0(x_0) \prod_{t=1}^{T} P(x_t|x_{t-1}, a_{t-1}) \tag{4}$$

which is also simply an EBM as the dynamics terms can be factored into the exponential. If the misconception were true, than one could also prove that all stochastic MDP models are weakly identifiable with the same proof for Theorem 1. As a counterexample to Eq. 4, consider an MDP with uniform random dynamics, i.e $\forall(x,a,x'), P(x'|x,a) = 1/|\mathcal{X}|$. Consider any reward function r that is state-only, i.e $\forall(x,a,a'), r(x,a) = r(x,a')$, and constant everywhere except for one higher reward state x^*, i.e $\forall x, a \in \mathcal{X} \times \mathcal{A} \setminus \{x^*\} \times \mathcal{A}, r(x,a) = c$ and $\forall a, r(x^*, a) = c + 10$. It's clear that the MaxEnt optimal policy for this MDP is a uniform random policy, since any other policy would have lower entropy yet obtain the same expected rewards due to the environment dynamics forcing uniform state visitation. However, $p_r(\tau)$ from Eq. 4 exponentially prefers trajectories with higher reward and is thus not that attained by a uniform random policy. Thus we can conclude that the distribution in Eq. 4 not always attainable in stochastic environments. We can also see that by setting $\hat{r}$ to be the constant reward, i.e $\forall(x,a), \hat{r}(x,a) = c$, we have two rewards $r, \hat{r}$ that are not trajectory equivalent since $\hat{r}$ has constant trajectory rewards, while r has a higher trajectory reward for trajectories that visit x^* as compared to those that do not. (Note that these two types of trajectories are both feasible since the transition dynamics is always fully supported on the next state) Yet $r, \hat{r}$ have the same uniform trajectory distribution $p_r, p_{\hat{r}}$. Thus, this counterexample also serves to show that $p_r = p_{\hat{r}} \not\Rightarrow r \cong_\tau \hat{r}$ and that not all stochastic MDP models are weakly identifiable. We leave to future work to further characterize weakly identifiability for stochastic MDP models.

5. Strongly Identifiability

5.1. Domain Graphs

The key conceptual idea that will be used throughout the remaining sections is to embed the domain of an MDP model into a graph and reason about how properties of the graph relate to identifiability of the MDP model. We first define domain graphs.

Definition 5. *A **domain graph** for a domain $d = (\mathcal{X}, \mathcal{A}, P, P_0, \gamma)$ is a tuple $G_d := (V_d, E_d, V_d^0)$ where*

1. *$V_d := \mathcal{X} \times \mathcal{A}$ are the vertices*
2. *$V_d^0 := \{(x,a) \mid P^0(x) > 0\}$ are the initial vertices.*
3. *$E_d := \{e := (v, v') = ((x,a),(x',a')) \mid v, v' \in V_d, P(x'|x,a) > 0\}$ are the edges*

In words, the domain graph has a vertex for each state-action pair and a directed edge between vertices if the corresponding transition occurs with positive probability under the domain dynamics. We refer to the source and destination vertex of a directed edge $e = (v, v')$ by $e^s = v$ and $e^d = v'$, respectively. The initial vertex set V_d^0 is the set of vertices that are feasible under the initial state distribution.

A *path* ζ of length $k \geq 0$ in the domain graph is a sequence of vertices $\zeta := (v_t)_{0 \leq t \leq k}$ such that $(v_t, v_{t+1}) \in E_d$

for all $0 \leq t < k$. We denote by $|\zeta|$ the length of the path. Note that there are $k+1$ (not necessarily distinct) vertices on the path. We further introduce indexing notation to extract subpaths. For integers $0 \leq a \leq b \leq k$, $\zeta_{a:} = (v_a, ..., v_k), \zeta_{:a} = (v_0, ..., v_{a-1}), \zeta_{:-a} = (v_0, ..., v_{k-a-1}), \zeta_{-a:} = (v_{k-a}, ..., v_k)$, and $\zeta_{a:b} = (v_a, v_{a+1}, ..., v_{b-1})$. A *simple path* is a path that does not contain the same vertex more than once. In G_d, we say that v' is reachable from v in k-steps if there exists a path of exactly length k that starts at v that ends at v', i.e there exists $\zeta = (v_t)_{0 \leq t \leq k}$ such that $v_0 = v$ and $v_k = v'$. A domain graph G_d is *strongly connected* if there exists a path between any two $v, v' \in V_d$. A *cycle* C is a path that starts and ends at the same vertex, i.e $v_0 = v_k$. We say that a domain graph G_d is *aperiodic* if there does not exist $n > 1$ that divides the length of every cycle in the graph, and *periodic* otherwise.

We denote by $Z[v, d, k]$ the set of all paths of length k that start from $v \in V_d^0$, i.e $Z[v, d, k] = \{\zeta = (v_t)_{0 \leq t \leq k} = (x_t, a_t)_{0 \leq t \leq k} \mid v_0 = v\}$, and subsequently $Z[d, k] = \bigcup_{v \in V_d^0} Z[v, d, k]$. It should be clear that $Z[d, k]$ has a bijection to the set of all feasible trajectories of length k in domain d, i.e $Z[d, k] \leftrightarrow \Omega[d, k]$. Given a reward function r and discount factor γ, we denote the reward of a path ζ to be $r(\zeta) := \sum_{t=0}^{k} \gamma^t r(v_t) = \sum_{t=0}^{k} \gamma^t r(x_t, a_t)$. When combined with path indexing notation, we write $r(\zeta_{a:b}) := \sum_{t=a}^{b-1} \gamma^t r(v_t)$. We now introduce a key concepts related to reachability starting with *layers*.

Definition 6. *The k^{th} layer of a vertex $v \in V_d$, denoted $L_k(v)$, is the set of all vertices reachable in exactly k-steps from v, i.e $L_k(v) = \{v' \in V_d \mid \exists \zeta = (v_t)_{0 \leq t \leq k}$ s.t $v_0 = v, v_k = v'\}$. We define $L_0(v) = \{v\}$ and for $V \subseteq V_d, L_k(V) = \bigcup_{v \in V} L_k(v)$.*

Intuitively, the size of the layers $|L_k(v)|$ should grow with k as vertices further away from the initial vertices in V_d^0 become reachable, although this is not strictly true, e.g bipartite graphs where certain vertices can only be reached in a odd or even number of steps. An important family of domain graphs are those that are *coverable*.

Definition 7. *A vertex $v \in V_d$ is said to be t-**covering** for $t \geq 1$ if $L_t(v) = V_d$. We say that a domain graph G_d is t-**coverable** (or just coverable) if there exists an initial vertex $v \in V_d^0$ that is t-covering.*

In words, if a domain graph is t-coverable, there exists some time t at which all vertices can be reached. Intuitively, a domain with a coverable domain graph is one in which the agent can sample trajectories that terminate at a diverse set of state-actions.

5.2. From Weak to Strong Identifiability

In this section, we show the conditions under which a weakly identifiable MDP model can become strongly identifiable.

Proposition 2. *Let $\mathcal{P}_{\text{MDP}}[R; d, T, J]$ be an MDP model that is weakly identifiable. Then, it is strongly identifiable if and only if for all $r, \hat{r} \in R, (r \cong_\tau \hat{r}) \Rightarrow (r \cong_{x,a} \hat{r})$. In other words, $\forall r \in R, [r]_\tau \subseteq [r]_{x,a}$.*

In words, a weakly identifiable MDP model is strongly identifiable if trajectory equivalence implies state-action equivalence. Along with domain graphs, Proposition 2 will be used in the later sections to prove necessary and sufficient conditions for strong identifiability.

Intuitively, Proposition 2 suggests that in order for an MDP model to be strongly identifiable, the set of feasible trajectories in the MDP should be diverse enough so that the state-action rewards can be deduced from the trajectory rewards. To formalize this intuition, we introduce a linear systems perspective of identifiability. For simplicity of thought, consider an MDP with only one initial state, i.e $|V_d^0| = 1$. A *path (trajectory) matrix* $A[d, k]$ is a matrix, of size $|Z[d, k]| \times |V_d|$, whose rows correspond to frequency counts of each node encountered on the feasible paths, i.e for some enumeration of the vertices V_d and paths $Z[d, k]$, $A_{ij}[d, k]$ is the number of times the j^{th} vertex (state-action) $v^{(j)} = (x, a)^{(j)} \in V_d$ was visited by the i^{th} path $\zeta^{(i)} = \tau^{(i)} \in Z[d, k]$. When the discount factor $\gamma = 1$, the reward of the i^{th} path is simply $r(\zeta^{(i)}) = \sum_{0 \leq j \leq |V_d|} A_{ij}[d, k] r(v^{(j)}) = \sum_{0 \leq j \leq |\mathcal{X} \times \mathcal{A}|} A_{ij}[d, k] r((x, a)^{(j)})$. In vector form, we may denote $\mathbf{r}_{x,a} = (r(v^{(j)}))_{0 \leq j \leq |V_d|}$ and $\mathbf{r}_\tau = (r(\tau^{(j)}))_{0 \leq j \leq |Z[d,k]|}$. Then,

$$A[d, k]\mathbf{r}_{x,a} = \mathbf{r}_\tau \tag{5}$$

We see that the trajectory rewards are a linear transformation of the state-action rewards with the transformation matrix depending on the set of feasible paths (trajectories). Proposition 2 states that a weakly identifiable model is strongly identifiable if the state-action rewards can be identified from trajectory rewards, which, for $\gamma = 1$, is an equivalent statement to saying that, for the linear system of Eq. 5 should have a unique solution (when treating $\mathbf{r}_{x,a}$ as an unknown and $\mathbf{r}_\tau$ as known). Thus, a weakly identifiable model is strongly identifiable if the trajectory (path) matrix is full rank, since this condition guarantees a unique solution to Eq 5.

Corollary 1. *Let $\mathcal{P}_{\text{MDP}}[R; d, T, J]$ be an MDP model that is weakly identifiable, R be the set of all rewards, $|\mathcal{X}^0| = 1$, and $\gamma = 1$. Then, it is strongly identifiable if and only if* $\text{rank}(A[d, T]) = |\mathcal{X} \times \mathcal{A}|$

In order for $A[d, T]$ to be full rank, the there should be a sufficient number of "linearly independent" paths (trajectories) in the feasible path set. Thus, intuitively, domain graphs that generate a sufficiently diverse collection of paths are

likely to yield strongly identifiable MDP models. This intuition will be captured in the strong identification theorem presented in the following section.

5.3. Conditions for Strong Identification

We now state the strong identifiability results, starting with a necessary and sufficient condition for strong identifiability when the domain graph of the MDP model is strongly connected. Indeed, there is a wide pool of real world domains that have strongly connected domain graphs, including physics environments, e.g Mujoco (Todorov et al., 2012), ThreeDWorld (Gan et al., 2020), Navigation environments, e.g Household Robot Navigation (Mo et al., 2018), Mazes (Brockman et al., 2016), and some investment environments (Rust, 1994). We also provide sufficient conditions for strong identifiability for domain graphs with weaker connectivity.

Theorem 2. (Strong Identification Condition) *For all (d, r, T, J) such that the MDP model $\mathcal{P}_{\text{MDP}}[R; d, T, J]$ is proper and G_d is strongly connected,*

- *(Sufficiency) $\mathcal{P}_{\text{MDP}}[R; d, T, J]$ is weakly identifiable, G_d is T_0-coverable, and $T \geq 2T_0 \Rightarrow \mathcal{P}_{\text{MDP}}[R; d, T, J]$ is strongly identifiable*
- *(Necessity) $\mathcal{P}_{\text{MDP}}[R; d, T, J]$ is strongly identifiable $\Rightarrow$ $\mathcal{P}_{\text{MDP}}[R; d, T, J]$ is weakly identifiable, G_d is coverable.*

In short, Theorem 2 states that coverability of the domain graph is a necessary and sufficient condition for strong identifiability. Intuitively, coverable domain graphs have a diverse collection of feasible paths (trajectories) since for every vertex (state-action) there exists a trajectory that terminates at that vertex. Recalling the linear systems perspective on identifiability from Proposition 1, Theorem 2 shows that indeed coverability guarantees that there exists a threshold time horizon $2T_0$ above which there will exist a sufficient number of linearly independent feasible trajectories to have a full rank trajectory matrix $A[d, T]$, and thus uniquely solve for $\mathbf{r}_{x,a}$. On the contrary, for a non-coverable domain graph there does not exist any time horizon at which the MDP model is strongly identifiable.

To better understand and picture the types of strongly connected domain graphs that are coverable, we first need to recognize that there is an equivalence between aperiodicity and coverability for strongly connected graphs.

Proposition 3. *Let G_d be strongly connected. Then, G_d is aperiodic if and only if it is coverable.*

The intuition behind Proposition 3 is that if a graph is aperiodic, then there exist cycles of coprime length which can be traversed an appropriate number of times before heading to a vertex of choice to terminate at. Since all natural numbers above a threshold can be expressed as a positive linear combination of coprime natural numbers (Denardo, 1977), there must exist a time horizon above which the feasible paths can terminate at any vertex. For the converse, if a graph is coverable at T, then it is also coverable at $T+1, T+2, ...$ Therefore, there is a cycle of length T that goes from an initial vertex back to itself as well as a cycle of length $T+1$ that does the same. Since, $T, T+1$ are coprime, the graph is aperiodic. Corollary 2 immediately follows by combining the results of Theorem 2 and Proposition 3.

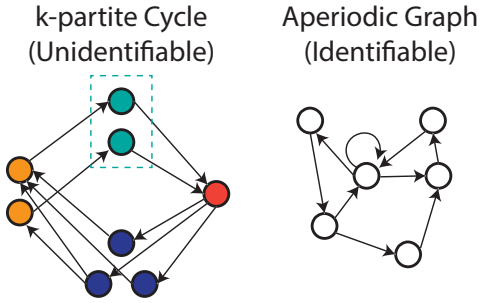

Figure 2. **Examples of domain graphs G_d.** On the left is a domain graph for an MDP model that is not strongly identifiable due to the graph being a k-partite cycle (vertices in the same partition have the same color) which is a periodic graph. Right shows a domain graph for a strongly identifiable MDP model due to the graph being aperiodic as a result of the self loop state.

Corollary 2. (Strong Identification Condition) *For all (d, r, T, J) such that $\mathcal{P}_{\text{MDP}}[R; d, T, J]$ is a proper MDP model and G_d is strongly connected,*

- *(Sufficiency) $\mathcal{P}_{\text{MDP}}[R; d, T, J]$ is weakly identifiable, G_d aperiodic $\Rightarrow \exists T_0 \geq 0$ such that $\forall T \geq T_0, \mathcal{P}_{\text{MDP}}[R; d, T, J]$ is strongly identifiable*
- *(Necessity) $\mathcal{P}_{\text{MDP}}[R; d, T, J]$ is strongly identifiable $\Rightarrow$ $\mathcal{P}_{\text{MDP}}[R; d, T, J]$ is weakly identifiable, G_d is aperiodic.*

Theorem 3 and Corollary 2 shows that strong identifiability, coverability, and aperiodicity are all equivalent properties for strongly connected domain graphs. This result is encouraging since the requirement of aperiodicity is fairly weak; there need only exist two cycles C_1, C_2 of coprime length such as $k, k+1$. In fact, if there is any vertex with a self-loop, i.e $\exists (x, a)$ such that $x \in \text{supp}(P(\cdot|x, a))$, the graph is aperiodic. For domains such as a physics environment, it's reasonable to expect that there exists a static state where you have an action that corresponds to not exerting any external forces into the environment to maintain the static state. Periodic graphs specifically have the topology of a k-partite cycle where vertices in each partition do not have edges between each other, and a vertices in a partition can only be reached in periodic time intervals. See Figure 2 for examples of different types of domain graphs.

5.4. Strong Identifiability Test for Strongly Connected Domain Graphs

Building on the results of Theorem 2 and Corollary 2, we present a simple algorithm which tests for Strongly Identifiability by checking if the underlying domain graph of the MDP model is aperiodic. Algorithm 1 builds off the `Period Finder` algorithm of Denardo (1977) which takes as input a directed graph and returns the greatest common divisor of lengths of every cycle in the graph.

Algorithm 1 Strong Identifiability Test for MDP models with Strongly Connected Domain Graphs

Procedure `MDPIdTest` ($\mathcal{P}_{\text{MDP}}[R; d, T, J]$)
1. Construct a domain graph $G_d = (V_d, E_d, V_d^0)$ from d.
2. Set $gcd =$ `Period Finder`(V_d, E_d) (Denardo, 1977)
3. **return** $gcd == 1$

`MDPIdTest` takes as input an MDP model, constructs a domain graph from d, and checks whether or not the domain graph is aperiodic. Algorithm 1 is correct and inherits the time and space efficiency of `Period Finder` (Denardo, 1977) as shown in Theorem 3.

Theorem 3. *Let $\mathcal{P}_{\text{MDP}}[R; d, T, J]$ be a weakly identifiable MDP model and G_d be strongly connected. Then,*

- *(Correctness)* `MDPIdTest`*($\mathcal{P}_{\text{MDP}}[R; d, T, J]$) returns* 1 *(True) if and only if $\exists T$ such that $\mathcal{P}_{\text{MDP}}[R; d, T, J]$ is strongly identifiable.*
- *(Efficiency)* `MDPIdTest` *runs with time and space complexity $O(|E_d|)$*

In the next section we show how to test for strong identifiability when the domain graph is not strongly connected.

5.5. Strong Identifiability Sufficiency Test for Graphs with Weaker Connectivity

Even when the domain graph is not strongly connected, the sufficient condition from Theorem 2 is still holds as shown in Corollary 3.

Corollary 3. (Strong Identification Condition) *For all (d, r, T, J) such that the MDP model $\mathcal{P}_{\text{MDP}}[R; d, T, J]$ is proper.*

- *(Sufficiency) $\mathcal{P}_{\text{MDP}}[R; d, T, J]$ is weakly identifiable, G_d is T_0-coverable, and $T \geq 2T_0 \Rightarrow \mathcal{P}_{\text{MDP}}[R; d, T, J]$ is strongly identifiable*

While coverability is still a sufficient condition, it is no longer a necessary condition for strong identifiability and Proposition 3 does not hold either. Thus, we propose an algorithm to test coverability directly as a sufficiency test of strong identifiability.

Algorithm 2 Strong Identifiability Sufficiency Test for General MDP models

Procedure `MDPCoverTest` ($\mathcal{P}_{\text{MDP}}[R; d, T, J]$)
4. Construct a transition matrix M from d. $M_{ij} = \tilde{P}(v^{(j)}|v^{(i)})$ where $\tilde{P}(x', a'|x, a) = P(x'|x, a)$.
5. Compute $M^{|V_d|^2}$
6. **if** *The rows for the initial vertices in $M^{|V_d|^2}$ contains only non-zero entries* **then**
   - **return** 1
   - **else**
   - **return** 0
   - **end**

We show in Theorem 4 that `MDPCoverTest` from Algorithm 2 is correct and efficient.

Theorem 4. *Let $\mathcal{P}_{\text{MDP}}[R; d, T, J]$ be a weakly identifiable MDP model. Then,*

- *(Correctness) If* `MDPCoverTest`*($\mathcal{P}_{\text{MDP}}[R; d, T, J]$) returns* 1 *(True) then, $\exists T_0$ such that $\forall T \geq T_0, \mathcal{P}_{\text{MDP}}[R; d, T, J]$ is strongly identifiable.*
- *(Efficiency)* `MDPCoverTest` *runs with time complexity $O(|V_d|^3 \log |V_d|)$ and space complexity $O(|V_d|^2)$*

In Algorithm 2, non-zero entries in the k^{th} power of the transition matrix represent vertices in the k^{th} layer of the domain graph. We check the $|V_d|^2$ power since the covering horizon is upper bounded by $|V_d|^2$ (see proof of Theorem 4 in the Appendix) We leave to future work to derive both necessary and sufficient conditions for strong identifiability for non-strongly connected graphs.

6. Related Work

In the Machine Learning (ML) literature, IRL (Ng et al., 2000; Ramachandran and Amir, 2007; Ziebart et al., 2011; Boularias et al., 2011) has been studied for two main purposes. The first is to run RL with the estimated reward in order to perform Imitation (Ng et al., 2000; Finn et al., 2016) or Apprenticeship Learning (Abbeel and Ng, 2004). IRL is an alternative to direct policy imitation (Pomerleau, 1991; Ho and Ermon, 2016; Kostrikov et al., 2020; Kim et al., 2020a;b) with the motivation being that the reward, in some cases, may be easier to estimate from fewer demonstrations than the policy. The second purpose is to run RL with the estimated reward in a different environment in order to transfer policies (Fu et al., 2017). The intuition is that rewards are likely to transfer more readily across environments compared to policies.

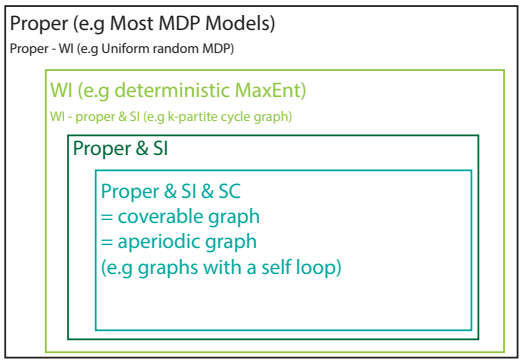

Figure 3. **Venn Diagram of MDP Models** with varying levels of identifiability. Weakly Identifiable (WI) Models, such as deterministic MaxEnt MDP models, are a strict subset of Proper Models and a stochastic MaxEnt MDP model with uniform random dynamics is an example of a proper model that is not WI (section 4.2). Proper and Strongly Identifiable (SI) models are a strict subset of WI models where a deterministic MaxEnt MDP model with a k-partite cycle domain graph (Figure 2) is an example of WI model that is not SI. Finally, proper, SI, and Strongly Connected (SC) models are equivalent to models with a coverable or aperiodic domain graph, such as a graph with a vertex that has a self loop (Figure 2)

Many prior works in IRL (Ng et al., 2000; Ziebart et al., 2011; Ziebart, 2010; Dvijotham and Todorov, 2010; Fu et al., 2017; Amin et al., 2017; Geng et al., 2020) have touched upon the identifiability problem, but none have formally addressed it. Although not widely known in the IRL community, the field of econometrics has a rich body of work on identifying Dynamic Discrete Choice (DDC) (Rust, 1994; Arcidiacono and Miller, 2011; 2020; Abbring and Daljord, 2020) models which is an equivalent problem to IRL. The main result is that for infinite horizon MDP models with stochastic dynamics fully supported on the state space, rewards are not strongly identifiable. DDC literature also separately explores identification for counter factual reasoning (Kalouptsidi et al., 2015; Christensen and Connault, 2019) which has various application areas such as in health care (Heckman and Navarro, 2007) and retail competition (Arcidiacono and Miller, 2011).

A seemingly unrelated area of Network Tomography is also related in the sense that they seek to identify edge level quantities in a graph by path level measurements. Although the specific formulation has subtle differences, such as the paths being constrained to start and end on designated "measurement nodes", several works have looked into necessary and sufficient conditions for identification of edge level quantaties (Ren and Dong, 2016), and several works have proposed algorithms to find the set of linearly independent paths that represent equations which can be solved to identify edge quantities (Gopalan and Ramasubramanian, 2011; Ren and Dong, 2016; Ma et al., 2013).

7. Discussion and future work

In this work we have formalized the reward identification problem in IRL, showed that deterministic MaxEnt MDP models are strongly identifiable if and only if the corresponding domain graph is aperiodic, and presented algorithms for testing identifiability in $O(|E_d|)$ time and space. A summary of our characteriziation of varying levels of identifiability is presented in Figure 3.

The usefulness of IRL in real-world applications, e.g using MDPs as computational models of decision makers for interpretation and counterfactual reasoning, depends heavily on whether the underlying rewards can be identified. Thus the main practical guidance our theory and algorithms provide is a formal framework to evaluate which modeling assumptions and domains are suitable for IRL. Imagine yourself as a neuroscientist studying the behavior of male mice navigating a maze with various stimuli: cheeses, scents of females, and traps. You seek to apply IRL to understand the cognitive processes of the mice by inferring its underlying utility. Important questions to ask are: how should I design the maze so that the mice's utility can be inferred from behavioral data? What type of MDP modeling assumptions should I make? By applying Theorem 1 and 2 we can conclude that a MaxEnt MDP model is appropriate and the maze should be designed so that the mouse has an option to remain static. (having a self-loop state guarantees strong identifiability)

There are a number of important open questions left to answer. First, when are stochastic MDP models weakly identifiable, if ever? The answer to this question can then be combined with our strong identifiability results to characterize when stochastic MDP models are strongly identifiable. Second, what are the necessary and sufficient conditions for strong identifiability when the domain graph is not strongly connected? Do there exist efficient algorithms for testing identifiability in more weakly connected domain graphs?

Acknowledgements

This research was supported by Sony, NSF (#1651565, #1522054, #1733686), ONR (N00014-19-1-2145), AFOSR (FA9550-19-1-0024).

References

Pieter Abbeel and Andrew Y Ng. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the twenty-first international conference on Machine learning*, page 1, 2004.

Jaap H Abbring and Øystein Daljord. Identifying the discount factor in dynamic discrete choice models. *Quantitative Economics*, 11(2):471–501, 2020.

Kareem Amin, Nan Jiang, and Satinder Singh. Repeated inverse reinforcement learning. *arXiv preprint arXiv:1705.05427*, 2017.

Peter Arcidiacono and Robert A Miller. Conditional choice probability estimation of dynamic discrete choice models with unobserved heterogeneity. *Econometrica*, 79(6): 1823–1867, 2011.

Peter Arcidiacono and Robert A Miller. Identifying dynamic discrete choice models off short panels. *Journal of Econometrics*, 215(2):473–485, 2020.

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. JMLR Workshop and Conference Proceedings, 2011.

Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Zaremba Wojciech. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

Timothy Christensen and Benjamin Connault. Counterfactual sensitivity and robustness. *arXiv preprint arXiv:1904.00989*, 2019.

Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, and Clifford Stein. *Introduction to algorithms*. MIT press, 2009.

Eric V Denardo. Periods of connected networks and powers of nonnegative matrices. *Mathematics of Operations Research*, 2(1):20–24, 1977.

Avinash K Dixit, Robert K Dixit, and Robert S Pindyck. *Investment under uncertainty*. Princeton university press, 1994.

Kenji Doya and Terrence J Sejnowski. A computational model of birdsong learning by auditory experience and auditory feedback. In *Central auditory processing and neural modeling*, pages 77–88. Springer, 1998.

Krishnamurthy Dvijotham and Emanuel Todorov. Inverse optimal control with linearly-solvable mdps. In *ICML*, 2010.

Chelsea Finn, Sergey Levine, and Pieter Abbeel. Guided cost learning: Deep inverse optimal control via policy optimization. In *International Conference on Machine Learning*, pages 49–58, June 2016.

Justin Fu, Katie Luo, and Sergey Levine. Learning robust rewards with adversarial inverse reinforcement learning. *arXiv preprint arXiv:1710.11248*, 2017.

Chuang Gan, Jeremy Schwartz, Seth Alter, Martin Schrimpf, James Traer, Julian De Freitas, Jonas Kubilius, Abhishek Bhandwaldar, Nick Haber, Megumi Sano, et al. Threedworld: A platform for interactive multi-modal physical simulation. *arXiv preprint arXiv:2007.04954*, 2020.

Sinong Geng, Houssam Nassif, Carlos Manzanares, Max Reppen, and Ronnie Sircar. Deep pqr: Solving inverse reinforcement learning using anchor actions. In *International Conference on Machine Learning*, pages 3431–3441. PMLR, 2020.

Abishek Gopalan and Srinivasan Ramasubramanian. On identifying additive link metrics using linearly independent cycles and paths. *IEEE/ACM Transactions on Networking*, 20(3):906–916, 2011.

James J Heckman and Salvador Navarro. Dynamic discrete choice and dynamic treatment effects. *Journal of Econometrics*, 136(2):341–396, 2007.

Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. In *Advances in Neural Information Processing Systems*, pages 4565–4573, 2016.

JP Jarvis and Douglas R Shier. Graph-theoretic analysis of finite markov chains. *Applied mathematical modeling: a multidisciplinary approach*, page 85, 1999.

Myrto Kalouptsidi, Paul T Scott, and Eduardo Souza-Rodrigues. Identification of counterfactuals in dynamic discrete choice models. Technical report, National Bureau of Economic Research, 2015.

Kuno Kim, Yihong Gu, Jiaming Song, Shengjia Zhao, and Stefano Ermon. Domain adaptive imitation learning. In *International Conference on Machine Learning*, pages 5286–5295. PMLR, 2020a.

Kuno Kim, Akshat Jindal, Yang Song, Jiaming Song, Yanan Sui, and Stefano Ermon. Imitation with neural density models. *arXiv preprint arXiv:2010.09808*, 2020b.

Ilya Kostrikov, Ofir Nachum, and Jonathan Tompson. Imitation learning via off-policy distribution matching. 2020.

Liang Ma, Ting He, Kin K Leung, Don Towsley, and Ananthram Swami. Efficient identification of additive link metrics via network tomography. In *2013 IEEE 33rd*

International Conference on Distributed Computing Systems, pages 581–590. IEEE, 2013.

Kaichun Mo, Haoxiang Li, Zhe Lin, and Joon-Young Lee. The AdobeIndoorNav Dataset: Towards deep reinforcement learning based real-world indoor robot visual navigation. 2018.

P Read Montague, Peter Dayan, Christophe Person, and Terrence J Sejnowski. Bee foraging in uncertain environments using predictive hebbian learning. *Nature*, 377 (6551):725–728, 1995.

Andrew Y Ng, Stuart J Russell, et al. Algorithms for inverse reinforcement learning. In *Icml*, pages 663–670, 2000.

Lars Relund Nielsen and Anders Ringgaard Kristensen. Markov decision processes to model livestock systems. In *Handbook of operations research in agriculture and the agri-food industry*, pages 419–454. Springer, 2015.

Yael Niv. Reinforcement learning in the brain. *Journal of Mathematical Psychology*, 53(3):139–154, 2009.

Dean A Pomerleau. Efficient training of artificial neural networks for autonomous navigation. *Neural computation*, 3(1):88–97, 1991. ISSN 0899-7667.

Deepak Ramachandran and Eyal Amir. Bayesian inverse reinforcement learning. In *IJCAI*, volume 7, pages 2586–2591, 2007.

Wei Ren and Wei Dong. Robust network tomography: K-identifiability and monitor assignment. In *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*, pages 1–9. IEEE, 2016.

John Rust. Structural estimation of markov decision processes. *Handbook of econometrics*, 4:3081–3143, 1994.

E Todorov, T Erez, and Y Tassa. MuJoCo: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033, October 2012. doi: 10.1109/IROS.2012.6386109.

Brian D Ziebart. Modeling purposeful adaptive behavior with the principle of maximum causal entropy. 2010.

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

Brian D Ziebart, J Andrew Bagnell, and Anind K Dey. Maximum causal entropy correlated equilibria for markov games. In *The 10th International Conference on Autonomous Agents and Multiagent Systems - Volume 1*, AAMAS '11, pages 207–214, Richland, SC, 2011. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9780982657157, 9780982657157.