Adversarial Inverse Optimal Control for General Imitation Learning Losses and Embodiment Transfer

Xiangli Chen **Mathew Monfort** **Brian D. Ziebart**
University of Illinois at Chicago
Chicago, IL 60607
{xchen40,mmonfo2,bziebart}@uic.edu

Peter Carr
Disney Research
Pittsburgh, PA 15213
peter.carr@disneyresearch.com

Abstract

We develop a general framework for inverse optimal control that distinguishes between rationalizing demonstrated behavior and imitating inductively inferred behavior. This enables learning for more general imitative evaluation measures and differences between the capabilities of the demonstrator and those of the learner (i.e., differences in embodiment). Our formulation takes the form of a zero-sum game between a learner attempting to minimize an imitative loss measure, and an adversary attempting to maximize the loss by approximating the demonstrated examples in limited ways. We establish the consistency and generalization guarantees of this approach and illustrate its benefits on real and synthetic imitation learning tasks.

1 INTRODUCTION

Inverse optimal control (IOC) [Kalman, 1964, Rust, 1988, Boyd et al., 1994] and inverse reinforcement learning (IRL) [Ng and Russell, 2000, Abbeel and Ng, 2004] attempt to rationalize demonstrated sequential decision making by estimating a reward/cost function that makes observed decision sequences optimal. When the learned reward is defined over abstract properties of states and actions [Ng and Russell, 2000], it can generalize to new decision processes with states and actions that are similarly described. In contrast, methods that directly estimate a policy mapping from states to controls—also known as “behavioral cloning” [Pomerleau, 1989]—often generalize poorly when attempting to predict goal-directed sequential decisions when aspects of the decision process change.

Unfortunately, the basic IOC problem—selecting a reward function that makes demonstrated decision sequences optimal—is ill-posed, since degenerate solutions exist (e.g., setting all rewards to zero makes every decision sequence optimal) [Ng and Russell, 2000]. When demonstrated behavior is noisy, only degenerate solutions may remain as valid solutions to the basic IOC problem. Existing methods pose the problem in various ways to avoid degenerate solutions. Maximum margin planning (MMP) [Ratliff et al., 2006] seeks a reward function that makes demonstrated sequences have larger reward than all alternatives by a structured loss measure. Maximum (causal) entropy IRL [Ziebart et al., 2010], and its extensions [Boularias et al., 2011, Levine et al., 2011], seek an entropy-maximizing distribution over sequences/policies that matches the feature-based components of the reward function with demonstrated sequences. Each method is constructed around a specific loss function: MMP minimizes the the structured hinge loss, while MaxEnt IRL minimizes the (causal) log loss.

A typical assumption in IOC is that the demonstrator and the learner operate under identical decision processes. In other words, it is assumed that the demonstrator and imitator utilize the same action space, and have the same state transition dynamics. In such settings, imitation can be effectively accomplished by adequately predicting what a demonstrator would do in a new situation. We consider generalized imitation learning problems where the abilities of the demonstrator and the learner are different. This situation arises frequently in practice due to differences in embodiment between human demonstrators and robotic imitators [Nehaniv and Dautenhahn, 2002, Alissandrakis et al., 2002], and, more generally, when autonomously-controlled devices are more expensive and less capable than manually-controlled devices.

We propose a more general framework for inverse optimal control that is both consistent and computationally practical for a range of loss functions and situations where imitation learning across different embodiments is required. The key philosophy of our approach is that unknown properties of *how the demonstrator would behave in new situations* should be treated as pessimistically as possible, since any unwarranted assumptions could lead to substantial errors when behavior is evaluated under more general loss functions or transferred across embodiments. Our formulation produces a zero-sum game between: the learner seeking a control policy to minimize loss; and an adversary seeking a control policy that adequately characterizes the demonstrations, but maximizes the learner's loss. We establish consistency and generalization guarantees, develop algorithms for inference and learning under this formulation, and illustrate the benefits of this approach on synthetic and real imitation learning tasks.

2 BACKGROUND & NOTATION

In this paper, we denote single variables with assigned values in lowercase (e.g., a, s), multivariates with values in bold (e.g., $\mathbf{s}_{1:T}$), and random variables using uppercase (e.g., A_t or $\mathbf{S}_{1:T}$). **Decision processes** are defined by state and action sets ($\mathcal{S}$ and $\mathcal{A}$) and the state transition dynamics τ, which describe the distribution of next states $s_{t+1} \in \mathcal{S}$ given current state $s_t \in \mathcal{S}$ and action $a_t \in \mathcal{A}$: $\tau(s_{t+1}|s_t, a_t)$. We make use of **causally conditioned probability distributions** [Kramer, 1998],

$$P(\mathbf{y}_{1:T}||\mathbf{x}_{1:T}) \triangleq \prod_{t=1}^{T} P(y_t|\mathbf{y}_{1:t-1}, \mathbf{x}_{1:t}),$$

to compactly represent a decision process's **state transition dynamics**,

$$\tau(\mathbf{s}_{1:T}||\mathbf{a}_{1:T-1}) \triangleq \prod_{t=1}^{T} \tau(s_t|\mathbf{s}_{1:t-1}, \mathbf{a}_{1:t-1}),$$

and **stochastic control policy**,

$$\pi(\mathbf{a}_{1:T}||\mathbf{s}_{1:T}) \triangleq \prod_{t=1}^{T} \pi(a_t|\mathbf{a}_{1:t-1}, \mathbf{s}_{1:t}).$$

Multiplied together, these produce a joint probability distribution over the states and actions:

$$P(\mathbf{a}_{1:T}, \mathbf{s}_{1:T}) = \pi(\mathbf{a}_{1:T}||\mathbf{s}_{1:T})\tau(\mathbf{s}_{1:T}||\mathbf{a}_{1:T-1}).$$

We denote **deterministic control policies** (a special case of stochastic control policies) mapping from states or state histories to actions as $\delta(s_t)$ or $\delta(\mathbf{s}_{1:t})$. In addition to denoting the demonstrator's full control policy, π, under dynamics, τ, we also consider distributions of trajectories sampled from the demonstrator's distribution as $\tilde{\pi}$, $\tilde{\tau}$, and a learner's control policy, $\hat{\pi}$, under a (possibly different) set of dynamics $\hat{\tau}$, and estimates of the demonstrator's policy, $\check{\pi}$. We similarly denote states, actions, and deterministic policies corresponds to these different sources as $\tilde{s}$, $\hat{s}$, $\check{s}$, $\tilde{a}$, $\hat{a}$, $\hat{\delta}$, etc.

3 PROBLEM DEFINITION

We begin by formally defining the supervised learning task of imitation learning with general loss measures in Definition 1. The learner's performance is measured by a loss function relating the expected state sequence of the learned control policy with the state sequence resulting from the demonstrator's control policy. The key inductive reasoning challenge is for the learner to produce a good control policy when demonstrations are unavailable by appropriately inferring the demonstrator's behaviors in such situations.

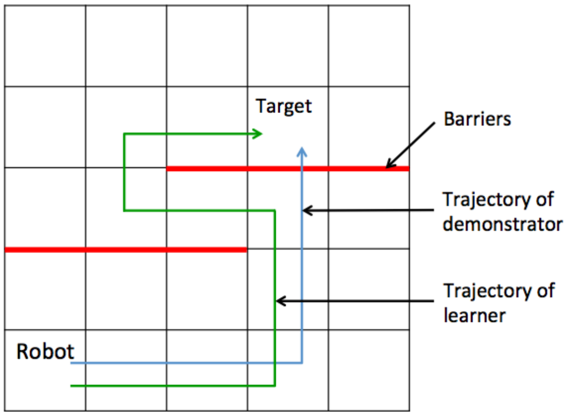

Figure 1: Learning to imitate a slower robot capable of walking over barriers.

Definition 1. *In the task of* **imitation learning with general losses and embodiments**, *at training time: demonstrated traces of behavior are available from distribution $\tilde{P}(\mathbf{A}_{1:T}, \mathbf{S}_{1:T})$ under a dynamical system with known dynamics, $\tau(\mathbf{S}_{1:T}||\mathbf{A}_{1:T})$, and unknown control policy $\pi(\mathbf{A}_{1:T}||\mathbf{S}_{1:T})$. The learner attempts to choose a control policy $\hat{\pi}(\hat{\mathbf{A}}_{1:T}||\hat{\mathbf{S}}_{1:T})$ for potentially different dynamics, $\hat{\tau}(\hat{\mathbf{S}}_{1:T}||\hat{\mathbf{A}}_{1:T})$, that, at testing time, minimizes a given loss function relating (unknown) demonstration policies and learned policies:* $\min_{\hat{\pi}}$ *loss*$_{\tau,\hat{\tau}}(\pi, \hat{\pi})$.

When the demonstrator and the learner operate under different state-action transition dynamics, $\tau \neq \hat{\tau}$, we refer to this setting as the **imitation learning across embodiments** problem. We assume that a loss function expressing the undesirability of the imitator's differences from the demonstrator is available. The key challenge is that the learner must still estimate the control policy of the demonstrator to be able to generalize to new situations, while also constructing its own control policy to overcome its differences in embodiment. We show a simple illustrative example of this in Figure 1.

The ability to minimize the desired imitative loss function when provided enough demonstration data and a sufficiently expressive characterization of decision policies is desired in an imitation learning algorithm. This is formally known as Fisher consistency (Def. 2).

Definition 2. *An imitation learning algorithm producing policy π_{imit} is* **Fisher consistent** *if, given the demonstrator's control policy for any demonstrator/imitator decision* *processes, $(\tau, \hat{\tau})$, and a sufficiently expressive feature representation for policies, the policy $\pi_{\textit{imit}}$ is a loss minimizer:*

$$\pi_{\textit{imit}} \in \operatorname*{argmin}_{\hat{\pi}} \mathbb{E}\left[\textit{loss}_{\tau,\hat{\tau}}(\pi, \hat{\pi})\right]. \tag{1}$$

We focus our attention in this work on loss functions that additively decompose over the state sequence[1]:

$$\mathbb{E}_{P(\mathbf{a}_{1:T},\mathbf{s}_{1:T},\hat{\mathbf{a}}_{1:T},\hat{\mathbf{s}}_{1:T})}\left[\sum_{t=1}^{T} \text{loss}(S_t, \hat{S}_t)\middle|\pi, \tau, \hat{\pi}, \hat{\tau}\right],$$

where the state-action distribution is obtained by combining a stochastic control policy with a state-transition dynamics distribution: $P(\mathbf{a}_{1:T}, \mathbf{s}_{1:T}, \hat{\mathbf{a}}_{1:T}, \hat{\mathbf{s}}_{1:T}) = \tau(\mathbf{s}_{1:T}||\mathbf{a}_{1:T-1})\ \pi(\mathbf{a}_{1:T}||\mathbf{s}_{1:T})\ \hat{\tau}(\hat{\mathbf{s}}_{1:T}||\hat{\mathbf{a}}_{1:T-1})\ \hat{\pi}(\hat{\mathbf{a}}_{1:T}||\hat{\mathbf{s}}_{1:T})$. Important to this problem definition is the independence between the demonstrator and the learner: there is no direct influence of one's actions on the other's state or actions, as shown in the factorization of the joint distribution.

4 ADVERSARIAL APPROACH

We develop an adversarial approach to the problem of imitation learning with general losses and embodiments (Definition 1) by combining the idea of rationalizing demonstrated behaviors from inverse optimal control [Abbeel and Ng, 2004] with a game-theoretic perspective [Topsøe, 1979, Grünwald and Dawid, 2004] that incorporates different imitative losses. Our approach assumes that except for certain properties of the limited samples of available demonstrated behavior, the demonstrator's policy is the worst-case possible for the learner. This avoids generalizing from available demonstrations in a optimistic manner that may be unrealistic and ultimately detrimental to the learner. Using tools from convex optimization (Theorem 3) and constraint generation (Algorithm 1), this formulation can be solved efficiently (Algorithm 2). Though the demonstrator's true policy is unlikely to be maximally detrimental to the learner, considering it as such leads to Fisher consistency (Theorem 1), provides strong generalization guarantees (Theorem 2), and avoids making any unwarranted assumptions.

4.1 ADVERSARIAL FORMULATION AND PROPERTIES

Our approach employs a game-theoretic formulation of the prediction task for additive state-based losses. We introduce an adversarially-estimated policy, $\check{\pi}$, which must be similar to demonstrated training data traces, but is the worst-case for the learner otherwise, as formalized in Definition 3.

Definition 3. *The* **adversarial inverse optimal control** *learner for the joint demonstrator/learner transition dynamics, $(\tau, \hat{\tau})$ is defined as a zero-sum game in which each player chooses a stochastic control policy, $\hat{\pi}$ or $\check{\pi}$, optimizing:*

$$\min_{\hat{\pi}} \max_{\check{\pi}\in\tilde{\Xi}} \mathbb{E}\left[\sum_{t=1}^{T} \textit{loss}(\hat{S}_t, \check{S}_t)\middle|\check{\pi}, \tau, \hat{\pi}, \hat{\tau}\right], \tag{2}$$

where $\tilde{\Xi}$ represents a convex set of constraints measured from characteristics of the demonstrated data (e.g., the moment-matching constraints: $\check{\pi} \in \tilde{\Xi} \iff \mathbb{E}[\sum_{t=1}^{T} \phi(\check{S}_t)|\check{\pi}, \tau] = \tilde{\mathbf{c}} \triangleq \mathbb{E}[\sum_{t=1}^{T} \phi(S_t)|\tilde{\pi}, \tilde{\tau}]$ of inverse reinforcement learning [Abbeel and Ng, 2004]) and the joint state-action distributions are realized by combining control policy and state-transition dynamics: e.g., $P(\hat{\mathbf{a}}_{1:T}, \hat{\mathbf{s}}_{1:T}) = \hat{\pi}(\hat{\mathbf{a}}_{1:T}||\hat{\mathbf{s}}_{1:T})\hat{\tau}(\hat{\mathbf{s}}_{1:T}||\hat{\mathbf{a}}_{1:T-1})$.

Though maximum margin methods, such as MMP [Ratliff et al., 2006] in the imitation learning setting, can similarly incorporate arbitrary structured loss functions, they are not Fisher consistent (Def. 2) even for the relatively simple Hamming loss (i.e., number of state mismatches between two sequences).[2] We establish the consistency of the adversarial inverse optimal control approach in Theorem 1.

Theorem 1. *Given a sufficiently rich feature representation defining the constraint set Ξ, the adversarial inverse optimal control learner is a Fisher consistent loss function minimizer for all additive, state-based losses.*

Proof. A sufficiently rich feature representation is equivalent to the constraint set Ξ containing only the true policy π. Then, under $\check{\pi} = \pi$, Eq. (2) then reduces to:

$$\min_{\hat{\pi}} \mathbb{E}\left[\sum_{t=1}^{T} \text{loss}(\hat{S}_t, \check{S}_t)\middle|\pi, \tau, \hat{\pi}, \hat{\tau}\right], \tag{3}$$

which is the loss function minimizer. □

An additional desirable property of this approach—even when the feature representation is not as expressive—is that if the set $\tilde{\Xi}$ can be defined to include the demonstrator's true policy, π, then generalization performance will be upper bounded by the expected adversarial training loss (Theorem 2).

Theorem 2. *The adversarial transfer IOC formulation (Definition 3) provides a useful generalization bound: if the true demonstrator policy π resides within the constraint set $\tilde{\Xi}$ with probability at least $1 - \alpha$, then the generalization*

[1] Loss functions for state-action pairs can also be incorporated by defining new states that (partially) "remember" previous state-action histories.

[2] This follows directly from the Fisher inconsistency of multiclass classification [Liu, 2007, Tewari and Bartlett, 2007] using the Crammer-Singer multi-class generalization of the hinge loss [Crammer and Singer, 2002], which is a special case of the imitation learning setting with a single time step.

error will be worse than the training error (expected game value) with probability no more than α:

$$P(\pi \in \tilde{\Xi}) \geq 1 - \alpha \implies P\Bigg(\mathbb{E}\left[\sum_{t=1}^{T} \mathit{loss}(\hat{S}_t, S_t)|\pi, \tau, \hat{\pi}, \hat{\tau}\right] \geq \mathbb{E}\left[\sum_{t=1}^{T} \mathit{loss}(\hat{S}_t, \check{S}_t)|\check{\pi}, \tau, \hat{\pi}, \hat{\tau}\right]\Bigg) \leq \alpha. \tag{4}$$

Proof. If $\pi \in \tilde{\Xi}$, then

$$\mathbb{E}\left[\sum_{t=1}^{T} \text{loss}(\hat{S}_t, S_t)|\pi, \tau, \hat{\pi}, \hat{\tau}\right] \leq \mathbb{E}\left[\sum_{t=1}^{T} \text{loss}(\hat{S}_t, \check{S}_t)|\check{\pi}, \tau, \hat{\pi}, \hat{\tau}\right],$$

since replacing π with a worst case policy (maximizer of the set), $\check{\pi}$, only makes the expected loss worse. Thus, bounds on $P(\pi \in \tilde{\Xi})$ provide generalization guarantees with at least as much probability. □

4.2 LEARNING AND INFERENCE ALGORITHMS

Building on recently developed methods for tractably solving adversarial prediction problems for classification with cost-sensitive [Asif et al., 2015] and multivariate [Wang et al., 2015] performance measures, we employ the method of Lagrange multipliers to simplify from a game with one player's actions jointly constrained to a parameterized game with only probabilistic constraints on each player's policy (Theorem 3).

Theorem 3. *An equilibrium for the game of Definition 3 is obtained by solving an unconstrained zero-sum game parameterized by a vector of Lagrange multipliers:*

$$\min_{\mathbf{w}} \min_{\hat{\pi}} \max_{\check{\pi}} \mathbb{E}\left[\sum_{t=1}^{T} \mathit{loss}(\check{S}_t, \hat{S}_t) + \mathbf{w} \cdot \phi(\check{S}_t) \middle| \check{\pi}, \tau, \hat{\pi}, \hat{\tau}\right] - \mathbf{w} \cdot \tilde{\mathbf{c}}.$$

Proof. The proof follows from applying the method of Lagrangian multipliers (a) to the constrained optimization problem of Eq. (2), and then employing strong Lagrangian duality and minimax duality (b):

$$\begin{aligned}
&\min_{\hat{\pi}} \max_{\check{\pi} \in \tilde{\Xi}} \mathbb{E}\left[\sum_{t=1}^{T} \text{loss}(\hat{S}_t, \check{S}_t) \middle| \check{\pi}, \tau, \hat{\pi}, \hat{\tau}\right] \\
\overset{(a)}{=}\ &\min_{\hat{\pi}} \max_{\check{\pi}} \min_{\mathbf{w}} \mathbb{E}\Bigg[\sum_{t=1}^{T} \text{loss}(\hat{S}_t, \check{S}_t) \\
&\qquad + \mathbf{w} \cdot \left(\sum_{t=1}^{T} \phi(\check{S}_t) - \tilde{\mathbf{c}}\right) \Bigg| \check{\pi}, \tau, \hat{\pi}, \hat{\tau}\Bigg] \\
\overset{(b)}{=}\ &\min_{\mathbf{w}} \min_{\hat{\pi}} \max_{\check{\pi}} \mathbb{E}\left[\sum_{t=1}^{T} \text{loss}(\hat{S}_t, \check{S}_t) + \mathbf{w} \cdot \phi(\check{S}_t) \middle| \check{\pi}, \tau, \hat{\pi}, \hat{\tau}\right] \\
&\qquad - \mathbf{w} \cdot \tilde{\mathbf{c}}.
\end{aligned}$$

Note that we assume that the loss function is an expected loss over state predictions. The objective function of our optimization is therefore a bilinear function of the learner's strategy and the adversary's strategy, which provides the strong Lagrangian duality that we employ. No stronger assumption about the state-based loss function is needed so long as it takes this bilinear form. □

We form the stochastic policy of each player $\check{\pi}$, $\hat{\pi}$ as a mixture of deterministic policies: $\check{\delta}$ and $\hat{\delta}$. Conceptually, the payoff matrix of the zero-sum game can be constructed by specifying each combination of deterministic policies, $\check{\delta}$, $\hat{\delta}$, having payoff: $\mathbb{E}[\sum_{t=1}^{T} \text{loss}(\check{S}_t, \hat{S}_t) + \mathbf{w} \cdot \phi(\check{S}_t)|\check{\delta}, \tau, \hat{\delta}, \hat{\tau}]$. An example payoff matrix is shown in Table 1 with the adversary choosing a distribution over columns, and the learner choosing a distribution over rows.

Table 1: The payoff matrix for the adversarial IOC prediction game with $\ell(\check{\delta}, \hat{\delta}) = \mathbb{E}[\sum_{t=1}^{T} \text{loss}(\check{S}_t, \hat{S}_t)|\check{\delta}, \tau, \hat{\delta}, \hat{\tau}]$ and $\psi(\check{\delta}) = \mathbf{w} \cdot \mathbb{E}[\sum_{t=1}^{T} \phi(\check{S}_t)|\check{\delta}, \tau]$.

	$\check{\delta}_1$	$\check{\delta}_2$	$\ldots$	$\check{\delta}_k$
$\hat{\delta}_1$	$\ell(\check{\delta}_1, \hat{\delta}_1)$ $+\psi(\check{\delta}_1)$	$\ell(\check{\delta}_2, \hat{\delta}_1)$ $+\psi(\check{\delta}_2)$	$\ldots$	$\ell(\check{\delta}_k, \hat{\delta}_1)$ $+\psi(\check{\delta}_k)$
$\hat{\delta}_2$	$\ell(\check{\delta}_1, \hat{\delta}_2)$ $+\psi(\check{\delta}_1)$	$\ell(\check{\delta}_2, \hat{\delta}_2)$ $+\psi(\check{\delta}_2)$	$\ldots$	$\ell(\check{\delta}_k, \hat{\delta}_2)$ $+\psi(\check{\delta}_k)$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
$\hat{\delta}_j$	$\ell(\check{\delta}_1, \hat{\delta}_j)$ $+\psi(\check{\delta}_1)$	$\ell(\check{\delta}_2, \hat{\delta}_j)$ $+\psi(\check{\delta}_2)$	$\ldots$	$\ell(\check{\delta}_k, \hat{\delta}_j)$ $+\psi(\check{\delta}_k)$

Unfortunately, this leads to a payoff matrix with a size that is exponential in terms of the actions in the decision process. This cannot be explicitly constructed for practical problems of even modest size. We employ the double oracle method [McMahan et al., 2003] to construct a smaller sub-portion of the matrix that supports the Nash equilibrium strategy for the full game. The basic strategy, outlined in Algorithm 1, iteratively computes a Nash equilibrium for a payoff sub-matrix and augments the payoff matrix with an additional column and row that provide the most improvement for each player.

Finding the best response for each player:

$$\begin{aligned}
&\operatorname*{argmin}_{\hat{\delta}} \mathbb{E}\left[\sum_{t=1}^{T} \text{loss}(\check{S}_t, \hat{S}_t) \middle| \check{\pi}, \tau, \hat{\delta}, \hat{\tau}\right]; \text{ or} \\
&\operatorname*{argmax}_{\check{\delta}} \mathbb{E}\left[\sum_{t=1}^{T} \text{loss}(\check{S}_t, \hat{S}_t) + \mathbf{w} \cdot \phi(\check{S}_t) \middle| \check{\delta}, \tau, \hat{\pi}, \hat{\tau}\right],
\end{aligned} \tag{5}$$

reduces to a time-varying optimal control problem. Consider finding the best demonstrator estimation policy $\check{\delta}^*$. The "expected loss" can be treated as a reward for state $s_t \in \mathcal{S}_D$ with a numerical value of:

$$\text{reward}(s_t) = \mathbb{E}[\text{loss}(s_t, \hat{S}_t)|\hat{\pi}] + \mathbf{w} \cdot \phi(s_t).$$

Algorithm 1 Double oracle method for adversarial IOC

Input: Demonstrator's state transition dynamics τ_D; learner's state transition dynamics, $\hat{\tau}$; loss function: $\text{loss}(s_t, \hat{s}_t)$; initial policy sets: $\check{\Pi}$ and $\hat{\Pi}$; feature function $\phi(s_t)$; and Lagrange multipliers $\mathbf{w}$.

Output: A Nash equilibrium $(\check{\pi}^*, \hat{\pi}^*)$.

```
repeat
    Compute Nash equilibrium $(\check{\pi}^*_D, \hat{\pi}^*)$ and its game value $\check{v}^*$ for sub-game $\check{\Pi}$, $\hat{\Pi}$, $\text{loss}(\cdot,\cdot)$, $\phi(\cdot)$, and $\mathbf{w}$
    Compute best response $\check{\delta}^*$ to $\hat{\pi}^*$ with value $\check{v}_{\check{\delta}^*}$
    if $\check{v}^* \neq v_{\check{\delta}^*}$ then
        Add action to set: $\check{\Pi} \leftarrow \check{\Pi} \cup \check{\delta}^*$
    end if
    Compute Nash equilibrium $(\check{\pi}^*, \hat{\pi}^*)$ and its game with value $\hat{v}^*$ for sub-game $\check{\Pi}$, $\hat{\Pi}$, $\text{loss}(\cdot,\cdot)$, $\phi(\cdot)$, and $\mathbf{w}$
    Compute best response $\hat{\delta}^*$ to $\check{\pi}$ value $\hat{v}_{\hat{\delta}^*}$
    if $\hat{v}^* \neq \hat{v}_{\hat{\delta}^*}$ then
        Add action to set: $\hat{\Pi} \leftarrow \hat{\Pi} \cup \hat{\delta}^*$
    end if
until $\check{v}_{\check{\delta}^*} = \hat{v}_{\hat{\delta}^*} = \check{v}^* = \hat{v}^*$
return $(\check{\pi}^*, \hat{\pi}^*)$
```

Once the reward function is constructed, this time-varying optimal control problem can be solved efficiently in $\mathcal{O}(|\mathcal{S}||\mathcal{A}|T)$ time using value iteration [Bellman, 1957]. We assume that the set of deterministic policies defining each player's stochastic policy is relatively small so that marginalizing to compute state rewards is dominated by the run time of solving the optimal control problem. Each player's best response can be constructed in this manner. Upon termination, neither player's (mixed) strategy can be improved with an additional game action (i.e., deterministic policy), and, thus, by definition $\check{\pi}^*$ and $\hat{\pi}^*$ must be an equilibrium pair [McMahan et al., 2003].

Algorithm 2 Learning algorithm for adversarial IOC

Input: Demonstration $\tilde{P}(\mathbf{A}_{1:T}, \mathbf{S}_{1:T})$ from given decision processes, $(\tau, \hat{\tau})$; loss function: $\text{loss}(\cdot,\cdot)$; and learning rate schedule λ_t

Output: Parameters $\mathbf{w}$ providing adversarial generalization

```
$\mathbf{w} \leftarrow \mathbf{0}$
while $\mathbf{w}$ not converged do
    Compute $\check{\pi}^*$ from parameters $\mathbf{w}$ using double oracle method (Alg. 1) given $\tau, \hat{\tau}$
    Gradient update of parameters: $\mathbf{w} \leftarrow \mathbf{w} - \lambda_t(\mathbb{E}_{P(\check{\mathbf{S}}_{1:T}, \check{\mathbf{A}}_{1:T})}[\sum_{t=1}^{T} \phi(\check{S}_t)|\check{\pi}^*, \tau] - \tilde{\mathbf{c}})$
end while
```

Model parameters $\mathbf{w}$ are estimated using a convex optimization routine described in Algorithm 2. We refer the reader to Asif et al. [Asif et al., 2015] for the proof of convexity for adversarial prediction learning problems of this form with payoff values that are constant with respect to the probability of each player's actions, but not the values themselves.

4.3 EXISTING METHOD RELATIONSHIPS

We conclude our development of the adversarial IOC method by highlighting its conceptual similarities to and differences from previous methods for imitating and predicting sequential decision making policies with the aid of Figure 2.

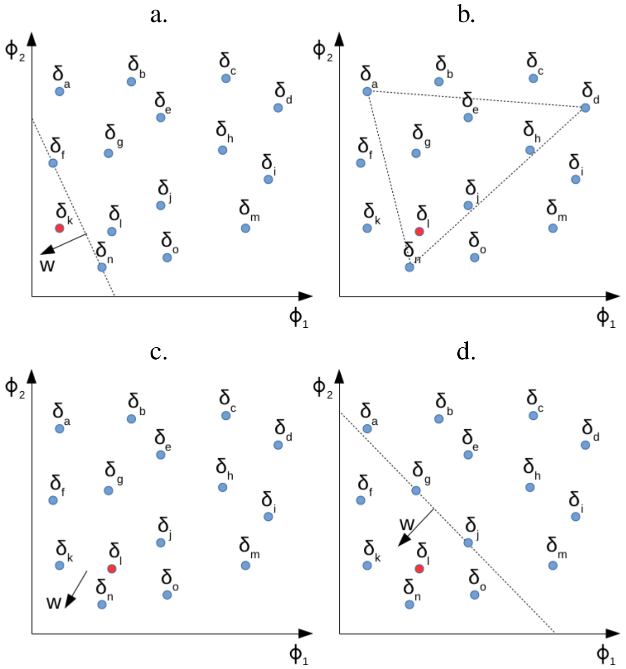

Figure 2: A set of deterministic policies, $\delta_a, \ldots, \delta_o$, represented as points in the two-dimensional feature space based on the expected sum of features under the policy, $\mathbb{E}[\sum_t \phi_k(S_t)|\delta]$. Maximum margin planning [Ratliff et al., 2006] chooses weight $\mathbf{w}$ to separate demonstration δ_k from δ_f and δ_n (top left); Abbeel & Ng's feature-matching algorithm [Abbeel and Ng, 2004] mixes between policies δ_a, δ_d, and δ_n (top right); maximum entropy IRL [Ziebart et al., 2010] chooses a weight direction and produces a probability distribution over all policies (bottom left); and our adversarial approach generates an equilibrium over deterministic policies, $\delta_f, \delta_g, \delta_j, \delta_k, \delta_l, \delta_n$, and δ_o, based on the learned weight $\mathbf{w}$ (bottom right).

When a single demonstrated trajectory resides on the convex hull of the expected feature space (e.g., δ_k in Figure 2a), Abbeel & Ng's feature-matching IRL algorithm [Abbeel and Ng, 2004], maximum margin planning [Ratliff et al., 2006], maximum causal entropy IRL [Ziebart et al., 2010] and our adversarial IOC approach will produce weight estimates $\mathbf{w}$ that make the demonstrated policy (uniquely) optimal. They differ in that Abbeel & Ng's algorithm will be satisfied with any weight $\mathbf{w}$ that makes δ_k uniquely optimal, since this would match feature counts with the distribution of demonstrated sequences: $\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[\sum_{t=1}^{T}\phi(S_t^{(i)})|\pi,\tau] = \frac{1}{n}\sum_{i=1}^{n}\sum_{t=1}^{T}\phi(s_t^{(i)})$, guaranteeing equivalent expected reward under the assumption that the reward function is linear in the feature vector, ϕ [Abbeel and Ng, 2004]. As a refinement to this idea, maximum margin planning [Ratliff et al., 2006] seeks parameter weights that make a demonstrated policy on the convex hull "more optimal" than other policies by a structured margin (using a structured loss to penalize being "almost as optimal" from a policy that is very different from the demonstration policy), as shown in Figure 2a. Maximum (causal) entropy inverse reinforcement learning [Ziebart et al., 2010], which employs a Boltzmann distribution over actions for each state, similarly converges to allocate all of its probability to the actions of the demonstrated policy. Adversarial IOC's behavior is equivalent to that of maximum margin planning (when a small amount of regularization is included) in this situation: it obtains a weight vector $\mathbf{w}$ so that the demonstrated policy is better than all alternatives by the structured loss.

When demonstrated trajectories are on the interior of the convex hull, as shown in Figure 2b-d, the behaviors of the methods differ substantially. Abbeel & Ng's feature-matching algorithm [Abbeel and Ng, 2004] produces a mixture of deterministic policies (e.g., a mixture of δ_a, δ_d, and δ_n with probabilities of 10%, 10%, and 80%, as shown in Figure 2b) that match demonstrated feature counts. Unfortunately, many such mixtures exist and switching between the extremes of the convex hull often proves to imitate poorly in practice. Maximum (causal) entropy inverse reinforcement learning [Ziebart et al., 2010] provides a distribution that places some probability on each deterministic policy, with higher probabilities specified by the learned weight vector $\mathbf{w}$, as shown in Figure 2c. This avoids mixing between extremely different deterministic policies [Abbeel and Ng, 2004], but requires a computationally expensive integration over all policies instead of using an optimal MDP policy solver as a sub-routine for learning.

An additional limitation of maximum (causal) entropy inverse reinforcement learning [Ziebart et al., 2010] is due to its global normalization over control policies. This normalization imposes burdensome implicit constraints on learned cost functions[3] due to cycle sensitivity [Monfort et al., 2015, Ziebart, 2010], as defined below and illustrated in Figure 3. These cost function constraints can increase the loss of resulting maximum entropy IRL predictions in practice even when demonstrated behavior trajectories do not include the states of the cycles. Conversely, removing a completely irrelevant cycle from a Markov decision process can drastically change the estimated reward/cost function.

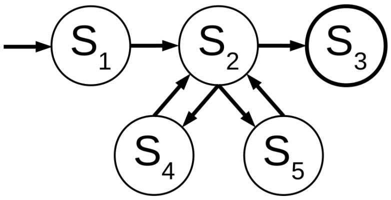

Figure 3: A deterministic Markov decision process with initial state s_1 and absorbing state s_3 in which we assume for simplicity that state two and four have identical features: $\phi(s_4) = \phi(s_5)$. Under maximum entropy inverse reinforcement learning [Ziebart et al., 2010], $P(s_{1:T}) \propto e^{-\mathbf{w}\cdot\sum_{t=1}^{T}\phi(s_t)}$ The number of paths terminating in the absorbing state of odd length $n \geq 3$ is $2^{\frac{n-3}{2}}$, each with cost of $C_0 + \frac{n-3}{2}C_1$, where $C_0 \triangleq \mathbf{w}\cdot(\phi(s_1)+\phi(s_2)+\phi(s_3))$ and $C_1 \triangleq \mathbf{w}\cdot(\phi(s_2)+\phi(s_4))$. The normalization constant under maximum entropy inverse optimal control is $\sum_{i=0}^{\infty}2^i e^{-C_0+iC_1} = e^{-C_0}\sum_{i=0}^{\infty}e^{i\ln 2 - iC_1}$, and requires that $C_1 > \ln 2$ for it to be finite.

Definition 4. *An inverse optimal control method is characterized as being* **cycle sensitive** *when differences in a decision process's state representation and dynamics—independent from demonstrated trajectories through the decision process—can introduce arbitrary additional constraints on the estimated cost function.*

When provided with sub-optimal demonstration policies, our adversarial approach mixes together deterministic policies to match feature expectations with demonstrated policies. Unlike the extreme convex hull policies of the feature-matching algorithm [Abbeel and Ng, 2004], the deterministic policies mixed together by the adversarial IOC method are "competitive" with the demonstrated policy. They are specified by the learned weight vector $\mathbf{w}$, which determines thresholds for which deterministic policies need to be considered for mixing. For example, deterministic policies $\delta_k, \delta_n, \delta_o, \delta_l, \delta_j, \delta_g$ are included in the strategic game and appropriately mixed together when δ_l is demonstrated, as shown in Figure 2d. From this perspective, adversarial IOC can be viewed as combining the mixing behavior of Abbeel & Ng's feature-matching algorithm [Abbeel and Ng, 2004] with MMP's margin-like [Ratliff et al., 2006] selection of policies to mix, while avoiding the integration over all policies required by maximum (causal) entropy inverse reinforcement learning [Ziebart et al., 2010] and its sensitivity to irrelevant cycles in the MDP.

[3] These implicit cost function constraints are in contrast to explicit constraint, like cost function non-negativity to prevent negative cost cycles.

5 EXPERIMENTS

We demonstrate the benefits of our approach on synthetic and real imitation learning tasks with application-specific imitation losses and/or different embodiments.

5.1 NAVIGATION ACROSS A GRID

Our first experiment considers trajectories collected from simulated navigation across a discrete grid with various characteristics. For each task, a robot navigates through the environment to reach a target location. Each cell of the grid world is denoted by its horizontal and vertical positions, (x, y), where each is an integer value from 1 to N. The robot's goal is to reach the target location while minimizing the navigation cost within a fixed period of time. We define this fixed time horizon as the maximum number of steps needed to reach any cell of the grid world. The navigation task stops once the robot reaches the target, which is equivalent to representing that the robot stays in the cell where the target exists until the end of the final time step. We formulate the robot navigation problem to be an optimal sequential decision-making problem in a finite Markov decision process (MDP) in which the policy minimizes the expected cost of successful navigation.

Differing initial positions for the robot and the target location are sampled uniformly from the $N \times N$ cells. We generate the cost $C(s)$ for the demonstrator to traverse a particular grid cell (x, y position in the grid) in our simulations based on a linear function of feature vectors, $\phi(s)$, which characterize the state: $C(s) = \theta^T \phi(s) + \varepsilon(s)$, and a noise component, $\varepsilon(s)$. We employ a 7-element feature function vector, $\phi(s)$, in these grid experiments and choose each element of θ by sampling from the uniform distribution $U(0, 1)$. The noise component is similarly sampled from a uniform distribution, $U(0, \varepsilon)$, bounded by a scalar parameter ε that controls the amount of noise in the imitation learning task. We set $C(s) = 0$ when the robot reaches the cell where the target exists. Note that the cost is stationary; all values of $C(s)$ are sampled and fixed for each navigation task. The robot can attempt to move one step from its position in each of the cardinal directions (north, south, east and west), except it is unable to move beyond the boundaries of the grid. When the state transition dynamics are stochastic, the robot may accidentally move into another neighboring cell rather the intended one (e.g., north or south when attempting to move east). The state transition dynamics are formally then:

$$p(s_{t+1}|s_t, a_t) = \begin{cases} p_m & \text{matching the action} \\ \frac{1-p_m}{\text{number of neighbor cells}} & \text{neighbor cells} \end{cases}$$

where we call p_m the matching probability. The optimal policy from solving the finite MDP problem gives the robot's navigation strategy which then can generate a navigation trajectory for learning.

We establish a specific set of grid world navigation simulation characteristics as the base setting of our simulations:

- The size of the grid world is 9×9;
- The noise weight ε is 1; and
- The matching probability p_m is 0.7.

We repeat the simulation 200 times, yielding 200 navigation trajectories of which we use 100 as training data, and the remainder as testing data. We compare adversarial IOC to MMP [Ratliff et al., 2006] across various settings of the size of the grid, the amount of feature noise, the matching probability, and the number of training/testing datapoints. For our grid navigation experiments, we evaluate the loss as the Euclidean distance between the demonstrator's grid position (x, y) and the imitator's grid position $(\hat{x}, \hat{y})$, normalized by the maximum loss, m:

$$\frac{1}{N_{\text{test}}} \sum_{n=1}^{N_{\text{test}}} \mathbb{E}\left[\sum_{t=1}^{T} m^{-1} \sqrt{\left(X_t^{(n)} - \hat{X}_t^{(n)}\right)^2 + \left(Y_t^{(n)} - \hat{Y}_t^{(n)}\right)^2}\right],$$

where ($X_t^{(n)}$ and $Y_t^{(n)}$) are random variables under the demonstrator's control policy—the policy from solving the simulated finite MDP problem—and ($\hat{X}_t^{(n)}$ and $\hat{Y}_t^{(n)}$) are the ones with estimated policy. We employ this normalized Euclidean loss as the structured loss function for the margin in MMP and the game payoff in our adversarial method.

As shown in the first four plots of Figure 4, our adversarial IOC approach (Adv) provides significant improvements in reducing the imitation loss over the trajectory compared to maximum margin planning (MMP) under equivalent embodiment setting (i.e., standard imitation learning). Though the imitator's performance generally becomes worse as the imitation task becomes more difficult (less determinism in the state transition dynamics, increased amounts of noise influencing the demonstrator's optimal policy, and larger sizes of the grid), adversarial IOC consistently outperforms MMP across all of these settings. Very little dependence of the imitation performance on the number of training examples in the fourth plot reveals the general efficiency of training using IOC/IRL methods that estimate the motivating cost function.

We also compare the performance of our adversarial IOC imitation policy with the policy produced by MMP [Ratliff et al., 2006] when demonstrator and imitator have different embodiments. We assume that the demonstration robot's dynamics are noise free and more flexible. In our first experiment, the demonstrator has deterministic state transition dynamics with matching probability 1, and we evaluate the performance of the learner operating under stochastic dynamics with various matching probabilities from 0.9 to 0.5. In the second experiment, we set some obstacles in the grid world so that the imitating robot has to make a detour when it faces any of them, but the demonstrator does not. We evaluate the performance of learner on various number of obstacles from 20 to 60.

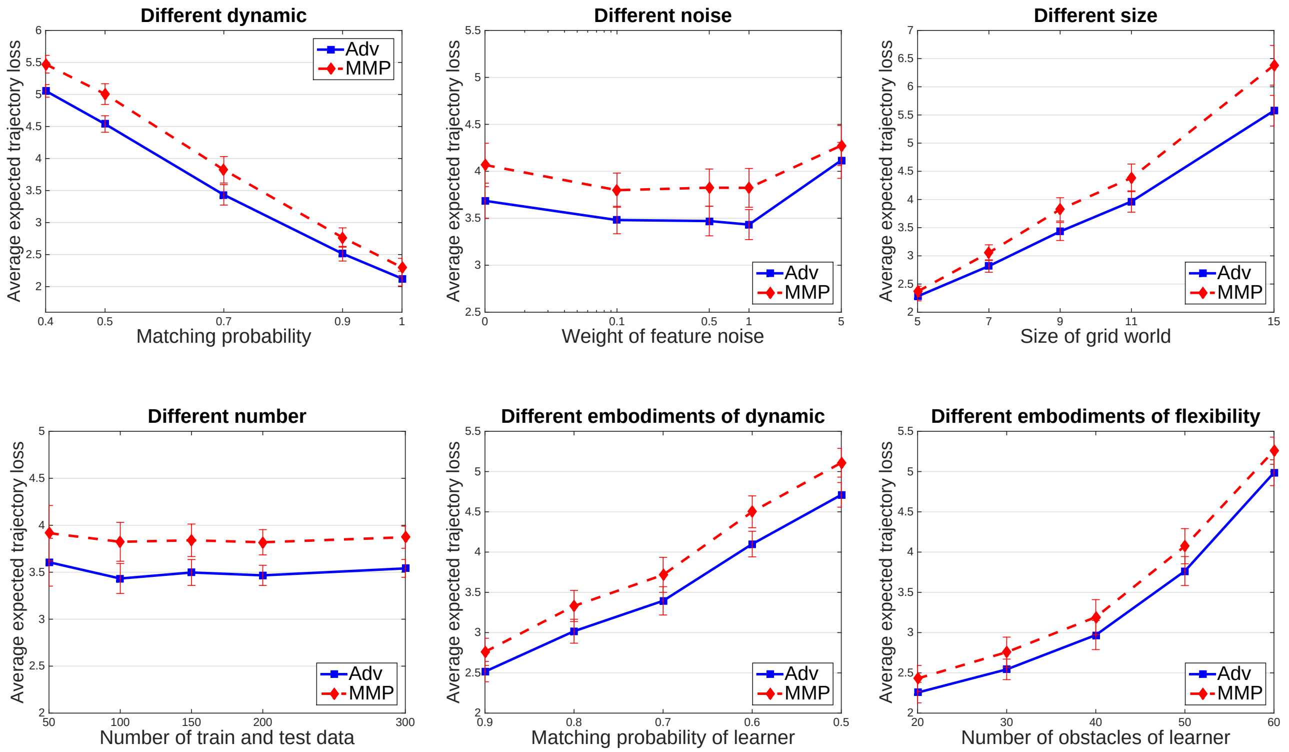

Figure 4: Experimental results with 95% confidence interval of various settings of the grid world's characteristics, including: the degree of stochasticity of the dynamics (top, left); varying amount of cost noise generating the demonstrator's trajectories (top, center); differences size of the grid world from 5x5 to 15x15 (top, right); different amount of training (test) data (bottom, left); the learner's dynamics differing from the demonstrator's (bottom, center); and the introduction of impassible obstacles for the learner (bottom, right).

The performance of the two methods under different embodiments is similarly evaluated according to the average expected trajectory loss of withheld test data, as shown in the final two plots of Figure 4. Our adversarial IOC method also outperforms MMP in these experimental settings.

5.2 LEARNING CAMERA CONTROL FROM DEMONSTRATION

We consider the task of learning to autonomously control a camera in a manner that appropriately captures the action of a basketball game based on human demonstrations of camera control [Chen and Carr, 2015]. The decision process characterizing camera control can be divided into a probabilistic model describing the state of the basketball game (the presence of players in different locations), and a dynamics model describing how camera movement controls effect the camera's state (quantized pan angle, θ, and quantize pan angle velocity, $\dot{\theta}$). As our focus is on the separation of rationalization and imitation evaluation measure, we assume that camera controls have no influence on the basketball game. Also based on this focus, we employ the empirical distribution of player locations rather than constructing a predictive model for those locations.

Our dataset is collected from high school basketball games. The camera recording the basketball game was operated by a human expert. The dataset consists of 46 sequences collected at 60Hz. The average number of frames for the sequences is 376. The output for each frame is the camera's horizontal pan angle, and the input is a 14 element vector that describes the state of the basketball game (the presence of players in different locations on the basketball court). The degree of the camera's pan angle in this dataset ranges from -30 degrees (left) to 30 degree (right), and we quantize the pan angle θ into discrete 61 levels. The pan angel velocity $\dot{\theta}$ of a particular frame is the difference between the current pan angle and the previous one, which is then mapped to 5 discrete levels $[-2, -1, 0, 1, 2]$ representing high speed of turning left to high speed of turning right. Overall, by combining the discrete pan angles and pan angle velocities, there are 305 total possible states for each frame. We use the first 23 sequences as our training dataset and the 23 remaining sequences as the testing dataset. We measure the performance of our adversarial IOC method and baseline methods using the average square

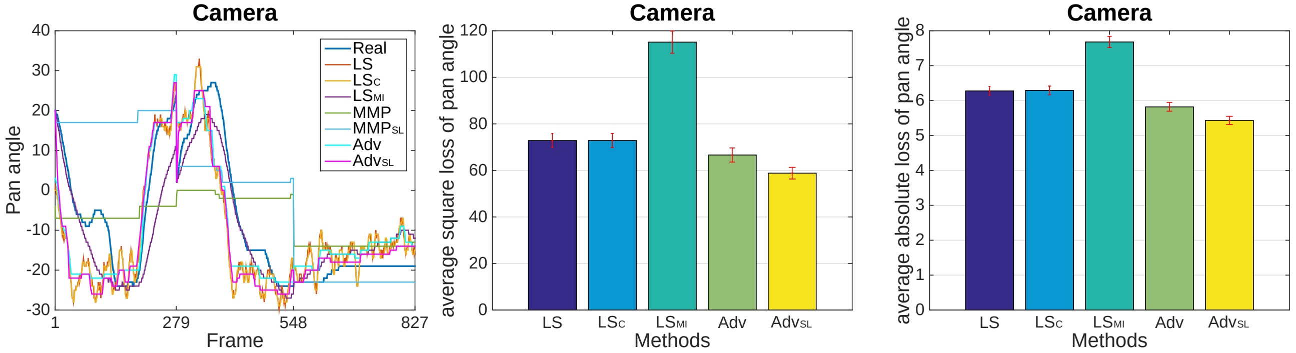

Figure 5: Imitating human camera operator's pan angle control (the Real trajectory on the left) using a regression approach, maximum margin planning, and our adversarial inverse optimal control method. Average squared loss and absolute loss of the imitator (with 95% mean confidence intervals estimates) are shown in the center and right plots, with maximum margin planning results suppressed due to being significantly worse and off of the presented scale.

loss per frame between pan angles:

$$\sum_{n=1}^{N}\sum_{t=1}^{T_n}(\theta_t^{(n)}-\hat{\theta}_t^{(n)})^2/\sum_{n=1}^{N}T_n. \tag{6}$$

We compare our adversarial structured prediction method with a few forms of least squares linear regression models: one that is not constrained by the camera dynamics (**LS**); one that is constrained by the empirical dynamics of the camera (**LS$_\mathbf{C}$**); and one Markovian-based model that also conditions on the previous camera location (**LS$_\mathbf{MI}$**). Additionally, we consider two variants of maximum marginal planning methods: **MMP$_\mathbf{SL}$** is provided with the starting location of the human-operated camera, while **MMP** is not. Similarly, **Adv$_\mathbf{SL}$** is our adversarial IOC method provided with the starting location of the human-operated camera, while **Adv** is not. Let X_t denotes the 14 entry feature vector of the state of the basketball game at timestep t. The feature vector $\phi(S_t)$ of our adversarial method in Definition 3 is a 33 entry vector $[\theta, \theta^2, \dot{\theta}, \dot{\theta}^2, \theta X_t, \dot{\theta} X_t]$, which combines the basketball game state features and the camera angle and angle velocity state. For the regression models, the estimated sequence is a standard linear regression method $\hat{\theta}_t = \hat{a} X_t + \hat{b}$ where $\hat{a}$ and $\hat{b}$ are trained from the training dataset. For the constrained regression method, the predicted camera angle is projected to the closest angle for which transitioning is feasible.

The result of a test sequence of our experiment is shown in the left plot of Figure 5. The first two regression methods are generally very noisy as the predicted pan angle changes rapidly based on the rapid changes of the underlying inputs corresponding to the game state. The Markovian regression model performs well initially, but diverges from the demonstrated trajectory over time. Both of the MMP methods have much worse performance than the other methods presented. Our adversarial approaches tend to be similar to the regression model, but are much less noisy and provide a closer match to the demonstrated trajectory with significantly lower amounts of squared and absolute loss, as shown in the other plots of Figure 5.

6 CONCLUSION

In this paper, we introduced an adversarial framework for imitation learning using inverse optimal control. It takes the form of a game between an adversary seeking to maximize loss by approximating the training data, and a learner seeking to minimize the loss. Algorithmically, our approach possesses similarities with existing inverse optimal control methods, while resolving some of the deficiencies of those methods (e.g., lack of consistency, sensitivity to low cost cycles) in a principled manner. A key benefit of our approach is that it separates the rationalization of demonstrated decision sequences with the learner's optimization of an imitative loss function. We focused this added flexibility on the problem of learning to imitate under differences in embodiment. This is an underexplored, but important problem for imitation learning to be employed in practice. We established the consistency and useful generalization bounds for our adversarial inverse optimal control approach. We developed and presented efficient algorithms for inference and learning under this formulation. Finally, we demonstrated the benefits of adversarial inverse optimal control in a set of synthetic experiments and an autonomous camera control task where an autonomous camera is trained based on observations of human camera control. In the future, we plan to apply the developed framework to imitation learning settings for robotics applications for which we believe that generalizing across different embodiments will be especially useful.

Acknowledgments

The research in this paper was supported in part by NSF NRI Award No. 1227495 and NSF RI Award No. 1526379.

References

[Abbeel and Ng, 2004] Abbeel, P. and Ng, A. Y. (2004). Apprenticeship learning via inverse reinforcement learning. In *Proc. International Conference on Machine Learning*, pages 1–8.

[Alissandrakis et al., 2002] Alissandrakis, A., Nehaniv, C. L., and Dautenhahn, K. (2002). Imitation with alice: Learning to imitate corresponding actions across dissimilar embodiments. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 32(4):482–496.

[Asif et al., 2015] Asif, K., Xing, W., Behpour, S., and Ziebart, B. D. (2015). Adversarial cost-sensitive classification. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*.

[Bellman, 1957] Bellman, R. (1957). A Markovian decision process. *Journal of Mathematics and Mechanics*, 6:679–684.

[Boularias et al., 2011] Boularias, A., Kober, J., and Peters, J. R. (2011). Relative entropy inverse reinforcement learning. In *International Conference on Artificial Intelligence and Statistics*, pages 182–189.

[Boyd et al., 1994] Boyd, S., El Ghaoui, L., Feron, E., and Balakrishnan, V. (1994). Linear matrix inequalities in system and control theory. *SIAM*, 15.

[Chen and Carr, 2015] Chen, J. and Carr, P. (2015). Mimicking human camera operators. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 215–222. IEEE.

[Crammer and Singer, 2002] Crammer, K. and Singer, Y. (2002). On the algorithmic implementation of multiclass kernel-based vector machines. *The Journal of Machine Learning Research*, 2:265–292.

[Grünwald and Dawid, 2004] Grünwald, P. D. and Dawid, A. P. (2004). Game theory, maximum entropy, minimum discrepancy, and robust Bayesian decision theory. *Annals of Statistics*, 32:1367–1433.

[Kalman, 1964] Kalman, R. (1964). When is a linear control system optimal? *Trans. ASME, J. Basic Engrg.*, 86:51–60.

[Kramer, 1998] Kramer, G. (1998). *Directed Information for Channels with Feedback*. PhD thesis, Swiss Federal Institute of Technology (ETH) Zurich.

[Levine et al., 2011] Levine, S., Popovic, Z., and Koltun, V. (2011). Nonlinear inverse reinforcement learning with gaussian processes. In *Advances in Neural Information Processing Systems*, pages 19–27.

[Liu, 2007] Liu, Y. (2007). Fisher consistency of multicategory support vector machines. In *International Conference on Artificial Intelligence and Statistics*, pages 291–298.

[McMahan et al., 2003] McMahan, H. B., Gordon, G. J., and Blum, A. (2003). Planning in the presence of cost functions controlled by an adversary. In *International Conference on Machine Learning*, pages 536–543.

[Monfort et al., 2015] Monfort, M., Lake, B. M., Ziebart, B., Lucey, P., and Tenenbaum, J. (2015). Softstar: Heuristic-guided probabilistic inference. In *Advances in Neural Information Processing Systems*, pages 2746–2754.

[Nehaniv and Dautenhahn, 2002] Nehaniv, C. L. and Dautenhahn, K. (2002). The correspondence problem. *Imitation in animals and artifacts*, 41.

[Ng and Russell, 2000] Ng, A. Y. and Russell, S. (2000). Algorithms for inverse reinforcement learning. In *Proc. International Conference on Machine Learning*, pages 663–670.

[Pomerleau, 1989] Pomerleau, D. (1989). Alvinn: An autonomous land vehicle in a neural network. In *Advances in Neural Information Processing Systems 1*.

[Ratliff et al., 2006] Ratliff, N., Bagnell, J. A., and Zinkevich, M. (2006). Maximum margin planning. In *Proc. International Conference on Machine Learning*, pages 729–736.

[Rust, 1988] Rust, J. (1988). Maximum likelihood estimation of discrete control processes. *SIAM Journal on Control and Optimization*, 26:1006–1024.

[Tewari and Bartlett, 2007] Tewari, A. and Bartlett, P. L. (2007). On the consistency of multiclass classification methods. *The Journal of Machine Learning Research*, 8:1007–1025.

[Topsøe, 1979] Topsøe, F. (1979). Information theoretical optimization techniques. *Kybernetika*, 15(1):8–27.

[Wang et al., 2015] Wang, H., Xing, W., Asif, K., and Ziebart, B. D. (2015). Adversarial prediction games for multivariate losses. In *Advances in Neural Information Processing Systems (NIPS)*.

[Ziebart, 2010] Ziebart, B. D. (2010). *Modeling Purposeful Adaptive Behavior with the Principle of Maximum Causal Entropy*. PhD thesis, Carnegie Mellon University.

[Ziebart et al., 2010] Ziebart, B. D., Bagnell, J. A., and Dey, A. K. (2010). Modeling interaction via the principle of maximum causal entropy. In *Proc. International Conference on Machine Learning*, pages 1255–1262.