Active Inverse Learning in Stackelberg Trajectory Games

Yue Yu, Jacob Levy, Negar Mehr, David Fridovich-Keil, and Ufuk Topcu

***Abstract*— Game-theoretic inverse learning is the problem of inferring the players' objectives from their actions. We formulate an inverse learning problem in a Stackelberg game between a leader and a follower, where each player's action is the trajectory of a dynamical system. We propose an active inverse learning method for the leader to infer which hypothesis among a finite set of candidates describes the follower's objective function. Instead of using passively observed trajectories like existing methods, the proposed method actively maximizes the differences in the follower's trajectories under different hypotheses to accelerate the leader's inference. We demonstrate the proposed method in a receding-horizon repeated trajectory game. Compared with uniformly random inputs, the leader inputs provided by the proposed method accelerate the convergence of the probability of different hypotheses conditioned on the follower's trajectory by orders of magnitude.**

I. Introduction

Learning to predict human behavior is a critical challenge in human-robot interaction. It enables robots to customize their strategies in various applications, including assisted driving [1], [2], traffic management [3], [4], and, in general, mitigating conflicts in human-in-the-loop robotic systems.

Game-theoretic inverse learning helps robots explain and predict human behavior in noncooperative interactions [5], [6], [7], [8], [9], [10], [11], [3]. The idea is to first model humans' objectives as parameterized functions, then infer the parameter value such that the corresponding game-theoretic strategies—such as Nash or Stackelberg equilibrium strategies—match the humans' actions in a dataset. Game-theoretic inverse learning is a necessary step in understanding human-robot interactions [12], [13], [3] and designing incentives for multiagent systems [14], [15].

The existing game-theoretic inverse learning methods are *passive*, which can be data inefficient. In particular, these methods record the dataset of human actions before and independently of the inference process. Hence some actions in the recorded dataset can be uninformative for inference purposes, or simply redundant. As a result, passive inverse learning lacks the data efficiency to enable rapid inference and support online real-time decision-making.

In contrast to passive inverse learning, active inverse learning helps robots to infer human intentions in cooperative interactions by actively provoking informative human actions. For example, when learning objectives that explain human's ranking or rating of presented options, active inverse learning methods first provoke informative human actions and record them in the dataset, then infer the human's objective function, and repeat this process if necessary [16], [17], [18], [19], [20]. These methods ensure that the human's actions are informative by maximizing the volume removed from the hypothesis space [18], [21], [22], [23], [24], [25] or by maximizing the information gain [26], [27], [4], [28], [29], [30]. By integrating dataset updates with inference, active inverse learning provides practical solutions for inferring human intentions from limited interactions.

Despite its successes, active inverse learning still requires investigation in noncooperative interactions. The existing active inverse learning methods rely on querying humans who volunteer informative responses. In contrast, humans in noncooperative interactions only take actions that optimize their own objectives. Therefore, how to provoke informative actions from noncooperative humans that reveal their objectives is, to our best knowledge, still an open question.

We formulate an inverse learning problem in a Stackelberg game where a rational leader, such as a robot, is inferring which hypothesis among finitely many candidates best explains the behavior of a boundedly rational follower, such as a human. This problem is particularly relevant in shared autonomy, *e.g.*, when an autopilot is inferring the type of a newly encountered human driver. We model each player's action as the trajectory of a linear time-invariant system. The follower tracks a linear function of the leader's trajectory—similar to how a human driver tracks the trajectory recommended by an autopilot—using a maximum-entropy linear quadratic regulator—which contains a parameterized objective function—that models bounded rationality in human decision-making [3]. The leader determines which hypothesis is most likely using the probability of each hypothesis conditioned on the follower's state trajectory.

We propose an active inverse learning method to provoke informative trajectories from the follower by optimizing the leader's inputs. In this optimization, we maximize the differences in the follower's trajectory distributions under different hypotheses. We show that this optimization is a difference-of-convex program [31], which can be solved efficiently via the convex-concave procedure [32]. We evaluate the performance of the proposed method in a receding-horizon repeated trajectory game. Compared with random inputs, the leader inputs provided by our method accelerate the convergence of the probability of different hypotheses conditioned on the follower's trajectory by orders of magnitude.

Notation: We let $\mathbb{R}$, $\mathbb{R}_{\geq 0}$, and $\mathbb{N}$ denote the set of real, nonnegative real numbers, and nonnegative integers, respectively. We let $\mathbb{S}^n_{\succeq 0}$ and $\mathbb{S}^n_{\succ 0}$ denote the set of n by n symmetric

Y. Yu, J. Levy, D. Fridovich-Keil, and U. Topcu are with the Oden Institute for Computational Engineering and Sciences, The University of Texas at Austin, Austin, TX, 78712, USA (emails: yueyu@utexas.edu, jake.levy@utexas.edu, dfk@utexas.edu, utopcu@utexas.edu). N. Mehr is with the Department of Aerospace Engineering at the University of Illinois at Urbana-Champaign, Urbana, IL, 61801, USA (email: negar@illinois.edu).

positive semidefinite and positive definite matrices, respectively. For any $x \in \mathbb{R}^n$, we let $\|x\| \coloneqq \sqrt{x^\top x}$, $\|x\|_1 \coloneqq \sum_{i=1}^n |x_i|$, $\|x\|_\infty \coloneqq \max_{1\le i\le n} |x_i|$, and $\|x\|_A^2 \coloneqq x^\top A x$ for all $A \in \mathbb{S}^n_{\succeq 0}$. We let 0_n denote the n-dimensional zero vector; I_n and $0_{n\times n}$ denote the n by n identity and zero matrix, respectively. We let $\mathcal{N}(\mu, \Sigma)$ denote the Gaussian distribution with mean $\mu \in \mathbb{R}^n$ and variance $\Sigma \in \mathbb{S}^n_{\succ 0}$. Given $n_1, n_2 \in \mathbb{N}$, we let $[n_1, n_2]$ denote the set of integers between n_1 and n_2. Given $a_i \in \mathbb{R}^n$ for all $i \in \mathbb{N}$, we let $a_{i:j} \coloneqq \begin{bmatrix} a_i^\top & a_{i+1}^\top & \cdots & a_j^\top \end{bmatrix}^\top$ for all $i < j$, $i, j \in \mathbb{N}$.

II. Linear quadratic Stackelberg trajectory games

We introduce a Stackelberg game between a rational leader, such as a robot, and a boundedly rational follower, such as a human with noisy behavior. The players' actions are trajectories of stochastic linear time-invariant systems.

A. The dynamics of the players' systems

We assume that the leader's state evolves according to the following discrete-time linear time-invariant dynamics:

$$x^{\mathrm{L}}_{t+1} = A^{\mathrm{L}} x^{\mathrm{L}}_t + B^{\mathrm{L}} u^{\mathrm{L}}_t + w^{\mathrm{L}}_t \tag{1}$$

for all $t \in \mathbb{N}$, where $x^{\mathrm{L}}_t \in \mathbb{R}^{n_{\mathrm{L}}}$, $u^{\mathrm{L}}_t \in \mathbb{R}^{m_{\mathrm{L}}}$, and $w^{\mathrm{L}}_t \in \mathbb{R}^{\mathrm{L}}$ are the state, input, and disturbance of the system at time $t \in \mathbb{N}$, respectively; $A^{\mathrm{L}} \in \mathbb{R}^{n_{\mathrm{L}}\times n_{\mathrm{L}}}$ and $B^{\mathrm{L}} \in \mathbb{R}^{n_{\mathrm{L}}\times m_{\mathrm{L}}}$ are the leader's system parameters. Equation (1) characterizes the dynamics of many common robots, such as the kinematics of rovers and drones.

Similarly, the follower's state evolves according to the following dynamics:

$$x^{\mathrm{F}}_{t+1} = A^{\mathrm{F}} x^{\mathrm{F}}_t + B^{\mathrm{F}} u^{\mathrm{F}}_t + w^{\mathrm{F}}_t \tag{2}$$

for all $t \in \mathbb{N}$, where $x^{\mathrm{F}}_t \in \mathbb{R}^{n_{\mathrm{L}}}$, $u^{\mathrm{F}}_t \in \mathbb{R}^{m_{\mathrm{L}}}$, and $w^{\mathrm{F}}_t \in \mathbb{R}^{n_{\mathrm{L}}}$ denote the state, input, and disturbance of the system at time $t \in \mathbb{N}$, respectively; $A^{\mathrm{F}} \in \mathbb{R}^{n_{\mathrm{L}}\times n_{\mathrm{L}}}$ and $B^{\mathrm{F}} \in \mathbb{R}^{n_{\mathrm{F}}\times m_{\mathrm{F}}}$ are the follower's system parameters.

Throughout, we assume that the disturbance in the leader and the follower's systems are independent, identically distributed Gaussian vectors, *i.e.*, there exists $\Omega^{\mathrm{L}} \in \mathbb{S}^{n_{\mathrm{L}}}_{\succ 0}$ and $\Omega^{\mathrm{F}} \in \mathbb{S}^{n_{\mathrm{F}}}_{\succ 0}$ such that, for any $t \in \mathbb{N}$, we have

$$w^{\mathrm{L}}_t \sim \mathcal{N}(0_{n_{\mathrm{L}}}, \Omega^{\mathrm{L}}),\ w^{\mathrm{F}}_t \sim \mathcal{N}(0_{n_{\mathrm{L}}}, \Omega^{\mathrm{F}}). \tag{3}$$

B. The players' objectives

We assume that the follower's objective is to track a linear function of the leader's trajectory. In particular, we let $M^{\mathrm{F}} \in \mathbb{R}^{n_{\mathrm{F}}\times n_{\mathrm{L}}}$ denote a matrix that maps the leader's internal state to an output reference observable to the follower. Let $x^{\mathrm{L}}_{0:\tau}$ denote a leader trajectory of length $\tau \in \mathbb{N}$. The follower's objective is to simultaneously track the corresponding output trajectory $\{M^{\mathrm{F}} x^{\mathrm{L}}_0, M^{\mathrm{F}} x^{\mathrm{L}}_1, \ldots, M^{\mathrm{F}} x^{\mathrm{L}}_\tau\}$ and minimize its input efforts.

We assume that the follower is boundedly rational and chooses its input according to the maximum entropy principle, which states that the distribution of u^{F}_t conditioned on x^{F}_t is Gaussian, *i.e.*, $\mu^{\mathrm{F}}_t | x^{\mathrm{F}}_t \sim \mathcal{N}(\mu_t, \Sigma_t)$ for some $\mu_t \in \mathbb{R}^{n_{\mathrm{F}}}$ and $\Sigma_t \in \mathbb{R}^{n_{\mathrm{F}}}$ [3]. In particular, $(\mu_{0:\tau-1}, \Sigma_{0:\tau-1})$ is optimal for the following stochastic trajectory optimization problem:

$$\begin{aligned} \underset{\mu_{0:\tau-1},\Sigma_{0:\tau-1}}{\text{minimize}} \quad & \textstyle\sum_{t=0}^{\tau} \mathbb{E}\left[\frac{1}{2}\left\|x^{\mathrm{F}}_t - M^{\mathrm{F}} x^{\mathrm{L}}_t\right\|^2_{Q^{\mathrm{F}}}\right] \\ & \textstyle+\frac{1}{2}\sum_{t=0}^{\tau-1}\left(\mathbb{E}\left[\left\|u^{\mathrm{F}}_t\right\|^2_{R^{\mathrm{F}}}\right] - \log\det\Sigma_t\right) \\ \text{subject to} \quad & x^{\mathrm{F}}_{t+1} = A^{\mathrm{F}} x^{\mathrm{F}}_t + B^{\mathrm{F}} u^{\mathrm{F}}_t + w^{\mathrm{F}}_t,\ x^{\mathrm{F}}_0 = \hat{x}^{\mathrm{F}}_0, \\ & u^{\mathrm{F}}_t | x^{\mathrm{F}}_t \sim \mathcal{N}(\mu_t, \Sigma_t),\ w^{\mathrm{F}}_t \sim \mathcal{N}(0_{n_{\mathrm{F}}}, \Omega^{\mathrm{F}}), \\ & t \in [0, \tau-1], \end{aligned} \tag{4}$$

where $\hat{x}^{\mathrm{F}}_0 \in \mathbb{R}^{\mathrm{F}}$ is the initial state of the follower's system, $\mathbb{E}[\cdot]$ denotes the expectation; $Q^{\mathrm{F}} \in \mathbb{S}^n_{\succeq 0}$, and $R^{\mathrm{F}} \in \mathbb{S}^n_{\succ 0}$ are the follower's cost parameters. The objective function in optimization (4) captures boundedly rational human decisions: it is noisy but centers around a cost-minimizing rational decision.

The following proposition provides a closed-form formula for the solution of optimization (4).

Proposition 1. *Let*

$$P^F_\tau = Q^F,\ P^F = Q^F + (A^F)^\top P^F_{t+1} E^F_t, \tag{5a}$$
$$F^F_t = B^F(R + (B^F)^\top P^F_{t+1} B^F)^{-1}(B^F)^\top, \tag{5b}$$
$$E^F_t = A^F - F^F_t P^F_{t+1} A^F, \tag{5c}$$
$$q^F_\tau = -Q^F M^F x^L_\tau,\ q^F_t = (E^F_t)^\top q^F_{t+1} - Q^F M^F x^L_t, \tag{5d}$$

for all $t \in [0, \tau-1]$. Given $x^L_{0:\tau}$, $(\mu_{0:\tau-1}, \Sigma_{0:\tau-1})$ is optimal for optimization (4) *if and only if*

$$\Sigma_t = (R^F + (B^F)^\top P^F_{t+1} B^F)^{-1}, \tag{6a}$$
$$\mu_t = -\Sigma_t (B^F)^\top (P^F_{t+1} A^F x^F_t + q^F_{t+1}). \tag{6b}$$

for all $t \in [0, \tau-1]$. Furthermore, if the constraints in (4) *hold, then $x^F_t \sim \mathcal{N}(\xi_t, \Lambda_t)$ for all $t \in [0, \tau]$, where*

$$\xi_{t+1} = E^F_t \xi_t - F^F_t q^F_{t+1}, \tag{7a}$$
$$\Lambda_{t+1} = E^F_t \Lambda_t (E^F_t)^\top + F^F_t + \Omega^F, \tag{7b}$$

for all $t \in [0, \tau-1]$, with $\xi_0 = \hat{x}^F_0$ and $\Lambda_0 = 0_{n_F \times n_F}$..

Proof. See Appendix. □

The leader's objective is to minimize a cost function that jointly depends on the expected follower's trajectory and the leader's trajectory. To this end, we assume that the leader acts rationally and chooses its trajectory $x^{\mathrm{L}}_{0:\tau}$ as a solution to the following trajectory optimization problem:

$$\begin{aligned} \underset{u_{0:\tau-1}}{\text{minimize}} \quad & \mathbb{E}[f(x^{\mathrm{L}}_{0:\tau}, x^{\mathrm{F}}_{0:\tau})] + g(u^{\mathrm{L}}_{0:\tau}) \\ \text{subject to} \quad & x^{\mathrm{L}}_{t+1} = A^{\mathrm{L}} x^{\mathrm{L}}_t + B^{\mathrm{L}} u^{\mathrm{L}}_t + w^{\mathrm{L}}_t,\ x^{\mathrm{L}}_0 = \hat{x}^{\mathrm{L}}_0, \\ & x^{\mathrm{F}}_{t+1} = A^{\mathrm{F}} x^{\mathrm{F}}_t + B^{\mathrm{F}} u^{\mathrm{F}}_t + w^{\mathrm{F}}_t,\ x^{\mathrm{F}}_0 = \hat{x}^{\mathrm{F}}_0, \\ & w^{\mathrm{L}}_t \sim \mathcal{N}(0_{n_{\mathrm{L}}}, \Omega^{\mathrm{L}}),\ w^{\mathrm{F}}_t \sim \mathcal{N}(0_{n_{\mathrm{F}}}, \Omega^{\mathrm{F}}), \\ & u_t \in \mathbb{U},\ u^{\mathrm{F}}_t | x^{\mathrm{F}}_t \sim \mathcal{N}(\mu_t, \Sigma_t),\ t \in [0, \tau-1], \\ & (\mu_{0:\tau-1}, \Sigma_{0:\tau-1}) \text{ is optimal for (4)}, \end{aligned} \tag{8}$$

where $\hat{x}^{\mathrm{L}}_0 \in \mathbb{R}^{n_{\mathrm{L}}}$ is the initial state of the leader's system, $\mathbb{U} \subset \mathbb{R}^{m_{\mathrm{F}}}$ is the set of feasible leader inputs at each time. Furthermore, $f : \mathbb{R}^{(\tau+1)n_{\mathrm{L}}} \times \mathbb{R}^{(\tau+1)n_{\mathrm{L}}} \to \mathbb{R}$ is the leader's cost function that jointly depends on the leader's state trajectory $x^{\mathrm{L}}_{0:\tau}$ and the follower's state trajectory $x^{\mathrm{F}}_{0:\tau}$;

$g : \mathbb{R}^{\tau m_{\text{L}}} \to \mathbb{R}$ is a cost function that only depends on the leader's input trajectory. By choosing different functions for f and g, optimization (8) achieves different trade-offs between optimizing the leader and the follower's trajectory.

Problem (8) is a Stackelberg game, also known as a *bilevel optimization*. See [33] and reference therein for a detailed discussion on bilevel optimization.

III. Active inverse learning via difference maximization

Given the Stackelberg trajectory game introduced in Section II, we now consider the case where the leader does not know the parameter tuple $(Q^{\text{F}}, R^{\text{F}}, M^{\text{F}})$ in the follower's objective, except that it is one of finitely many candidates. In other words, the leader knows that there exist $Q^1, \dots, Q^d \in \mathbb{R}^{n^{\text{F}} \times n^{\text{F}}}$, $R^1, \dots, R^d \in \mathbb{R}^{m^{\text{F}} \times m^{\text{F}}}$, and $M^1, \dots, M^d \in \mathbb{R}^{n^{\text{F}} \times n^{\text{L}}}$ such that

$$(Q^{\text{F}}, R^{\text{F}}, M^{\text{F}}) = (Q^i, R^i, M^i) \tag{9}$$

for some $i \in [1, d]$. This case arises, for example, when a robot already learned different types of human behavior offline but needs to determine the type of a newly encountered human via online interaction. In the following, we let $\theta^{\text{F}} \coloneqq (Q^{\text{F}}, R^{\text{F}}, M^{\text{F}})$ and $\theta^i \coloneqq (Q^i, R^i, M^i)$ for all $i \in [1, d]$. We say that *hypothesis i is true* if (9) holds.

Based on a prior probability distribution of all hypotheses that gives the value of $\mathbb{P}(\theta^{\text{F}} = \theta^i | x_0^{\text{F}})$ for all $i \in [1, d]$ and the follower's trajectory $x_{1:\tau}^{\text{F}}$, the leader can infer whether hypothesis i is more likely than hypothesis j to be true by computing the following ratio:

$$\frac{\mathbb{P}(\theta^{\text{F}} = \theta^i | x_{0:\tau}^{\text{F}})}{\mathbb{P}(\theta^{\text{F}} = \theta^j | x_{0:\tau}^{\text{F}})} = \frac{\mathbb{P}(\theta^{\text{F}} = \theta^i | x_0^{\text{F}}) \mathbb{P}(x_{1:\tau}^{\text{F}} | \theta^{\text{F}} = \theta^i)}{\mathbb{P}(\theta^{\text{F}} = \theta^j | x_0^{\text{F}}) \mathbb{P}(x_{1:\tau}^{\text{F}} | \theta^{\text{F}} = \theta^j)} \tag{10}$$

The ratio in (10) being bigger than one implies that trajectory $x_{0:\tau}^{\text{F}}$ is more likely to occur under hypothesis i rather than hypothesis j, and vice versa.

However, observing the follower's trajectory can be uninformative for the inference above if the trajectories under different hypotheses are similar. For example, suppose that

$$\mathbb{P}(x_{1:\tau}^{\text{F}} | \theta^{\text{F}} = \theta^i) \approx \mathbb{P}(x_{1:\tau}^{\text{F}} | \theta^{\text{F}} = \theta^j) \tag{11}$$

for some $i \neq j$, then (10) implies that $\frac{\mathbb{P}(\theta^{\text{F}} = \theta^i | x_{0:\tau}^{\text{F}})}{\mathbb{P}(\theta^{\text{F}} = \theta^j | x_{0:\tau}^{\text{F}})} \approx \frac{\mathbb{P}(\theta^{\text{F}} = \theta^i | x_0^{\text{F}})}{\mathbb{P}(\theta^{\text{F}} = \theta^j | x_0^{\text{F}})}$. In other words, observing the follower's ongoing trajectory $x_{1:\tau}^{\text{F}}$ does not help the leader distinguish hypothesis i from j. In the following, we discuss how the leader can actively avoid the scenarios where (11) happens by making the follower's trajectories under different hypotheses as different as possible.

A. The follower's trajectories under different hypotheses

To avoid the scenarios where (11) happens, it suffices to make the follower's trajectory distributions under different hypotheses as different as possible. To this end, we first take a closer look at the follower's trajectory. Proposition 1 shows that the follower's state at each time is a Gaussian random variable. Particularly, let

$$E_t^i = A^{\text{F}} - F_t^i P_{t+1}^i A^{\text{F}}, \tag{12a}$$

$$F_t^i = B^{\text{F}} (R^{\text{F}} + (B^{\text{F}})^\top P_{t+1}^i B^{\text{F}})^{-1} (B^{\text{F}})^\top, \tag{12b}$$

$$P_\tau^i = Q^i,\ P_t^i = Q^i + (A^{\text{F}})^\top P_{t+1}^i E_t^i, \tag{12c}$$

$$\Lambda_0^i = 0_{n_{\text{F}} \times n_{\text{F}}},\ \Lambda_{t+1}^i = E_t^i \Lambda_t^i (E_t^i)^\top + F_t^i + \Omega^{\text{F}}, \tag{12d}$$

$$q_\tau^i = -Q^i M^i x_\tau^{\text{L}},\ q_t^i = (E_t^i)^\top q_{t+1}^i - Q^i M^i x_t^{\text{L}}, \tag{12e}$$

$$\xi_0^i = \hat{x}_0^{\text{F}},\ \xi_{t+1}^i = E_t^i \xi_t^i + F_t^i q_{t+1}^i, \tag{12f}$$

for all $t \in [0, \tau - 1]$. Proposition 1 implies that, if hypothesis i is true, *i.e.*, $\theta^{\text{F}} = \theta^i$, then

$$x_t^{\text{F}} \sim \mathcal{G}_t^i \coloneqq \mathcal{N}(\xi_t^i, \Lambda_t^i). \tag{13}$$

To measure the differences between the trajectory distribution under different hypotheses, we introduce a distance function. To this end, let

$$\mathbb{D} \coloneqq \{(i, j) | i < j,\ i, j \in [1, d]\}. \tag{14}$$

A popular measure of the difference between two Gaussian distributions is the *KL-divergence*. Given $(i, j) \in \mathbb{D}$ and $t \in [1, \tau]$, let $\mathcal{G}_t^i$ and $\mathcal{G}_t^j$ be defined as in (13). The KL-divergence from $\mathcal{G}_t^i$ to $\mathcal{G}_t^j$ is as follows:

$$\begin{aligned} D_{KL}(\mathcal{G}_t^i || \mathcal{G}_t^j) \coloneqq & \tfrac{1}{2} \left\| \xi_t^i - \xi_t^j \right\|^2_{(\Lambda_t^j)^{-1}} - \tfrac{(\tau+1) n_{\text{F}}}{2} \\ & + \tfrac{1}{2} \log \left(\tfrac{\det \Lambda_t^j}{\det \Lambda_t^i} \right) + \operatorname{tr}((\Lambda_t^j)^{-1} \Lambda_t^i). \end{aligned} \tag{15}$$

Notice that, since $\Omega^{\text{F}} \in \mathbb{S}^{n_{\text{F}}}_{\succ 0}$, (12) implies that $\Lambda_t^i \in \mathbb{S}^{n_{\text{F}}}_{\succ 0}$ for all $t \in [1, \tau]$ and $k \in [1, d]$. However, KL divergence is not symmetric, *i.e.*, $D_{KL}(\mathcal{G}_t^i || \mathcal{G}_t^j) \neq D_{KL}(\mathcal{G}_t^j || \mathcal{G}_t^i)$. To define a symmetric distance function, we propose the following function

$$D(\mathcal{G}_t^i, \mathcal{G}_t^j) \coloneqq \left\| \xi_t^i - \xi_t^j \right\|^2_{(\Lambda_t^i)^{-1} + (\Lambda_t^j)^{-1}} \tag{16}$$

for all $t \in [1, \tau]$ and $(i, j) \in \mathbb{D}$. The intuition behind this distance function is to first evaluate (up to a constant of 2, for the convenience of notation) the sum of $D_{KL}(\mathcal{G}_t^i || \mathcal{G}_t^j)$ and $D_{KL}(\mathcal{G}_t^j || \mathcal{G}_t^i)$, then remove the terms that are independent of ξ_t^i and ξ_t^j, which are, as suggested by (12), independent of the leader's trajectory. Later, we use this function to optimize the leader's inputs.

B. Maximizing the worst-case pairwise distance

To avoid the scenarios where (11) happens, we need to maximize the value of the distance function in (16) for any $(i, j) \in \mathbb{D}$. To this end, define the following *worst-case distance function*, which evaluates the minimum value of function (16) among all $(i, j) \in \mathbb{D}$:

$$\begin{aligned} & \min_{(i,j) \in \mathbb{D}} \left\{ \textstyle\sum_{t=1}^\tau \left\| \xi_t^i - \xi_t^i \right\|^2_{(\Lambda_t^i)^{-1} + (\Lambda_t^j)^{-1}} \right\} \\ & = \sum_{(i,j) \in \mathbb{D}} \textstyle\sum_{t=1}^\tau \left\| \xi_t^i - \xi_t^i \right\|^2_{(\Lambda_t^i)^{-1} + (\Lambda_t^j)^{-1}} \\ & - \max_{(i,j) \in \mathbb{D}} \left\{ \sum_{(k,l) \in \mathbb{D} \setminus \{(i,j)\}} \textstyle\sum_{t=1}^\tau \left\| \xi_t^k - \xi_t^l \right\|^2_{(\Lambda^k)_t^{-1} + (\Lambda^l)_t^{-1}} \right\}. \end{aligned} \tag{17}$$

The second step in (17) uses the fact that, given any $\alpha_1, \dots, \alpha_n \in \mathbb{R}$, we have

$$\min_{i \in [1,n]} \alpha_i = \sum_{i \in [1,n]} \alpha_i - \max_{i \in [1,n]} \sum_{j \in [1,n], j \neq i} \alpha_j.$$

Notice that the distance in (17) does not include the terms for $t = 0$ since, due to (12), $\xi_0^i = \xi_0^j$ for all $(i, j) \in \mathbb{D}$.

Based on the worst-case distance in (17), we propose to optimize the leader's input trajectory $u_{0:\tau-1}^{\text{L}}$ via solving optimization (18) (see next page) instead of optimization (8), where E_t^i, F_t^i and Λ_t^i are given by (12). The idea of optimization (18) is to first approximate the leader's state trajectory $x_{0:\tau}^{\text{L}}$ with its expectation, denoted by $\eta_{0:\tau}$, which satisfies the averaged dynamics $\eta_{t+1} = A^{\text{L}}\eta_t + B^{\text{L}}u_t^{\text{L}}$. This approximation replaces the disturbance term w_t^{L} in (8) with its most probable estimation of its distribution $\mathcal{N}(0_{n_{\text{L}}}, \Omega^{\text{L}})$, given by $w_t^{\text{L}} = 0_{n_{\text{L}}}$. In addition, optimization (18) replaces the $\mathbb{E}[f(x_{0:\tau}^{\text{L}}, x_{0:\tau}^{\text{F}})]$ term in (8) with the negative of the worst-case distance in (17). In particular, the constraints in (18) imply that

$$\max_{(i,j)\in\mathbb{D}} \left\{ \sum_{(k,l)\in\mathbb{D}\setminus\{(i,j)\}} \sum_{t=1}^{\tau} \left\| \xi_t^k - \xi_t^l \right\|^2_{(\Lambda^k)_t^{-1}+(\Lambda^l)_t^{-1}} \right\} \leq s.$$

Since the objective function in (18) is minimizing the value of s, the above inequality holds as an equality at optimality. Hence the objective function in (18) is equivalent to the one in (8) except that the $\mathbb{E}[f(x_{0:\tau}^{\text{L}}, x_{0:\tau}^{\text{F}})]$ term in (8) is replaced by the negative of the worst-case distance in (17). The idea of this replacement is to maximize the worst-case distance in (17), which ensures that the distance in (16) is large for any $(i, j) \in \mathbb{D}$.

Optimization (18) is a difference-of-convex program: all of its constraints are convex, but its objective function is the difference between two convex functions [31]. A popular solution method for difference-of-convex programs is the *convex-concave procedure*, which guarantees global convergence to a stationary point [34], and provides locally optimal solutions in practice [32].

IV. Numerical Experiments

We empirically demonstrate our results using a receding-horizon repeated trajectory game between a boundedly rational follower that controls one ground rover and a rational leader that controls multiple ground rovers. The leader is inferring which leading rover the follower is following. At each time step, the players play a Stackelberg trajectory game and implement only the first step of theire respective input trajectories.

We define the game parameters as follows. We model the dynamics of the following rover using an instance of (1), where $A^{\text{F}} = \exp\left(\delta \begin{bmatrix} 0_{2\times2} & I_2 \\ 0_{2\times2} & 0_{2\times2} \end{bmatrix}\right)$, $B^{\text{F}} = \int_0^\delta \exp\left(t \begin{bmatrix} 0_{2\times2} & I_2 \\ 0_{2\times2} & 0_{2\times2} \end{bmatrix}\right) \mathrm{d}t \begin{bmatrix} 0_{2\times2} \\ I_2 \end{bmatrix}$, $\Omega^{\text{F}} = \frac{1}{1000} I_4$, and $\delta = 0.2$ is the discretization step size. For the leader, we model the joint dynamics of $d \in \mathbb{N}$ ground rovers using the joint dynamics of d double-integrators, where we let $A^{\text{L}} = I_d \otimes A$, $B^{\text{L}} = I_d \otimes B$, and $\Omega^{\text{L}} = \frac{1}{1000} I_{4d\times4d}$. For the follower's trajectory optimization in (4), we let $Q^i = \mathrm{diag}([\,1\;1\;0\;0\,])$, $R^i = \frac{1}{100} I_2$, and $M^i = [\,0_{4\times4(i-1)}\; I_4\; 0_{4\times4(d-i)}\,]$ for all $i \in [1, d]$. Without loss of generality, we assume $(Q^{\text{F}}, R^{\text{F}}, M^{\text{F}}) = (Q^1, R^1, M^1)$. For the leader's trajectory optimization in (8), we let $\mathbb{U} \coloneqq \{u \in \mathbb{R}^{2d} | \, \|u\|_\infty \leq 2\}$, and $g(u_{0:\tau-1}^{\text{L}}) \sum_{t=1}^{\tau-2} \left\| u_{t+1}^{\text{L}} - u_t^{\text{L}} \right\|^2$.

We demonstrate the proposed learning methods using the simulation of a receding-horizon repeated trajectory game. At time $t\delta$ for some $t \in \mathbb{N}$, the leader first observes its current state $\overline{x}_t^{\text{L}}$ and the follower's current state $\overline{x}_t^{\text{F}}$, then solves optimization (18) with $\hat{x}_0^{\text{L}} = \overline{x}_t^{\text{L}}$ and $\hat{x}_0^{\text{F}} = \overline{x}_t^{\text{F}}$ to obtain the optimal input sequence $u_{0:\tau-1}^{\text{L}}$. Next, the leader simulates a state trajectory $x_{0:\tau-1}^{\text{L}}$ according to (1) and shares it with the follower, then applies u_0^{L}. Meanwhile, at time $t\delta$, the follower observes its current state $\hat{x}_t^{\text{F}}$ and receives the leader's simulated trajectory $x_{0:\tau}^{\text{L}}$. Next the follower solves optimization (4) with $\hat{x}_0^{\text{F}} = \overline{x}_t^{\text{F}}$ and obtain the optimal mean and covariance sequences $(\mu_{0:\tau-1}, \Sigma_{0:\tau-1})$, and finally samples $u_0^{\text{F}} \sim \mathcal{N}(\mu_0, \Sigma_0)$ and applies u_0^{F} to its system.

We simulate the players' trajectories in this receding-horizon repeated trajectory game, where we solve the leader's trajectory optimization (18) using the convex-concave procedure [32]. Fig. 1 shows the position trajectories of the leader's rovers when $d = 3$ and $d = 5$. We can see that, to make the follower's trajectories under different hypotheses as different as possible, optimization (18) ensures that different leading rovers are moving in different directions.

We further showcase the advantage of the proposed method in terms of distinguishing different hypotheses. To this end, we let $p^\star = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}^\top \in \mathbb{R}^d$ denote the ground truth probability of all hypotheses (note that we let $(Q^{\text{F}}, R^{\text{F}}, M^{\text{F}}) = (Q^1, R^1, M^1)$). In addition, we let p^t denote the d-dimensional vector such that

$$p_i^t \coloneqq \mathbb{P}((Q^{\text{F}}, R^{\text{F}}, M^{\text{F}}) = (Q^i, R^i, M^i) | \overline{x}_{0:t}^{\text{F}}) \tag{19}$$

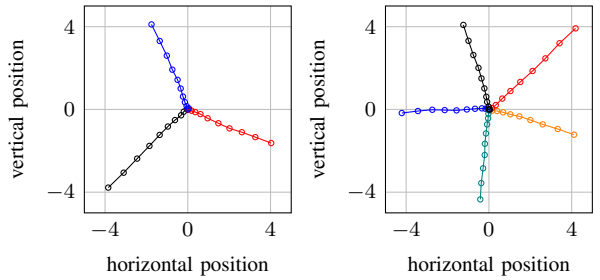
for all $i \in [1, d]$, where we let $\mathbb{P}((Q^{\text{F}}, R^{\text{F}}, M^{\text{F}}) = (Q^i, R^i, M^i) | \overline{x}_0^{\text{F}}) \coloneqq \frac{1}{d}$ for all $i \in [1, d]$, *i.e.*, we choose the uniform distribution as the leader's prior distribution over all hypotheses given the follower's initial state. Notice that one can compute p^t recursively using the Bayes rule. Fig. 2 shows the time history of the ℓ_1-norm distance between p^t and $p^\star$, and compare the results where the leader uses, instead of the inputs optimal for optimization (18), independent and identically distributed input sampled from the uniform distribution over $\mathbb{U}$. The results in Fig. 2 show that, when compared with uniformly random inputs, the proposed method provide leader inputs that reduce the difference between p^t and $p^\star$ by orders of magnitudes.

V. Conclusion

We formulated an inverse learning problem in a Stackelberg trajectory game, where the leader is inferring the type of the follower's cost function by observing its trajectories. We proposed an active inverse learning method to accelerate the leader inference by making the follower's trajectories under different hypotheses as different as possible. This method accelerates the convergence of the probability of different hypotheses by orders of magnitude when compared against random inputs.

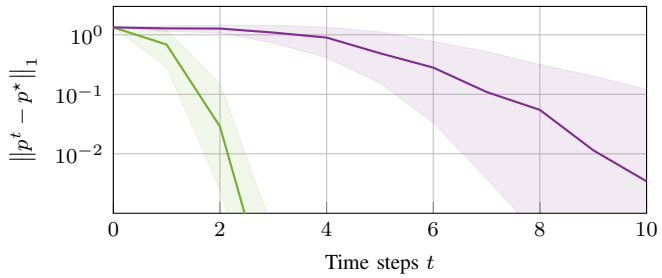
However, the current work still has limitations. For example, it considers neither nonlinear dynamics nor constraints for input limits and collision avoidance in the players' trajectory optimization. In addition, it ignores the possibility

$$
\begin{aligned}
&\underset{s,u^{\mathrm{L}}_{0:\tau-1}}{\text{minimize}} \quad s - \textstyle\sum_{(i,j)\in\mathbb{D}} \sum_{t=0}^{\tau} \left\|\xi^i_t - \xi^j_t\right\|^2_{(\Lambda^i_t)^{-1}+(\Lambda^j_t)^{-1}} + g(u_{0:\tau-1}) \\
&\text{subject to} \quad \eta^{\mathrm{L}}_{t+1} = A^{\mathrm{L}}\eta^{\mathrm{L}}_t + B^{\mathrm{L}}u^{\mathrm{L}}_t,\ \eta^{\mathrm{L}}_0 = \hat{x}^{\mathrm{L}}_0, \\
&\quad\ q^i_t = (E^i_t)^\top q^i_{t+1} - Q^i M^i \eta_t,\ q^i_\tau = -Q^i M^i \eta_\tau,\ \xi^i_{t+1} = E^i_t \xi^i_t - F^i_t q^i_{t+1},\ \xi^i_0 = \hat{x}^{\mathrm{F}}_0, \\
&\quad\ \textstyle\sum_{(k,l)\in\mathbb{D}\setminus\{(i,j)\}} \sum_{t=1}^{\tau} \left\|\xi^k_t - \xi^l_t\right\|^2_{(\Lambda^k)_t^{-1}+(\Lambda^l)_t^{-1}} \le s,\ \forall t \in [0,\tau-1],\ i \in [1,d],\ (i,j) \in \mathbb{D}.
\end{aligned}
\tag{18}
$$

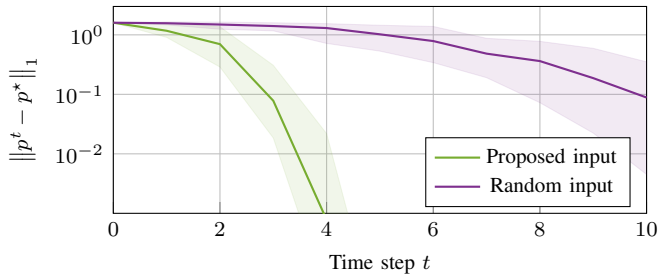


(a) $d = 3$ (b) $d = 5$

Fig. 1: The position trajectories of the leader's rovers, where different rover's trajectories are marked by different colors.



(a) $d = 3$.



(b) $d = 5$.

Fig. 2: The comparison of the convergence of p^t when the leader obtains its input by solving optimization 18 versus sampling uniformly in set $\mathbb{U}$. The solid lines show the median over 100 simulations, while the boundaries of the colored areas mark the corresponding first and third quartiles.

of deceptive actions of the follower. In future work, we plan to address these limitations and develop information-gathering strategies in general Bayesian games.

APPENDIX

Proof of Proposition 1

Given $y \in \mathbb{R}^{n_{\mathrm{F}}}$ and $t \in [0,\tau]$, let

$$
\begin{aligned}
&\psi_t(y, \mu_{t:\tau}, \Sigma_{t:\tau}) \\
&:= \textstyle\sum_{j=t}^{\tau} \mathbb{E}\left[\frac{1}{2}\left\|x^{\mathrm{F}}_j - M^{\mathrm{F}} x^{\mathrm{L}}_j\right\|^2_{Q^{\mathrm{F}}} | x^{\mathrm{F}}_t = y\right] \\
&+ \tfrac{1}{2}\textstyle\sum_{j=t}^{\tau-1}\left(\mathbb{E}\left[\left\|u^{\mathrm{F}}_j\right\|^2_{R^{\mathrm{F}}} | x^{\mathrm{F}}_t = y\right] - \log\det\Sigma_j\right).
\end{aligned}
\tag{20}
$$

where $x^{\mathrm{F}}_{j+1} = A^{\mathrm{F}}x^{\mathrm{F}}_j + B^{\mathrm{F}}u^{\mathrm{F}}_j$ and $u^{\mathrm{F}}_j \sim \mathcal{N}(\mu_j, \Sigma_j)$. Furthermore, let

$$
V_t(y) := \min_{\mu_{t:\tau-1}, \Sigma_{t:\tau-1}} \psi_t(y, \mu_{t:\tau}, \Sigma_{t:\tau}) \tag{21}
$$

for all $t \in [0,\tau]$. Then one can verify that the optimal value of optimization (4) is $V(\hat{x}^{\mathrm{F}}_0)$. In addition, we can show that $V_\tau(y) = \frac{1}{2}y^\top Q^{\mathrm{F}} y - \langle Q^{\mathrm{F}} M^{\mathrm{F}} x^{\mathrm{L}}_\tau, y\rangle + \frac{1}{2}\left\|M^{\mathrm{F}}x^{\mathrm{L}}_\tau\right\|^2_{Q^{\mathrm{F}}}$, *i.e.*, $V_\tau(y)$ is a quadratic function of y. Suppose that $V_{t+1}(y)$ is a quadratic function of y, *i.e.*, there exists $P^{\mathrm{F}}_{t+1} \in \mathbb{R}^{n_{\mathrm{F}}\times n_{\mathrm{F}}}$, $q^{\mathrm{F}}_{t+1} \in \mathbb{R}^{\mathrm{F}}$, and $\nu^{\mathrm{F}}_{t+1} \in \mathbb{R}$, such that

$$
V_{t+1}(y) = \tfrac{1}{2}y^\top P^{\mathrm{F}}_{t+1} y + \langle q^{\mathrm{F}}_{t+1}, y\rangle + \nu^{\mathrm{F}}_{t+1}. \tag{22}
$$

Then (21) and the principle of dynamic programming together imply that

$$
\begin{aligned}
&V_t(y) \\
&= \min_{\mu_t, \Sigma_t} \mathbb{E}\left[\tfrac{1}{2}\left\|u^{\mathrm{F}}_t\right\|^2_{R^{\mathrm{F}}} + V_{t+1}(A^{\mathrm{F}}x^{\mathrm{F}}_t + B^{\mathrm{F}}u^{\mathrm{F}}_t + w^{\mathrm{F}}_t) | x^{\mathrm{F}}_t = y\right] \\
&+ \tfrac{1}{2}\left\|y - M^{\mathrm{F}}x^{\mathrm{F}}_t\right\|^2_{Q^{\mathrm{F}}} - \tfrac{1}{2}\log\det\Sigma_t
\end{aligned}
\tag{23}
$$

where $u^{\mathrm{F}}_t | x^{\mathrm{F}}_t \sim \mathcal{N}(\mu_t, \Sigma_t)$. Observe that

$$
\begin{aligned}
&\mathbb{E}[\left\|u^{\mathrm{F}}_t\right\|^2_{R^{\mathrm{F}}} | x^{\mathrm{F}}_t = y] \\
&= \mu_t^\top R^{\mathrm{F}}\mu_t + \mathbb{E}[\mathrm{tr}((u^{\mathrm{F}}_t - \mu_t)(u^{\mathrm{F}}_t - \mu_t)^\top R^{\mathrm{F}}) | x^{\mathrm{F}}_t = y] \\
&= \mu_t^\top R^{\mathrm{F}}\mu_t + \mathrm{tr}(\Sigma_t R^{\mathrm{F}}).
\end{aligned}
\tag{24}
$$

In addition, by using (22) we can show that

$$
\begin{aligned}
&\mathbb{E}[V_{t+1}(A^{\mathrm{F}}x^{\mathrm{F}}_t + B^{\mathrm{F}}u^{\mathrm{F}}_t + w^{\mathrm{F}}_t) | x^{\mathrm{F}}_t = y] \\
&= \tfrac{1}{2}(A^{\mathrm{F}}y + B^{\mathrm{F}}\mu_t)^\top P^{\mathrm{F}}_{t+1}(A^{\mathrm{F}}y + B^{\mathrm{F}}\mu_t) \\
&+ \mathbb{E}[(A^{\mathrm{F}}x^{\mathrm{F}}_t + B^{\mathrm{F}}\mu_t)^\top P^{\mathrm{F}}_{t+1}B^{\mathrm{F}}(u^{\mathrm{F}}_t - \mu_t) | x^{\mathrm{F}}_t = y] \\
&+ \tfrac{1}{2}\mathbb{E}[(B^{\mathrm{F}}(u^{\mathrm{F}}_t - \mu_t))^\top P^{\mathrm{F}}_{t+1}B^{\mathrm{F}}(u_t - \mu_t) | x^{\mathrm{F}}_t = y] \\
&+ \langle q^{\mathrm{F}}_{t+1}, A^{\mathrm{F}}y + B^{\mathrm{F}}\mu_t\rangle + \frac{1}{2}\mathbb{E}[\mathrm{tr}(w^{\mathrm{F}}_t (w^{\mathrm{F}}_t)^\top P_{t+1})] + \nu^{\mathrm{F}}_{t+1} \\
&= \tfrac{1}{2}(A^{\mathrm{F}}y + B^{\mathrm{F}}\mu_t)^\top P^{\mathrm{F}}_{t+1}(A^{\mathrm{F}}y + B^{\mathrm{F}}\mu_t) + \tfrac{1}{2}\mathrm{tr}(\Omega^{\mathrm{F}}P_{t+1}) \\
&+ \tfrac{1}{2}\mathrm{tr}(\Sigma_t (B^{\mathrm{F}})^\top P^{\mathrm{F}}_{t+1}B^{\mathrm{F}}) + \langle q^{\mathrm{F}}_{t+1}, A^{\mathrm{F}}y + B^{\mathrm{F}}\mu_t\rangle + \nu^{\mathrm{F}}_{t+1}.
\end{aligned}
\tag{25}
$$

Substituting (25) and (24) into (23) gives

$$
\begin{aligned}
&V_t(y)\\
&= \tfrac{1}{2} y^\top (Q^{\mathrm{F}} + (A^{\mathrm{F}})^\top P^{\mathrm{F}}_{t+1} A^{\mathrm{F}}) y + \langle A^\top q^{\mathrm{F}}_{t+1} - Q^{\mathrm{F}} M^{\mathrm{F}} x^{\mathrm{L}}_t, y\rangle\\
&+ \tfrac{1}{2} \mu_t^\top (R^{\mathrm{F}} + (B^{\mathrm{F}})^\top P^{\mathrm{F}}_{t+1} B^{\mathrm{F}}) \mu_t + \tfrac{1}{2} (x^{\mathrm{L}}_t)^\top (M^{\mathrm{F}})^\top Q^{\mathrm{F}} M^{\mathrm{F}} x^{\mathrm{L}}_t\\
&+ \langle (B^{\mathrm{F}})^\top q^{\mathrm{F}}_{t+1} + (B^{\mathrm{F}})^\top P^{\mathrm{F}}_{t+1} A^{\mathrm{F}} y, \mu_t\rangle - \frac{1}{2} \log\det \Sigma_t\\
&+ \tfrac{1}{2} \operatorname{tr}(\Sigma_t R^{\mathrm{F}} + \Omega^{\mathrm{F}} P_{t+1} + (B^{\mathrm{F}})^\top P^{\mathrm{F}}_{t+1} B^{\mathrm{F}})) + \nu^{\mathrm{F}}_{t+1}.
\end{aligned} \tag{26}
$$

By setting the derivative of $V_t(y)$ with respect to μ_t and Σ_t to zero, we obtain (6). By substituting (6) into (26), we can show that $V_t(y) = \frac{1}{2} y^\top P^{\mathrm{F}}_t y + \langle q^{\mathrm{F}}_t, y\rangle + \nu^{\mathrm{F}}_t$, where Q^{F}_t and q^{F}_t satisfy (5).

Next, let $K_t = -\Sigma_t (B^{\mathrm{F}})^\top P^{\mathrm{F}}_{t+1} A^{\mathrm{F}}$ and $b_t = -\Sigma_t (B^{\mathrm{F}})^\top q^{\mathrm{F}}_{t+1}$. Then (6) implies that $u^{\mathrm{F}}_t | x^{\mathrm{F}}_t \sim \mathcal{N}(K_t x^{\mathrm{F}}_t + b_t, \Sigma_t)$. Since $x^{\mathrm{F}}_0 = \hat{x}^{\mathrm{F}}_0$, by using the results in [35, p. 91] we can show the following:

$$
\begin{bmatrix} x^{\mathrm{F}}_1 \\ u^{\mathrm{F}}_1 \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \xi_0 \\ K_0 \xi_0 + b_0 \end{bmatrix}, \begin{bmatrix} \Lambda_0 & K_0 \Lambda_0 \\ \Lambda_0 K_0^\top & \Sigma_0 + K_0^\top \Lambda_0 K_0 \end{bmatrix} \right).
$$

Therefore $x^{\mathrm{F}}_{t+1} = A^{\mathrm{F}} x^{\mathrm{F}}_t + B^{\mathrm{F}} u^{\mathrm{F}}_t + w^{\mathrm{F}}_t \sim \mathcal{N}(\xi_1, \Lambda_1)$, where ξ_1 and Σ_1 satisfy (7). By repeating similar steps for $t \in [2, \tau]$ we can show that (7) holds for all $t \in [0, \tau - 1]$, which completes the proof.

REFERENCES

[1] V. A. Shia, Y. Gao, R. Vasudevan, K. D. Campbell, T. Lin, F. Borrelli, and R. Bajcsy, "Semiautonomous vehicular control using driver modeling," *IEEE Trans. Intel. Transp. Syst.*, vol. 15, no. 6, pp. 2696–2709, 2014.

[2] N. Mehr, R. Horowitz, and A. D. Dragan, "Inferring and assisting with constraints in shared autonomy," in *Pro. IEEE Conf. Decision Control*. IEEE, 2016, pp. 6689–6696.

[3] N. Mehr, M. Wang, M. Bhatt, and M. Schwager, "Maximum-entropy multi-agent dynamic games: Forward and inverse solutions," *IEEE Trans. Robot.*, 2023.

[4] D. Sadigh, S. S. Sastry, S. A. Seshia, and A. Dragan, "Information gathering actions over human internal state," in *Proc. Int. Conf. Intel. Robots Syst.* IEEE, 2016, pp. 66–73.

[5] K. Waugh, B. D. Ziebart, and J. A. Bagnell, "Inverse correlated equilibrium for matrix games," *Adv. Neural Inform. Process. Syst.*, 2010.

[6] V. Kuleshov and O. Schrijvers, "Inverse game theory: Learning utilities in succinct games," in *Int. Conf. Web Internet Econ.* Springer, 2015, pp. 413–427.

[7] D. Bertsimas, V. Gupta, and I. C. Paschalidis, "Data-driven estimation in equilibrium using inverse optimization," *Math. Prog.*, vol. 153, no. 2, pp. 595–633, 2015.

[8] L. Peters, D. Fridovich-Keil, V. R. Royo, C. J. Tomlin, and C. Stachniss, "Inferring objectives in continuous dynamic games from noise-corrupted partial state observations," in *Robotics: Sci. and Syst. 2021*, 2021.

[9] T. L. Molloy, J. I. Charaja, S. Hohmann, and T. Perez, "Inverse optimal control and inverse noncooperative dynamic game theory," 2022.

[10] Y. Yu, J. Salfity, D. Fridovich-Keil, and U. Topcu, "Inverse matrix games with unique quantal response equilibrium," *IEEE Control Syst. Lett.*, vol. 7, pp. 643–648, 2022.

[11] J. Li, C.-Y. Chiu, L. Peters, S. Sojoudi, C. Tomlin, and D. Fridovich-Keil, "Cost inference for feedback dynamic games from noisy partial state observations and incomplete trajectories," in *Proc. Int. Conf. Auton. Agents and Multiagent Syst.*, 2023, pp. 1062–1070.

[12] C. K. Ling, F. Fang, and J. Z. Kolter, "What game are we playing? end-to-end learning in normal and extensive form games," in *Proc. Int. Joint Conf. Artif. Intell.*, 2018, pp. 396–402.

[13] S. Chen, Y. Yu, D. Fridovich-Keil, and U. Topcu, "Soft-bellman equilibrium in affine markov games: Forward solutions and inverse learning," *arXiv preprint arXiv:2304.00163*, 2023.

[14] T. Başar, "Affine incentive schemes for stochastic systems with dynamic information," *SIAM J. Control Optim.*, vol. 22, no. 2, pp. 199–210, 1984.

[15] N. Nisan and A. Ronen, "Algorithmic mechanism design," *Games Econ. Behav*, vol. 35, no. 1, pp. 359–379, 2015.

[16] M. Lopes, F. Melo, and L. Montesano, "Active learning for reward estimation in inverse reinforcement learning," in *Proc. Eur. Conf. Mach. Learn. Knowl. Discovery Databases*. Springer Berlin Heidelberg, 2009, pp. 31–46.

[17] R. Akrour, M. Schoenauer, and M. Sebag, "April: Active preference learning-based reinforcement learning," in *Proc. Eur. Conf. Mach. Learn. Knowl. Discovery Databases*. Springer, 2012, pp. 116–131.

[18] D. Sadigh, A. Dragan, S. Sastry, and S. Seshia, "Active preference-based learning of reward functions," in *Robotics: Science and Systems XIII*, July 2017.

[19] P. F. Christiano, J. Leike, T. Brown, M. Martic, S. Legg, and D. Amodei, "Deep reinforcement learning from human preferences," *Adv. Neural Inform. Process. Syst.*, vol. 30, 2017.

[20] C. Daniel, M. Viering, J. Metz, O. Kroemer, and J. Peters, "Active reward learning." in *Robot.: Sci. Syst.*, vol. 98, 2014.

[21] E. Bıyık and D. Sadigh, "Batch active preference-based learning of reward functions," in *Proc. Conf. Robot Learn.* PMLR, 2018, pp. 519–528.

[22] E. Bıyık, D. A. Lazar, D. Sadigh, and R. Pedarsani, "The green choice: Learning and influencing human decisions on shared roads," in *Proc. IEEE Conf. Decision Control*. IEEE, 2019, pp. 347–354.

[23] M. Palan, G. Shevchuk, N. Charles Landolfi, and D. Sadigh, "Learning reward functions by integrating human demonstrations and preferences," in *Robot.: Sci. Syst.*, 2019.

[24] S. M. Katz, A.-C. Le Bihan, and M. J. Kochenderfer, "Learning an urban air mobility encounter model from expert preferences," in *Proc. IEEE/AIAA Digit. Avionics Syst. Conf.* IEEE, 2019, pp. 1–8.

[25] C. Basu, E. Bıyık, Z. He, M. Singhal, and D. Sadigh, "Active learning of reward dynamics from hierarchical queries," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.* IEEE, 2019, pp. 120–127.

[26] R. Cohn, E. Durfee, and S. Singh, "Comparing action-query strategies in semi-autonomous agents," in *Proc. AAAI Conf. on Artif. Intell.*, vol. 25, no. 1, 2011, pp. 1102–1107.

[27] C. Daniel, O. Kroemer, M. Viering, J. Metz, and J. Peters, "Active reward learning with a novel acquisition function," *Autonomous Robots*, vol. 39, pp. 389–405, 2015.

[28] Y. Cui and S. Niekum, "Active reward learning from critiques," in *Proc. IEEE Int. Conf. Robot. Automat.* IEEE, 2018, pp. 6907–6914.

[29] E. Bıyık, M. Palan, N. C. Landolfi, D. P. Losey, and D. Sadigh, "Asking easy questions: A user-friendly approach to active reward learning," in *Conference on Robot Learning*. PMLR, 2020, pp. 1177–1190.

[30] V. Myers, E. Bıyık, N. Anari, and D. Sadigh, "Learning multimodal rewards from rankings," in *Proc. Conf. Robot Learn.* PMLR, 2022, pp. 342–352.

[31] R. Horst and N. V. Thoai, "Dc programming: overview," *J. Optim. Theory Appl.*, vol. 103, pp. 1–43, 1999.

[32] T. Lipp and S. Boyd, "Variations and extension of the convex–concave procedure," *Optim. Eng.*, vol. 17, pp. 263–287, 2016.

[33] S. Dempe and A. Zemkoho, "Bilevel optimization," in *Springer Optimization and Its Applications*. Springer, 2020, vol. 161.

[34] G. Lanckriet and B. K. Sriperumbudur, "On the convergence of the concave-convex procedure," *Adv. Neural Inform. Process. Syst.*, vol. 22, 2009.

[35] C. M. Bishop and N. M. Nasrabadi, *Pattern recognition and machine learning*. Springer, 2006, vol. 4, no. 4.