Risk-sensitive Inverse Reinforcement Learning via Coherent Risk Models

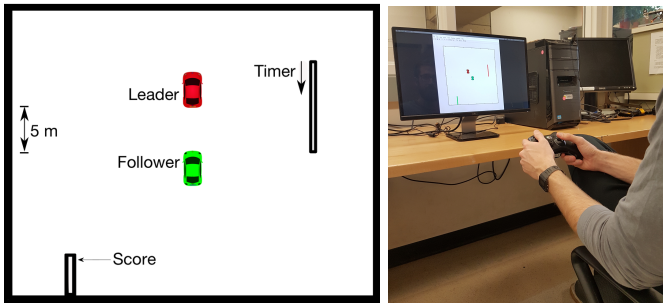
Anirudha Majumdar†, Sumeet Singh†, Ajay Mandlekar*, and Marco Pavone†
†Department of Aeronautics and Astronautics, *Electrical Engineering
Stanford University, Stanford, CA 94305
Email: {anirudha,ssingh19,amandlek,pavone}@stanford.edu

***Abstract*—The literature on Inverse Reinforcement Learning (IRL) typically assumes that humans take actions in order to minimize the expected value of a cost function, i.e., that humans are *risk neutral*. Yet, in practice, humans are often far from being risk neutral. To fill this gap, the objective of this paper is to devise a framework for *risk-sensitive* IRL in order to explicitly account for an expert's risk sensitivity. To this end, we propose a flexible class of models based on *coherent risk metrics*, which allow us to capture an entire spectrum of risk preferences from risk-neutral to worst-case. We propose efficient algorithms based on Linear Programming for inferring an expert's underlying risk metric and cost function for a rich class of static and dynamic decision-making settings. The resulting approach is demonstrated on a simulated driving game with ten human participants. Our method is able to infer and mimic a wide range of qualitatively different driving styles from highly risk-averse to risk-neutral in a data-efficient manner. Moreover, comparisons of the Risk-Sensitive (RS) IRL approach with a risk-neutral model show that the RS-IRL framework more accurately captures observed participant behavior both qualitatively and quantitatively.**

I. Introduction

Imagine a world where robots and humans coexist and work seamlessly together. In order to realize this vision, robots must be able to (1) accurately predict the actions of humans in their environment, (2) quickly learn the preferences of human agents in their proximity and act accordingly, and (3) learn how to accomplish new tasks from human demonstrations. Inverse Reinforcement Learning (IRL) [41, 32, 2, 29, 38, 50, 16] is a powerful and flexible framework for tackling these challenges and has been previously used for tasks such as modeling and mimicking human driver behavior [1, 28, 43], pedestrian trajectory prediction [51, 31], and legged robot locomotion [52, 27, 35]. The underlying assumption behind IRL is that humans act optimally with respect to an (unknown) cost function. The goal of IRL is then to infer this cost function from observed actions of the human. By learning the human's underlying preferences (in contrast to, e.g., directly learning a policy for a given task), IRL allows one to generalize one's predictions to novel scenarios and environments.

The prevalent modeling assumption made by existing IRL techniques is that humans take actions in order to minimize the *expected value* of a cost function. This modeling assumption corresponds to the *expected utility/cost (EU)* theory in economics [47]. Despite the historical prominence of EU theory in modeling human behavior, a large body of literature from the theory of human decision making strongly suggests that humans behave in a manner that is *inconsistent* with an EU model. An elegant illustration of this is the *Ellsberg paradox* [15]. Imagine an urn (Urn 1) containing 50 red and 50 black balls. Urn 2 also contains 100 red and black balls, but the relative composition of colors is unknown. Suppose that there is a payoff of \$10 if a red ball is drawn (and no payoff for black). In human experiments, subjects display an overwhelming preference towards having a ball drawn from Urn 1. However, now suppose the subject is told that a black ball has \$10 payoff (and no payoff for red). Humans *still* prefer to draw from Urn 1. But, this is a paradox since choosing to draw from Urn 1 in the first case (payoff for red) indicates that the proportion of red in Urn 1 is higher than in Urn 2, while choosing Urn 1 in the second case (payoff for black) indicates a lower proportion of red in Urn 1 than in Urn 2.



(a) Visualization of screen seen by human. (b) Joystick setup.

Fig. 1: The simulated driving game considered in this paper. The human controls the follower car using a joystick and must follow the leader (an "erratic driver") as closely as possible without colliding. We observed a wide range of behaviors from participants reflecting varying attitudes towards risk.

Intuitively, there are two main limitations of EU theory: (1) the standard model assumes that humans are *risk neutral* with respect to their utility function, and (2) it assumes that humans make no distinction between scenarios in which the probabilities of outcomes are known (e.g., Urn 1 in the Ellsberg paradox) and ones in which the outcomes are unknown (e.g., Urn 2). The known and unknown probability scenarios are referred to as *risky* and *ambiguous* respectively in the decision theory literature. While EU theory does allow some consideration of risk, e.g., via concave utility functions, experimental results demonstrate that it is not a good model for human behavior in risky scenarios [26]. This has prompted work on various non-EU theories (see e.g., [5, 15, 26, 9]). Further, one way to interpret the Ellsberg paradox is that humans are not only risk averse, but are also *ambiguity averse* – an observation that has sparked an alternative set of literature in decision theory on "ambiguity-averse" modeling; see, e.g., the recent review [18]. The assumptions made by EU theory thus represent significant restrictions from a modeling perspective in an IRL context since a human expert is likely to be risk and ambiguity averse, especially in safety critical applications such as driving where outcomes are inherently ambiguous and can possibly incur very high cost.

The key insight of this paper is to address these challenges by modeling humans as evaluating costs according to an (unknown) *risk metric*. A risk metric is a function that maps an uncertain cost to a real number (the expected value is thus a particular risk metric and corresponds to risk neutrality). In particular, we will consider the class of *coherent risk metrics* (CRMs) [7, 44, 42]. These metrics were introduced in the operations research literature and have played an influential role within the modern theory of risk in finance [40, 4, 3, 39]. This theory has also recently been adopted for risk-sensitive (RS) Model Predictive Control and decision making [12, 13], and autonomous exploration [8].

Coherent risk metrics enjoy a number of advantages over EU theory in the context of IRL. First, they capture an entire spectrum of risk assessments from risk-neutral to worst-case and thus offer a significant degree of modeling flexibility (note that EU theory is a special case of a coherent risk model). Second, they capture risk sensitivity in an *axiomatically justified* manner; they formally capture a number of intuitive notions that one would expect any risk metric to satisfy (ref. Section II-B). Third, a representation theorem for CRMs (Section II-B) implies that they can be interpreted as computing the expected value of a cost function in a worst-case sense over a *set* of probability distributions (referred to as the *risk envelope*). Thus, CRMs capture both risk and ambiguity aversion within the same modeling framework since the risk envelope can be interpreted as capturing uncertainty about the underlying probability distribution that generates outcomes in the world. Finally, they are tractable from a computational perspective; the representation theorem allows us to solve both the inverse and forward problems in a computationally tractable manner for a rich class of static and dynamic decision-making settings.

Statement of contributions: To our knowledge, the results in this paper constitute the first attempt to explicitly take into account risk sensitivity in the context of IRL under *general* axiomatically justified risk models that jointly capture risk and ambiguity. To this end, this paper makes three primary contributions. First, we propose a flexible modeling framework for capturing risk sensitivity in experts by assuming that the expert acts according to a CRM. This framework allows us to capture an entire spectrum of risk assessments from risk-neutral to worst-case. Second, we develop efficient algorithms based on Linear Programming (LP) for inferring an expert's underlying risk metric for a broad range of static (Section III) and dynamic (Section IV) decision-making settings. We consider cases where the expert's underlying risk metric is unknown but the cost function is known, and also the more general case where both are unknown. Third, we demonstrate our approach on a simulated driving game (visualized in Figure 1) and present results on ten human participants (Section V). We show that our approach is able to infer and mimic qualitatively different driving styles ranging from highly risk-averse to risk-neutral using only a minute of training data from each participant. We also compare the predictions made by our risk-sensitive IRL (RS-IRL) approach with one that models the expert using EU theory and demonstrate that the RS-IRL framework more accurately captures observed participant behavior both qualitatively and quantitatively.

Related Work: Restricted versions of the problems considered here have been studied before. In particular, there is a large body of work on RS decision making. For instance, in [24] the authors leverage the exponential (or entropic) risk. This has historically been a very popular technique for parameterizing risk-attitudes in decision theory but suffers from the usual drawbacks of the EU framework as elucidated in [37]. Other RS Markov Decision Process (MDP) formulations include Markowitz-inspired mean-variance [17] and percentile-based risk measures (e.g., Conditional value-at-risk (CVaR) [10]). This has driven research in the design of learning-based solution algorithms, i.e., RS reinforcement learning [30, 46, 36, 45]. Ambiguity in MDPs is also well studied via the robust MDP framework, see e.g., [33, 49], as well as [34, 13] where the duality between risk and ambiguity as a result of CRMs is exploited. The key difference between this literature and the present work is that we consider the *inverse* reinforcement learning problem.

Results in the RS-IRL setting are more limited and have largely been pursued in the *neuroeconomics* literature [21]. For example, [25] performed Functional Magnetic Resonance Imaging (FMRI) studies of humans making decisions in risky and ambiguous settings and modeled risk and ambiguity aversion using parametric utility and weighted probability models. In a similar vein, [45] models risk aversion using utility based shortfalls and presents FMRI studies on humans performing a sequential investment task. While this literature may be interpreted in the context of IRL, the models used to predict risk and ambiguity aversion are quite limited. For example, *shortfall* is fixed as the risk metric in [45] while estimating parameters in a utility model. In contrast, the class of risk metrics we consider are significantly more flexible.

II. Problem Formulation

A. Dynamics

Consider the following discrete-time dynamical system:

$$x_{k+1} = f(x_k, u_k, w_k), \tag{1}$$

where k is the time index, $x_k \in \mathbb{R}^n$ is the state, $u_k \in \mathbb{R}^m$ is the control input, and $w_k \in \mathcal{W}$ is the disturbance. The control input is assumed to be bounded component-wise: $u_k \in \mathcal{U} := \{u : u^- \leq u \leq u^+\}$. We take $\mathcal{W}$ to be a finite set $\{w^{[1]}, \ldots, w^{[L]}\}$ with probability mass function (pmf) $p := [p(1), p(2), \ldots, p(L)]$, where $\sum_{i=1}^{L} p(i) = 1$ and $p(i) > 0, \forall i \in \{1, \ldots, L\}$. The time-sampling of the disturbance w_k will be discussed in Section IV. We assume that we are given demonstrations from an *expert* in the form of sequences of state-control pairs $\{(x_k^*, u_k^*)\}_k$ and that the expert has knowledge of the underlying dynamics (1) but does *not* necessarily have access to the disturbance set pmf p.

B. Model of the Expert

We model the expert as a *risk-aware* decision-making agent acting according to a *coherent risk metric* (defined formally below). We refer to such a model as a *coherent risk model*.

We assume that the expert has a cost function $C(x_k, u_k)$ that captures his/her preferences about outcomes. Let Z denote the cumulative cost accrued by the agent over a horizon N:

$$Z := \sum_{k=0}^{N} C(x_k, u_k). \tag{2}$$

Note that since the process $\{x_k\}$ is stochastic, Z is a random variable adapted to the sequence $\{x_k\}$. A *risk metric* is a

function $\rho(Z)$ that maps this uncertain cost to a real number. We will assume that the expert is assessing risks according to a *coherent risk metric*, defined as follows.

Definition 1 (Coherent Risk Metrics)**.** *Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space and let $\mathcal{Z}$ be the space of random variables on Ω. A coherent risk metric (CRM) is a mapping $\rho : \mathcal{Z} \to \mathbb{R}$ that obeys the following four axioms. For all $Z, Z' \in \mathcal{Z}$:*

A1. Monotonicity: $Z \leq Z' \Rightarrow \rho(Z) \leq \rho(Z')$.
A2. Translation invariance: $\forall a \in \mathbb{R}$, $\rho(Z+a) = \rho(Z)+a$.
A3. Positive homogeneity: $\forall \lambda \geq 0$, $\rho(\lambda Z) = \lambda\rho(Z)$.
A4. Subadditivity: $\rho(Z + Z') \leq \rho(Z) + \rho(Z')$.

These axioms were originally proposed in [7] to ensure the "rationality" of risk assessments. For example, A1 states that if a random cost Z is less than or equal to a random cost Z' *regardless of the disturbance realizations*, then Z must be considered less risky (one may think of the different random costs arising from different control policies). A4 reflects the intuition that a risk-averse agent should prefer to *diversify*. We refer the reader to [7] for a thorough justification of these axioms. An important characterization of CRMs is provided by the following representation theorem.

Theorem 1 (Representation Theorem for Coherent Risk Metrics [7])**.** *Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space, where Ω is a finite set with cardinality $|\Omega|$, $\mathcal{F}$ is the $\sigma-$algebra over subsets (i.e., $\mathcal{F} = 2^\Omega$), probabilities are assigned according to $\mathbb{P} = (p(1), p(2), \ldots, p(|\Omega|))$, and $\mathcal{Z}$ is the space of random variables on Ω. Denote by $\mathcal{C}$ the set of valid probability densities:*

$$\mathcal{C} := \left\{ \zeta \in \mathbb{R}^{|\Omega|} \mid \sum_{i=1}^{|\Omega|} p(i)\zeta(i) = 1, \zeta \geq 0 \right\}. \tag{3}$$

Define $q_\zeta \in \mathbb{R}^{|\Omega|}$ where $q_\zeta(i) = p(i)\zeta(i)$, $i = 1, \ldots, |\Omega|$. A risk metric $\rho : \mathcal{Z} \to \mathbb{R}$ with respect to the space $(\Omega, \mathcal{F}, \mathbb{P})$ is a CRM if and only if there exists a compact convex set $\mathcal{B} \subset \mathcal{C}$ such that for any $Z \in \mathcal{Z}$:

$$\rho(Z) = \max_{\zeta \in \mathcal{B}} \mathbb{E}_{q_\zeta}[Z] = \max_{\zeta \in \mathcal{B}} \sum_{i=1}^{|\Omega|} p(i)\zeta(i)Z(i). \tag{4}$$

This theorem is important for two reasons. Conceptually, it gives us an interpretation of CRMs as computing the worst-case expectation of the cost function over a set of densities $\mathcal{B}$ (referred to as the risk envelope). Coherent risks thus allow us to consider risk and ambiguity (ref. Section I) in a unified framework since one may interpret an agent acting according to a coherent risk model as being *uncertain about the underlying probability density*. Second, it provides us with an algorithmic handle over CRMs and will form the basis of our approach to measuring experts' risk preferences.

In this work, we will take the risk envelope $\mathcal{B}$ to be a polytope. We refer to such risk metrics as *polytopic risk metrics*, which were also considered in [14]. By absorbing the density ζ into the pmf p, we can represent (without loss of generality) a polytopic risk metric as:

$$\rho(Z) = \max_{p \in \mathcal{P}} \mathbb{E}_p[Z], \tag{5}$$

where $\mathcal{P}$ is a polytopic subset of the probability simplex:

$$\mathcal{P} = \left\{ p \in \Delta^{|\Omega|} \mid A_{\text{ineq}} p \leq b_{\text{ineq}} \right\}, \tag{6}$$

where $\Delta^{|\Omega|} := \{p \in \mathbb{R}^{|\Omega|} \mid \sum_{i=1}^{|\Omega|} p(i) = 1,\ p \geq 0\}$. Polytopic risk metrics constitute a rich class of risk metrics, encompassing a spectrum ranging from risk neutrality ($\mathcal{P} = \{p\}$) to worst-case assessments ($\mathcal{P} = \Delta^{|\Omega|}$). Examples include CVaR, mean absolute semi-deviation, spectral risk measures, optimized certainty equivalent, and the distributionally robust risk [12]. We further note that the *ambiguity* interpretation of CRMs is reminiscent of Gilboa & Schmeidler's Minmax EU model for ambiguity-aversion [19] which was shown to outperform various competing models in [22] for single-stage decision problems, albeit with more restrictions on the set $\mathcal{B}$.

Goal: Given demonstrations from the expert in the form of state-control trajectories, our goal will be to *conservatively approximate* the expert's risk preferences by finding an *outer approximation* of the risk envelope $\mathcal{P}$.

III. Risk-sensitive IRL: Single Decision Period

In this section we consider the single step decision problem, i.e., $N = 0$ in equation (2). Thus, the probability space $(\Omega, \mathcal{F}, \mathbb{P})$ is simply $(\mathcal{W}, 2^\mathcal{W}, p)$.

A. Known Cost Function

We first consider the static decision-making setting where the expert's cost function is known but the risk metric is unknown. A coherent risk model then implies that the expert is solving the following optimization problem at state x in order to compute an optimal action:

$$\tau^* := \min_{u \in \mathcal{U}} \ \rho(C(x,u)) = \min_{u \in \mathcal{U}} \max_{p \in \mathcal{P}} \ \mathbb{E}_p[C(x,u)] \tag{7}$$

$$:= \min_{u \in \mathcal{U}} \max_{p \in \mathcal{P}} \ g(x,u)^T p, \tag{8}$$

where $\rho(\cdot)$ is a CRM with respect to the space $(\mathcal{W}, 2^\mathcal{W}, p)$ (i.e., $\mathcal{P} \subseteq \Delta^L$). In the last equation, $g(x,u)(j)$ is the cost when the disturbance $w^{[j]} \in \mathcal{W}$ is realized. Since the inner maximization problem is linear in p, the optimal value is achieved at a vertex of the polytope $\mathcal{P}$. Denoting the set of vertices of $\mathcal{P}$ as $\text{vert}(P) = \{v_i\}_{i \in \{1,\ldots,N_V\}}$, we can thus rewrite problem (7) above as follows:

$$\begin{aligned} \min_{u \in \mathcal{U}, \tau} \quad & \tau \\ s.t. \quad & \tau \geq g(x,u)^T v_i, \quad i \in \{1, \ldots, N_V\} \end{aligned} \tag{9}$$

If the cost function $C(\cdot,\cdot)$ is convex in the control input u, the resulting optimization problem is convex. Given a dataset $\mathcal{D} = \{(x^{*,d}, u^{*,d})\}_{d=1}^D$ of state-control pairs of the expert taking action $u^{*,d}$ at state $x^{*,d}$, our goal is to deduce a *conservative* approximation (i.e., an outer approximation) $\mathcal{P}_o$ of $\mathcal{P}$ from the given data. The key idea of our technical approach is to examine the Karush-Kuhn-Tucker (KKT) conditions for Problem (9). The use of KKT conditions for Inverse Optimal Control is a technique also adopted in [16]. The KKT conditions are necessary for optimality in general and are also sufficient in the case of convex problems. We can thus use the KKT conditions along with the dataset $\mathcal{D}$ to *constrain the constraints* of Problem (9). In other words, the KKT conditions will allow us to constrain where the vertices of $\mathcal{P}$ must lie in order to be consistent with the fact that the state-control pairs represent optimal solutions to Problem (9). Importantly, we will *not* assume access to the number of vertices N_V of $\mathcal{P}$.

Let (x^*, u^*) be an optimal state-control pair and let $\mathcal{J}^+$ and $\mathcal{J}^-$ denote the sets of components of the control input u^*

that are saturated above and below respectively (i.e., $u(j) = u^+(j), \forall j \in \mathcal{J}^+$ and $u(j) = u^-(j), \forall j \in \mathcal{J}^-$).

Theorem 2. *Consider the following optimization problem:*

$$\begin{aligned} \max_{\substack{v\in\Delta^L \\ \sigma_+,\sigma_-\geq 0}} \quad & g(x^*,u^*)^T v \qquad (10)\\ s.t. \quad & 0 = \nabla_{u(j)} g(x,u)^T v\Big|_{x^*,u^*} + \sigma_+(j), \forall j \in \mathcal{J}^+ \\ & 0 = \nabla_{u(j)} g(x,u)^T v\Big|_{x^*,u^*} - \sigma_-(j), \forall j \in \mathcal{J}^-. \\ & 0 = \nabla_{u(j)} g(x,u)^T v\Big|_{x^*,u^*}, \forall j \notin \mathcal{J}^+, j \notin \mathcal{J}^-. \end{aligned}$$

Denote the optimal value of this problem by τ' and define the halfspace:

$$\mathcal{H}_{(x^*,u^*)} := \{v \in \mathbb{R}^L \mid \tau' \geq g(x^*,u^*)^T v\}. \qquad (11)$$

Then, the risk envelope $\mathcal{P}$ satisfies $\mathcal{P} \subset (\mathcal{H}_{(x^*,u^*)} \cap \Delta^L)$.

Proof: The KKT conditions for Problem (9) are:

$$1 = \sum_{i=1}^{N_V} \lambda_i, \qquad (12)$$

$$0 = \lambda_i [g(x^*,u^*)^T v_i - \tau],\ i = 1,\dots,N_V, \qquad (13)$$

For $j = 1,\dots,m$:

$$0 = \sigma_+(j) - \sigma_-(j) + \sum_{i=1}^{N_V} \lambda_i\ \nabla_{u(j)} g(x,u)^T v_i\Big|_{x^*,u^*}, \qquad (14)$$

$$0 = \sigma_+(j)[u^*(j) - u^+(j)],\ 0 = \sigma_-(j)[u^-(j) - u^*(j)], \qquad (15)$$

where $\lambda_i, \sigma_+(j), \sigma_-(j) \geq 0$ are multipliers. Now, suppose there are multiple optimal vertices $\{v_i\}_{i\in\mathcal{I}}$ for Problem (9) in the sense that $\tau^* = g(x^*,u^*)^T v_i,\ \forall i \in \mathcal{I}$. Defining $\bar{v} := \sum_{i\in\mathcal{I}} \lambda_i v_i$, we see that $\bar{v}$ satisfies:

$$0 = \nabla_{u(j)} g(x^*,u^*(j))^T \bar{v} + \sigma_+(j) - \sigma_-(j), j = 1,\dots,m, \qquad (16)$$

and $\tau^* = g(x^*,u^*)^T \bar{v}$ since $\sum_{i\in\mathcal{I}} \lambda_i = 1$. Now, since $\bar{v}$ satisfies the constraints of Problem (10) (which are implied by the KKT conditions), it follows that $\tau' \geq \tau^*$. From problem (9), we see that $\tau' \geq \tau^* \geq g(x^*,u^*)^T v_i,\ \forall v_i \in \text{vert}(\mathcal{P})$ and thus $\mathcal{P} \subset (\mathcal{H}_{(x^*,u^*)} \cap \Delta^L)$. ∎

Problem (10) is a *Linear Program (LP)* and can thus be solved efficiently. For each demonstration $(x^{*,d}, u^{*,d}) \in \mathcal{D}$, Theorem 2 provides a halfspace constraint on the risk envelope $\mathcal{P}$. By aggregating these constraints, we obtain a *polytopic* outer approximation $\mathcal{P}_o$ of $\mathcal{P}$. This is summarized in Algorithm 1. Note that Algorithm 1 operates sequentially through the data $\mathcal{D}$ and is thus directly applicable in *online* settings.

Algorithm 1 Outer Approximate Risk Envelope

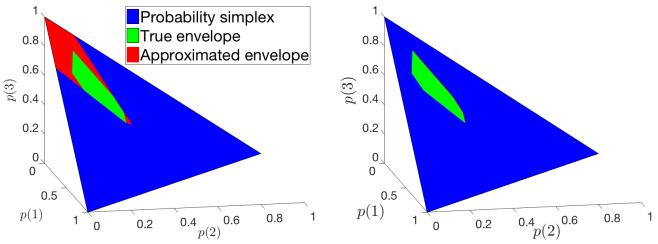
1: Initialize $\mathcal{P}_o = \Delta^L$
2: **for** $d = 1,\dots,D$ **do**
3: Solve Linear Program (10) with $(x^{*,d}, u^{*,d})$ to obtain a hyperplane $\mathcal{H}_{(x^{*,d},u^{*,d})}$
4: Update $\mathcal{P}_o \leftarrow \mathcal{P}_o \cap \mathcal{H}_{(x^{*,d},u^{*,d})}$
5: **end for**
6: Return $\mathcal{P}_o$

As we collect more half-space constraints in Algorithm 1, the constraint $v \in \Delta^L$ in Problem (10) above can be replaced by $v \in \mathcal{P}_o$, where $\mathcal{P}_o$ is the current outer approximation of the risk envelope. It is easily verified that the results of Theorem 2 still hold. This allows us to obtain a tighter (i.e., lower) upper bound τ' for τ^*, thus resulting in tighter halfspace constraints.

Once we have recovered an outer approximation $\mathcal{P}_o$ of $\mathcal{P}$, we can solve the "forward" problem (i.e., compute actions at a given state x) by solving the optimization problem (7) with $\mathcal{P}_o$ as the risk envelope.

1) Example: Linear-Quadratic System: As a simple illustrative example to gain intuition for the convergence properties of Algorithm 1, consider a linear dynamical system with multiplicative uncertainty of the form $f(x_k,u_k,w_k) = A(w_k)x_k + B(w_k)u_k$. We consider the one-step decision-making process with a quadratic cost on state and action: $Z := u_0^T R u_0 + x_1^T Q x_1$, where $x_1 = A(w_0)x_0 + B(w_0)u_0$. Here, $R \succ 0$ and $Q \succeq 0$. We consider a 10-dimensional state space with a 5-dimensional control input space. The number of realizations is taken to be $L = 3$ for ease of visualization. The L different $A(w_k)$ and $B(w_k)$ matrices corresponding to each realization are generated randomly by independently sampling elements of the matrices from the standard normal distribution $\mathcal{N}(0,1)$. The cost matrix Q is a randomly generated psd matrix and R is the identity. States x^* are drawn from $\mathcal{N}(0,1)$.

Figure 2 shows the outer approximations of the risk envelope obtained using Algorithm 1. We observe rapid convergence (approximately 20 sampled states x^*) of the outer approximations $\mathcal{P}_o$ (red) to the true risk envelope $\mathcal{P}$ (green).



(a) 5 data points (b) 20 data points

Fig. 2: Rapid convergence of the outer approximation of the risk envelope.

B. Unknown Cost Function

Now we consider the more general case where both the expert's cost function and risk metric are unknown. We parameterize our cost function as a linear combination of features in (x,u). Then, the expected value of the cost function w.r.t. $p \in \Delta^L$ can be written as $g(x,u)^T p$, where:

$$g(x,u)(j) = \sum_{h=1}^{H} c(h)\phi_{j,h}(x,u),\ j = 1,\dots,L, \qquad (17)$$

with nonnegative weights $c \in \mathbb{R}^H_{\geq 0}$. Since the solution of problem (7) solved by the expert is invariant to positive scalings of the cost function due to the positive homogeneity property of coherent risks (ref. Definition 1), we can assume without loss of generality that the feature weights sum to 1.

With this cost structure, we see that the KKT conditions derived in Section III-A involve *products* of the feature weights c and the vertices v_i of $\mathcal{P}$. Similarly, an analogous version of optimization problem (10) can be used to bound the optimal value. This problem again contains products of the unknown feature weights c and the probability vertex v. The key idea here is to introduce new decision variables z that

replace each product $v(j)c(h)$ by a new variable z_{jh} which allows us to re-write problem (10) as an LP in (z, σ_+, σ_-), with the addition of the following two simple constraints: $0 \leq z_{jh} \leq 1, \forall j, h, \quad \sum_{j,h} z_{jh} = 1$. In a manner analogous to Theorem 2, this optimization problem allows us to obtain bounding hyperplanes *in the space of product variables* z which can then be aggregated as in Algorithm 1. Denoting this polytope as $\mathcal{P}_z$, we can then proceed to solve the "forward" problem (i.e., computing actions at a given state x) by solving the following optimization problem:

$$\min_{u \in \mathcal{U}} \max_{z \in \mathcal{P}_z} \sum_{j,h} z_{jh} \phi_{j,h}(x, u). \tag{18}$$

This problem can be solved by enumerating the vertices of the polytope $\mathcal{P}_z$ in a manner similar to problem (9).

Remark 1. *While the above procedure operates in the space of product variables z and does not require explicitly recovering the cost function and risk envelope separately, it may nevertheless be useful to do so in order to obtain additional insights into the expert's decision-making process and generalize to novel scenarios. We have explored an approach for doing this, but will defer the results to an extended version of this work due to space limitations.*

IV. Risk-sensitive IRL: Multi-step case

We now consider the *dynamical system* given by (1) and generalize the one-step decision problem to the multi-step setting. We consider a model where the disturbance w_k is sampled every N time-steps and held constant in the interim. Such a model is quite general and more realistic in high-level decision-making settings than one where disturbances are sampled i.i.d. at every time step. A *scenario tree* for this model is sketched in Figure 3. We note that the results presented here are easily extended to *aperiodic* disturbances where the expert knows the timing of disturbances. We will consider the more general case of uncertain timing in future work.

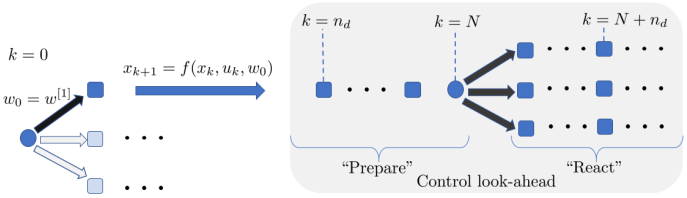


Fig. 3: Scenario tree as visualized at time $k = 0$. The disturbance is sampled every N steps. The control look-ahead has two phases: "prepare" and "react".

We model the expert as planning in a *receding horizon* manner by looking ahead for a finite horizon. Owing to the need to account for future disturbances, the multi-step finite-horizon problem is a search over control *policies* (i.e., the executed control inputs depend on which disturbance is realized). We decompose the expert's policy into two phases (shown in Figure 3), which we will refer to as "prepare" and "react". The "prepare" phase precedes each disturbance realization by $N - n_d$ steps while the "react" phase follows it for n_d steps. Intuitively, this model captures the idea that in the period preceding a disturbance (i.e., the "prepare" phase) the expert controls the system to a state from which he/she can recover well (in the "react" phase) once a disturbance is realized. Note that this model assumes that the expert is planning by taking into account only a single branching of the scenario tree (and not considering further branching in the future), which will lead to computationally tractable algorithms. Studies showing that humans have a relatively short look-ahead horizon in uncertain decision-making settings lend credence to such a model [11]. The dynamic model used in the multi-period control setting can be formally written as:

$$\begin{aligned} x_{k+1} &= f(x_k, u_k, w_k), \quad k \in [n_d, N + n_d] \\ w_k &= \begin{cases} w_0 & \text{for } k \in [n_d, N-1] \\ w_N & \text{for } k \in [N, N + n_d - 1] \end{cases} \end{aligned} \tag{19}$$

where w_0 is the disturbance realization from the sample preceding the current instantiation of the multi-period problem. Note that since the system is time-invariant, we state all equations in this section for the first "prepare" and "react" episode, with the understanding that the model repeats after N steps. Define a control policy at time k to be a function $\pi_k : \mathcal{X} \to \mathcal{U}$ and let $C_{k:l}(x, \pi(x)) := C(x_k, \pi_k(x_k)) + \ldots + C(x_l, \pi_l(x_l))$ for $k < l$. The multi-period optimization problem is then given as:

$$\begin{aligned} \min_{\pi_{n_d:N+n_d-1}} \; & C_{n_d:N}(x, \pi(x)) + \\ & \rho\Big(C_{N+1:N+n_d-1}(x, \pi(x)) + C(x_{N+n_d}, 0)\Big), \end{aligned} \tag{20}$$

where $\rho(\cdot)$ is a CRM with respect to the space $(\mathcal{W}, 2^{\mathcal{W}}, p)$ (i.e., the same as in the single-step case since we are planning over one disturbance sample). While the problem above is defined as an optimization over *Markov* control policies, we re-define the problem as an optimization over *history-dependent* policies. This additional flexibility will allow us to reformulate Problem (20) in a form similar to (9). Consider the following parameterization of *history-dependent* control policies. Let $j \in \{0, \ldots, L\}$ be the realized disturbance index at time $k = N$. Define $\overline{X}_k := x_k$, $\overline{U}_k := u_k$ for $k \in [n_d, N]$ and denote the (history-dependent) state and control at times $k \in [N + 1, N + n_d]$ as $\overline{X}_k^{[j]}$ and $\overline{U}_k^{[j]}$, for $j = 1, \ldots, L$. The system dynamics (19) can now be written as:

$$\begin{aligned} & \overline{X}_{k+1} = f(\overline{X}_k, \overline{U}_k, w_0), \; k \in [n_d, N-1] \\ & \text{for } j = 1, \ldots, L : \overline{X}_{N+1}^{[j]} = f(\overline{X}_N, \overline{U}_N, w^{[j]}), \text{ and} \\ & \overline{X}_{k+1}^{[j]} = f(\overline{X}_k^{[j]}, \overline{U}_k^{[j]}, w^{[j]}), \; k \in [N+1, N+n_d-1]. \end{aligned} \tag{21}$$

For ease of notation, define the vector $g(\overline{X}, \overline{U}) \in \mathbb{R}^L$ with j^{th} element

$$g(\overline{X}, \overline{U})(j) = C_{N+1:N+n_d-1}(\overline{X}^{[j]}, \overline{U}^{[j]}) + C(\overline{X}_{N+n_d}^{[j]}, 0).$$

Thus, $g(\overline{X}, \overline{U})(j)$ is the net cost accrued over the reaction phase when $w_N = w^{[j]}$. Extending the notation in (9) to this setting, problem (20) can be re-formulated as follows:

$$\begin{aligned} \min_{\substack{\tau, \overline{U}_k, k \in [n_d, N] \\ \overline{U}_k^{[j]}, j \in \{1, \ldots, L\} \\ k \in [N+1, N+n_d-1]}} \quad & C_{n_d:N}(\overline{X}, \overline{U}) + \tau \\ s.t. \; & \tau \geq g(\overline{X}, \overline{U})^T v_i, \forall v_i \in \text{vert}(\mathcal{P}) \\ & \overline{U}_k \in \mathcal{U}, \; k \in [n_d, N] \\ & \overline{U}_k^{[j]} \in \mathcal{U}, \; k \in [N+1, N+n_d-1] \\ & j \in \{1, \ldots, L\} \\ & \text{Dynamics (21).} \end{aligned} \tag{22}$$

Denote $(\overline{X}^*, \overline{U}^*)_{n_d:N}$ to be an optimal state and control pair preparation sequence, and $(\overline{X}^{[j]^*}, \overline{U}^{[j]^*})_{N+1:N+n_d-1}, j \in \{1, \ldots, L\}$, to be the set of optimal state and control pair reaction sequences. In similar fashion to the one-step optimal control problem, we will leverage the KKT conditions for problem (22) to constrain the risk envelope $\mathcal{P}$. Notice that since the problem is re-solved every N steps, the agent is assumed to execute the reaction sequence $\overline{U}^{[j]^*}_{N+1:N+n_d-1}$ in "open-loop" fashion having observed the disturbance $w_N = w^{[j]}$. Thus, the *observable* data from the expert corresponding to each instance of the problem above is the optimal preparation sequence and the optimal reaction sequence for the *realized* disturbance $w_N = w^{[j]}$. However, in order to deduce the risk-envelope $\mathcal{P}$, we will also require knowledge of the state and control pairs for the reaction phase for unrealized disturbances $w^{[l]} \neq w_N$. Accordingly, the data must be processed in two steps which depend upon whether the cost function $C(x, u)$ is known or not.

A. Known Cost Function

In the first step, we use the (observed) optimal state and control pair $(\overline{X}^*_N, \overline{U}^*_N)$ to compute the agent's reaction policies $\overline{U}^{[l]}_{N+1:N+n_d-1}$ for the *unrealized* disturbance branches by application of the Bellman principle which asserts that the reaction policies (i.e., tail policies for the overall finite horizon control problem) must be optimal for the tail subproblem (which is simply a deterministic optimal control problem). In the second step, we once again leverage the KKT conditions for problem (22) (omitted here for brevity) to yield the following maximization problem:

$$\tau' = \max_{\substack{v \in \Delta^L \\ \sigma_{+,k}, \sigma_{-,k}, \sigma^{[j]}_{+,k}, \sigma^{[j]}_{-,k} \geq 0}} g(\overline{X}^*, \overline{U}^*)^T v \tag{23}$$

$$s.t. \quad \text{KKT conditions} \tag{24}$$

where $\sigma_{-,k}, \sigma_{+,k}, \sigma^{[j]}_{-,k}, \sigma^{[j]}_{+,k} \geq 0,\ k \in [n_d, N + n_d - 1], j = 1, \ldots, L$ are the Lagrange multipliers (defined analogously to σ_+, σ_- in problem (10)). It follows then that $\tau' \geq \tau^*$ and thus the bounding hyperplane from this sequence of data is given by $\tau' \geq g(\overline{X}^*, \overline{U}^*)^T v$.

B. Unknown Cost Function

In the case where the cost function $C(x, u)$ is parameterized as a linear combination of features with unknown weights, i.e., similar to equation (17), one may adopt two methods. It can be shown that the KKT conditions (i.e., equation (24)) are linear in the products $v(j)c(h)$ and the weights $c(h)$. Thus one could solve problem (23) with respect to the product variables z_{jh} and the weights $c(h)$ in a manner analogous to Section III-B.

Alternatively, recall that once a disturbance has been realized, the control policy for the reaction phase is a solution to a deterministic optimal control problem. Thus, by leveraging standard IRL techniques [16], one can recover the cost function weights using the observed tail sequence only, i.e., $\overline{U}^{[j]*}_{N+1:N+n_d-1}$, and infer the contingent plans for the unrealized disturbance tails by solving a simple optimal control problem. One would then solve (23) as given. This is the approach adopted for the results in Section V and is summarized in Algorithm 2.

Algorithm 2 Outer Approximate Risk Envelope: Multi-step

1: Given: sequence of optimal state-control pairs $\{(x^*_k, u^*_k)\}_k$
2: Extract $(\overline{X}^{*,d}_p, \overline{U}^{*,d}_p)$ ("prepare") and $(\overline{X}^{*,d}_r, \overline{U}^{*,d}_r)$ ("react") phases for $d = 1, \ldots, D$ (where D is the total number of realized disturbances)
3: Infer cost function $C(x, u)$ from "react" phases using standard IRL techniques
4: Initialize $\mathcal{P}_o = \Delta^L$
5: **for** $d = 1, \ldots, D$ **do**
6: Compute "tail policies" for unrealized disturbances using $C(x, u)$ by solving deterministic optimal control problems
7: Solve Linear Program (23) to obtain hyperplane $\mathcal{H}_d$
8: Update $\mathcal{P}_o \leftarrow \mathcal{P}_o \cap \mathcal{H}_d$
9: **end for**
10: Return $\mathcal{P}_o$

V. Example: Driving Game Scenario

We now apply our RS-IRL framework on a simulated driving game (Figure 1) with ten human participants to demonstrate that our approach is able to infer individuals' varying attitudes toward risk and mimic the resulting driving styles.

A. Experimental Setting

The setting consists of a leader car and a follower car. Participants controlled the follower car with a joystick (Figure 1). The follower's state $[x_f, y_f]^T \in \mathbb{R}^2$ consists of its x and y positions and its dynamics are given by $x_{f,k+1} = x_{f,k} + u_{x,k}\Delta t$, $y_{f,k+1} = y_{f,k} + v\Delta t + u_{y,k}\Delta t$. Here, $v = 20$ m/s is a nominal forward speed and the control inputs $u_x \in [-5, 5]$ m/s and $u_y \in [-10, 10]$ m/s are mapped linearly from the joystick position. The time step Δt is 0.1s.

The leader car plays the role of an "erratic driver". The dynamics of its state $[x_l, y_l]^T$ are given by $x_{l,k+1} = x_{l,k} + w_{x,k}\Delta t$, $y_{l,k+1} = y_{l,k} + v\Delta t + w_{y,k}\Delta t$ The leader's control input $[w_x, w_y]^T$ is chosen from a finite set $\mathcal{W} = \{w^{[1]}, \ldots, w^{[L]}\}$ with $L = 5$:

$$\mathcal{W} = \left\{ \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 5 \end{bmatrix} \begin{bmatrix} 0 \\ -7.5 \end{bmatrix} \begin{bmatrix} 2.5 \\ 0 \end{bmatrix} \begin{bmatrix} -2.5 \\ 0 \end{bmatrix} \right\} \text{ m/s.} \tag{25}$$

These "disturbance" realizations correspond to different speed settings for the leader and are generated randomly according to the pmf $p = [0.65, 0.025, 0.025, 0.2, 0.1]$. The disturbance is sampled every 20 time steps (2 seconds) and held constant in the interim. The leader car can thus be viewed as executing a random *maneuver* every 2 seconds. The dynamics of the relative positions x_{rel} and y_{rel} between the leader and follower cars can thus be written as an affine dynamical system.

Participants in the study were informed that their primary goal was to follow the leader car (described as an "erratic driver"), as closely as possible in the y direction. They were also instructed to track the leader's x position, but that this was not as important. A visual input in the form of a scoring bar (Figure 1) whose height is a linear function of y_{rel} was provided to them. Participants were informed that this bar represents an instantaneous score which will be aggregated over time, but that they will incur "significant penalties" for crossing the leader's y position (i.e., when $y_{\text{rel}} < 0$).

The leader car's five actions were described to participants, along with the fact that these actions are generated every two seconds and then held constant. In order to aid the participant in keeping track of the timing of the leader's maneuvers, a

visual input in the form of a timer bar (Figure 1) that counts down to the next disturbance was provided.

The experimental protocol for each participant consisted of three phases. The first phase (two minutes) involved the leader car moving forwards at the nominal speed and was meant for the participant to familiarize themselves with the simulation and joystick controls. The second and third phases (one and two minutes respectively) involved the leader car acting according to the model described above (with actions being sampled according to the pmf p). These two phases were identical, with the exception that participants were informed that the second phase was a training phase in which they could familiarize themselves with the entire simulation and the third phase would be the one where they are tested. For the results presented below, we split the data collected from the *third* phase into training and test sets of one minute each (corresponding to 30 two-second epochs where a disturbance is sampled for both the training and test sets).

Note that the pmf p is *not* shared with the participants. This experimental setting may thus be considered *ambiguous*. However, since participants are exposed to a training phase where they may build a mental model of disturbances, the setting may also be interpreted as one involving risk.

B. Modeling and Implementation

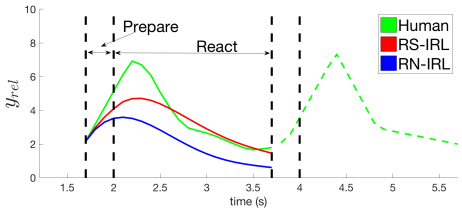
We model participants' behavior using the "prepare-react" framework presented in Section IV with the "prepare" phase starting 0.3 seconds before the leader's action is sampled. The "react" phase thus extends to 1.7 seconds after the disturbance. This parameter was chosen as being roughly reflective of observed participant behavior on our game scenario.

We represent our cost function as a linear combination of the following features (with unknown weights): $\psi_1 = x_{\text{rel}}^2$, $\psi_2 = (u_{y,k} - u_{y,k-1})^2$, $\psi_3 = \log(1 + e^{r y_{\text{rel}}}) - \log(2)$, $\psi_4 = \log(1 + e^{-r y_{\text{rel}}}) - \log(2)$. The second feature captures differences in users' joystick input in the $y-$direction (which is a more accurate indication of control effort for a joystick than its absolute position). The third and fourth features together form a differentiable approximation to the maximum of two linear functions and thus allow us to capture the asymmetric costs when $y_{\text{rel}} < 0$ and $y_{\text{rel}} \geq 0$. We set $r = 10$.

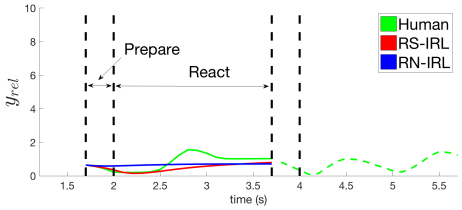
We apply Algorithm 2 for inferring the feature weights (using an implementation of the Inverse KKT approach [16]) and risk envelopes. The resulting LPs are solved using MOSEK [6] and take ~ 0.1 seconds to solve for each 2 second "prepare-react" period data on a 2.7GHz QuadCore 2015 MacBook Pro with 8GB RAM. Once the cost and risk envelope have been inferred from training data, predictions for control actions taken by participants on test data are made by solving Problem (22). This is a *convex* optimization problem since the chosen features are convex and the dynamical system is affine. Our MATLAB implementation takes approximately 1-3 seconds to solve using TOMLAB [23] and the SNOPT solver [20].

C. Results

Interestingly, our simulated driving scenario was rich enough to elicit a wide variety of qualitative behaviors from the ten participants. In particular, we observed two extreme policies. One extreme involved the follower pulling back significantly from the leader shortly before a disturbance is sampled and then getting close to the leader again once it selects a new action (e.g., Figure 4(a)). Another extreme was to follow the leader very closely with a small separation (e.g., Figure 4(b)). These two extremes can be interpreted as reflecting varying attitudes towards risk. The first policy is highly risk-averse as it always prepares for the worst-case eventuality (leader slowing down). The second policy corresponds to risk-neutral behavior, where low probability (but high cost) events are largely disregarded. We also observed a range of behaviors that lie between these two extremes.

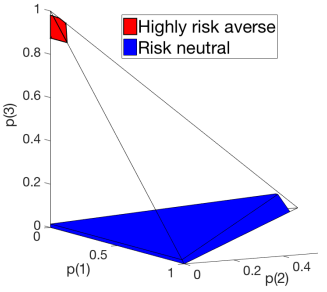


(a) Risk-averse participant

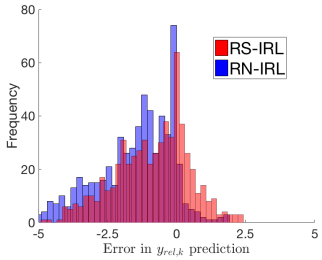


(b) Risk-neutral participant

Fig. 4: Comparisons of human trajectories with predictions from Risk-sensitive and Risk-neutral IRL for a risk-sensitive and risk-neutral participant.



(a) Risk envelopes



(b) Error histogram

Fig. 5: (a) Extracted risk envelopes (projected onto 3 dimensions) and (b) Histogram of errors in predictions of relative y-position for a risk-averse participant aggregated over 30 test trajectories.

Figure 5(a) presents risk envelopes extracted using our approach for two participants who exhibit these two extreme behaviors. Note that the polytopes, which are subsets of the 5-dimensional probability simplex, have been projected down to the first three dimensions (corresponding to the leader moving at the nominal speed, higher speed, and lower speed respectively) for visualization. Interestingly, the two polytopes correspond very well to our intuitive notions of risk/ambiguity averseness and neutrality. The risk-averse polytope is concentrated almost entirely in the region of the simplex corresponding to high probabilities of the leader slowing down, while the risk-neutral polytope considers only points in the simplex that assign a very low probability to this outcome.

Figures 4(a) and Figure 4(b) present examples of human trajectories (green) for the two participants compared with the predictions using our RS-IRL approach (i.e., by solving

problem (22) using the inferred cost function and polytope) for a single 2 second prepare-react period. We see that our RS-IRL approach reproduces the qualitatively different driving styles of the two participants. For the risk-averse participant, the predicted trajectory backs off the leader car in the "prepare" stage and moves close again once the leader chooses its action. For the risk-neutral participant, the predicted trajectory remains in close proximity to the leader at all times.

Figure 4 also compares the RS-IRL approach with one where the expert is modeled as minimizing the expected value of his/her cost function computed with respect to the pmf p. Since this risk-neutrality is the standard assumption made by traditional IRL approaches, it constitutes an important benchmark for comparison. We refer to this approach as risk-neutral IRL (RN-IRL). The predicted trajectories (blue) using this model are generated by solving Problem (20) with the risk operator ρ replaced by the expected value. As one would expect, the predictions using RS-IRL and RN-IRL are similar for the risk-neutral participant (Figure 4(b)). However, as Figure 4(a) demonstrates, the RN-IRL model does not predict the significant backing off behavior exhibited by the human and significantly underestimates y_{rel} over the 2 second period.

Figure 5(b) plots a histogram of errors $(y_{\text{rel},k}^{predicted} - y_{\text{rel},k}^{human})$ $(k = 0, ..., 20)$ for the predictions made by RS-IRL and RN-IRL computed for all 30 trajectories in our test set for the risk-averse participant. We observe that RN-IRL significantly underestimates y_{rel}, while the RS-IRL approach makes noticeably more accurate predictions. However, we note that RS-IRL still exhibits a slight bias towards under-predicting y_{rel}. This is because the risk-averse participant consistently backs off the leader car by a very large amount (as observed in Figure 4(a)), which may be explained by the fact that while our model assumes that the expert has an exact knowledge of the magnitudes of the disturbances/speeds in the set $\mathcal{W}$, this is only approximately true in reality (especially since the participants experienced the low speed setting quite rarely in the training phase). Thus, one would expect a more significant backing off maneuver if the participant overestimates the difference in the nominal and slow leader speed settings. This issue could potentially be dealt with by considering additional "spurious" disturbance settings (e.g., introducing a lower speed setting in $\mathcal{W}$) when applying our RS-IRL approach.

Table I presents comparisons of the average (over 30 test trajectories) of simulation errors in y_{rel} computed for RS-IRL and RN-IRL as $\Delta y_{\text{rel}} := \frac{1}{30}\sum_{i=1}^{30}\sqrt{\sum_k (y_{\text{rel},k,i}^{predicted} - y_{\text{rel},k,i}^{human})^2}$. Here, $y_{\text{rel},k,i}^{predicted}$ and $y_{\text{rel},k,i}^{human}$ are y_{rel} at time k for trajectory i for the predicted and actual trajectories respectively (Δx_{rel} is computed similarly). The RS-IRL predictions for y_{rel} are more accurate than RN-IRL for 8 out of 10 participants, with as much as a 30% improvement in some cases. For comparison, Participants $\#1$ and $\#8$ are the highly risk-averse and risk-neutral participants respectively from our previous case studies. As expected, RS-IRL is significantly better than RN-IRL for the risk-averse participant (and comparable for the risk-neutral one). The only significant outlier is Participant 9, for whom we found that the inferred polytope encompassed almost the entire simplex. This may indicate the need for more training data for that particular participant. Errors in x_{rel} for RS-IRL and RN-IRL are comparable for all participants as expected (since we don't expect risk aversion along this state).

Participant #	1	2	3	4	5	6	7	8	9	10
Δy_{rel} (RS-IRL)	6.7	8.5	6.3	5.2	5.8	2.6	4.0	3.9	11.6	3.9
Δy_{rel} (RN-IRL)	9.1	9.7	7.4	5.8	6.1	2.7	4.1	3.4	5.6	5.7
Δx_{rel} (RS-IRL)	2.6	2.9	1.9	2.2	2.8	2.0	1.7	3.4	4.9	3.7
Δx_{rel} (RN-IRL)	2.7	2.9	1.8	2.2	3.1	1.9	1.7	3.3	4.1	4.0

TABLE I: Comparisons of average (over 30 test trajectories) of simulation errors computed for Risk-sensitive and Risk-neutral IRL models. The RS-IRL predictions for y_{rel} are more accurate than the Expected Value model (RN-IRL) for 8 out of the 10 participants, with as much as 30% improvement.

VI. Discussion and Conclusions

We have presented an approach for IRL that explicitly accounts for risk sensitivity in experts. We proposed a flexible modeling framework based on coherent risk metrics that allows us to capture an entire spectrum of risk assessments from risk-neutral to worst-case for a rich class of static and dynamic decision-making settings. We developed efficient LP based algorithms for inferring an expert's risk preferences from demonstrations. Results on a simulated driving game with ten participants demonstrate that our technique is able to infer and mimic qualitatively different driving styles ranging from risk-neutral to highly risk-averse in a data-efficient manner, while more accurately capturing participant behavior than a risk-neutral model. To our knowledge, the results in this paper constitute the first attempt to explicitly take into account risk-sensitivity in IRL under *general* axiomatically justified risk models that jointly capture risk and ambiguity.

Challenges and future work: At the modeling level, we plan to relax some of the assumptions made about the expert in the multi-step model in Section IV. In particular, while we can easily handle *aperiodic* disturbances where the expert knows the timing of disturbances, handling cases where the disturbance times themselves are uncertain is important. For our experiments, we provided participants with a visual aid to keep track of timing; without this timing knowledge, we may see qualitatively different behavior (e.g., risk-averse participants maintaining a large but constant distance from the leader instead of periodically pulling back and getting closer). We will also extend our model to the case where the expert's look-ahead horizon extends to multiple disturbances. At an algorithmic level, being able to elegantly handle outliers and changes in the expert's policy are important considerations. For example, if an expert changes his/her policy from risk-averse to risk-neutral midway through demonstrations, our approach will conclude that the expert is risk-averse (since once portions of the simplex have been pruned away, there is no way to "undo" this in our approach). From a theoretical perspective, we plan to study the convergence properties of Algorithm 1 with careful consideration to notions of *observability* of the risk envelope (and cost). Future experiments will focus on more realistic driving scenarios and learning risk preferences of human UAV pilots in cluttered environments. Finally, we plan on studying the *game theoretic* IRL setting (e.g., [43, 48]), where multiple risk-sensitive agents interact.

We believe that the approach described here along with the indicated future directions represent an important step towards endowing our robotic systems with the ability to predict, infer, and mimic risk-sensitive behavior, which is crucial for safety-critical applications where humans and robots interact.

Acknowledgments

The authors were partially supported by the Office of Naval Research, Science of Autonomy Program, under Contract N00014-15-1-2673, and by the Toyota Research Institute ("TRI"). This article solely reflects the opinions and conclusions of its authors and not ONR, TRI or any other Toyota entity.

References

[1] P. Abbeel and A. Y. Ng. Apprenticeship learning via inverse reinforcement learning. In *Int. Conf. on Machine Learning*, 2004.

[2] P. Abbeel and A. Y. Ng. Exploration and apprenticeship learning in reinforcement learning. In *Int. Conf. on Machine Learning*, 2005.

[3] C. Acerbi. Spectral measures of risk: A coherent representation of subjective risk aversion. *Journal of Banking & Finance*, 26(7):1505–1518, 2002.

[4] C. Acerbi and D. Tasche. On the coherence of expected shortfall. *Journal of Banking & Finance*, 26(7):1487–1503, 2002.

[5] Maurice Allais. Le comportement de lhomme rationnel devant le risque: critique des postulats et axiomes de lécole américaine. *Econometrica*, 21(4):503–546, 1953.

[6] MOSEK ApS. MOSEK optimization software. Available at https://mosek.com/.

[7] P. Artzner, F. Delbaen, J.-M. Eber, and D. Heath. Coherent measures of risk. *Mathematical Finance*, 9(3): 203–228, 1999.

[8] A. Axelrod, L. Carlone, G. Chowdhary, and S. Karaman. Data-driven prediction of EVAR with confidence in time-varying datasets. In *IEEE Conf. on Decision and Control*, 2016.

[9] N. C. Barberis. Thirty years of prospect theory in economics: A review and assessment. *Journal of Economic Perspectives*, 27(1):173–195, 2013.

[10] N. Bäuerle and J. Ott. Markov decision processes with average-value-at-risk criteria. *Mathematics of Operations Research*, 74(3):361–379, 2011.

[11] D. Carton, V. Nitsch, D. Meinzer, and D. Wollherr. Towards assessing the human trajectory planning horizon. *PLoS ONE*, 11(12):e0167021, 2016.

[12] Y. Chow and M. Pavone. A framework for time-consistent, risk-averse model predictive control: Theory and algorithms. In *American Control Conference*, June 2014.

[13] Y. Chow, A. Tamar, S. Mannor, and M. Pavone. Risk-sensitive and robust decision-making: a CVaR optimization approach. In *Advances in Neural Information Processing Systems*, 2015.

[14] A. Eichhorn and W. Römisch. Polyhedral risk measures in stochastic programming. *SIAM Journal on Optimization*, 16(1):69–95, 2005.

[15] D. Ellsberg. Risk, ambiguity, and the savage axioms. *The Quarterly Journal of Economics*, 75(4):643–669, 1961.

[16] P. Englert and M. Toussaint. Inverse KKT learning cost functions of manipulation tasks from demonstrations. In *Int. Symp. on Robotics Research*, 2015.

[17] J. A. Filar, L. C. M. Kallenberg, and H.-M. Lee. Variance-penalized Markov decision processes. *Mathematics of Operations Research*, 14(1):147–161, 1989.

[18] I. Gilboa and M. Marinacci. Ambiguity and the Bayesian paradigm. In *Readings in Formal Epistemology*, chapter 21. First edition, 2016.

[19] I. Gilboa and D. Schmeidler. Maxmin expected utility with non-unique prior. *Journal of Mathematical Economics*, 18(2):141–153, 1989.

[20] P. E. Gill, W. Murray, and M. A. Saunders. SNOPT: An SQP algorithm for large-scale constrained optimization. *SIAM Review*, 47(1):99–131, 2005.

[21] P.W. Glimcher and E. Fehr. *Neuroeconomics*. Elsevier, second edition, 2014.

[22] J. D. Hey, G. Lotito, and A. Maffioletti. The descriptive and predictive adequacy of theories of decision making under uncertainty/ambiguity. *Journal of Risk and Uncertainty*, 41(2):81–111, 2010.

[23] K. Holmström and M. M. Edvall. The TOMLAB optimization environment. In *Modeling Languages in Mathematical Optimization*. 2004.

[24] R. Howard and J. Matheson. Risk-sensitive Markov decision processes. *Management Science*, 8(7):356–369, 1972.

[25] M. Hsu, M. Bhatt, R. Adolphs, D. Tranel, and C. F. Camerer. Neural systems responding to degrees of uncertainty in human decision-making. *Science*, 310 (5754):1680–1683, 2005.

[26] Daniel Kahneman and Amos Tversky. Prospect theory: An analysis of decision under risk. *Econometrica*, pages 263–291, 1979.

[27] J. Z. Kolter, P. Abbeel, and A. Y. Ng. Hierarchical apprenticeship learning with application to quadruped locomotion. In *Advances in Neural Information Processing Systems*, 2007.

[28] M. Kuderer, S. Gulati, and W. Burgard. Learning driving styles for autonomous vehicles from demonstration. In *Proc. IEEE Conf. on Robotics and Automation*, 2015.

[29] S. Levine and V. Koltun. Continuous inverse optimal control with locally optimal examples. In *Int. Conf. on Machine Learning*, 2012.

[30] O. Mihatsch and R. Neuneier. Risk-sensitive reinforcement learning. *Machine Learning*, 49(2):267–290, 2002.

[31] K. Mombaur, A. Truong, and J.-P Laumond. From human to humanoid locomotion—an inverse optimal control approach. *Autonomous Robots*, 28(3):369–383, 2010.

[32] A. Ng and S.J. Russell. Algorithms for inverse reinforcement learning. In *Int. Conf. on Machine Learning*, 2000.

[33] A. Nilim and L. El Ghaoui. Robust control of Markov decision processes with uncertain transition matrices. *Operations Research*, 53(5):780–798, 2005.

[34] T. Osogami. Robustness and risk-sensitivity in Markov decision processes. In *Advances in Neural Information Processing Systems*, 2012.

[35] T. Park and S. Levine. Inverse optimal control for humanoid locomotion. In *Robotics: Science and Systems Workshop on Inverse Optimal Control and Robotic Learning from Demonstration*, 2013.

[36] M. Petrik and D. Subramanian. An approximate solution method for large risk-averse Markov decision processes. In *Proc. Conf. on Uncertainty in Artificial Intelligence*, 2012.

[37] M. Rabin. Risk aversion and expected-utility theory: A calibration theorem. *Econometrica*, 68(5):1281–1292, 2000.

[38] D. Ramachandran and E. Amir. Bayesian inverse reinforcement learning. In *Proc. Int. Conf. on Autonomous Agents and Multiagent Systems*, 2007.

[39] R. T. Rockafellar. Coherent approaches to risk in optimization under uncertainty. In *OR Tools and Ap-*

plications: Glimpses of Future Technologies, chapter 3. INFORMS, 2007.

[40] R. T. Rockafellar and S. Uryasev. Optimization of conditional value-at-risk. *Journal of Risk*, 2:21–41, 2000.

[41] S. Russell. Learning agents for uncertain environments. In *Proc. Computational Learning Theory*, 1998.

[42] A. Ruszczyński. Risk-averse dynamic programming for Markov decision process. *Mathematical Programming*, 125(2):235–261, 2010.

[43] D. Sadigh, S. Sastry, S. A. Seshia, and A. D. Dragan. Planning for autonomous cars that leverage effects on human actions. In *Robotics: Science and Systems*, 2016.

[44] A. Shapiro. On a time consistency concept in risk averse multi-stage stochastic programming. *Operations Research Letters*, 37(3):143–147, 2009.

[45] Y. Shen, M. J. Tobia, T. Sommer, and K. Obermayer. Risk-sensitive reinforcement learning. *Neural Computation*, 26(7):1298–1328, 2014.

[46] A. Tamar, D. Di Castro, and S. Mannor. Policy gradients with variance related risk criteria. In *Int. Conf. on Machine Learning*, 2012.

[47] J. von Neumann and O. Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, 1944.

[48] Kevin Waugh, Brian D. Ziebart, and J. Andrew Bagnell. Computational rationalization: The inverse equilibrium problem. In *Int. Conf. on Machine Learning*, 2011.

[49] H. Xu and S. Mannor. Distributionally robust Markov decision processes. In *Advances in Neural Information Processing Systems*, 2010.

[50] B. D. Ziebart, A. Maas, J. A. Bagnell, and A. K. Dey. Maximum entropy inverse reinforcement learning. In *Proc. AAAI Conf. on Artificial Intelligence*, 2008.

[51] B. D. Ziebart, N. Ratliff, G. Gallagher, C. Mertz, K. Peterson, J. A. Bagnell, M. Hebert, A. K. Key, and S. Srinivasa. Planning-based prediction for pedestrians. In *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, 2009.

[52] M. Zucker, J. A. Bagnell, C. G. Atkeson, and J. Kuffner. An optimization approach to rough terrain locomotion. In *Proc. IEEE Conf. on Robotics and Automation*, 2010.