Bounded Risk-Sensitive Markov Game and Its Inverse Reward Learning Problem

Ran Tian* **Liting Sun*** **Masayoshi Tomizuka**

{rantian, litingsun, tomizuka}@berkeley.edu
Department of Mechanical Engineering
University of California at Berkeley
* First two authors contributed equally

Abstract

Classical game-theoretic approaches for multi-agent systems in both the forward policy learning/design problem and the inverse reward learning problem often make strong rationality assumptions: agents are risk-neutral to all uncertainties and have unlimited computation resources to infer others' actions and solve for the optimal actions that maximize their expected rewards. Such assumptions, however, mismatch with substantial observations of humans' behaviors such as satisficing with sub-optimal actions, risk-seeking, and loss-aversion decisions. In this paper, we investigate the problem of bounded risk-sensitive Markov Game (BRSMG) and its inverse reward learning problem. Instead of assuming agents perform infinite levels of reasoning over possible actions of others, in BRSMG, we consider the influence of finite-level intelligence. Instead of assuming agents maximize their expected rewards (a risk-neutral measure), in BRSMG, we consider the impact of risk-sensitive measures such as the cumulative prospect theory. Convergence proofs of BRSMG for both the forward policy learning and inverse reward learning are provided. Moreover, simulation results in an indoor navigation scenario show that the behaviors of agents in BRSMG demonstrate both risk-averse and risk-seeking phenomena, which are consistent with observations from humans. In the inverse reward learning problem, the proposed risk-sensitive inverse learning algorithm can effectively recover both the rewards and the parameters of the risk-sensitive measure of agents given demonstrations of their interactive behaviors.

1 Introduction

Markov Game (MG), as an approach to model interactions and decision-making processes of intelligent agents in multi-agent systems, dominates in many domains spanning from economics [1] to games [17], and to human-robot/machine interaction [3, 7]. In classical MGs, all agents are assumed to be perfectly rational. Namely, all agents are fully aware of the mutual influence among them and perform infinite levels of strategic reasoning over the possible responses of others in order to find out the optimal actions that maximize their own expected discounted rewards. For instance, in a two-player game, at each step t, agent 1 makes decisions based on his/her belief in agent 2's behavior model in which agent 2 is assumed to behave according to his/her belief in agent 1's model .... If the beliefs match the actual models, perfect Markov strategies of all agents can be found by solving the Markov-perfect equilibrium (MPE) of the game where a Nash equilibrium is reached. Under such assumptions, we can either explore humans' optimal strategies with pre-defined rewards or perform the inverse learning problem that recovers humans' reward structures by observing their behaviors based on, for instance, the maximum entropy inverse reinforcement learning [22].

However, real human behaviors often significantly deviate from such "perfect rationality" assumptions from two major aspects. First, mounting evidence has shown that rather than spending a great amount of efforts to hunt for the best choice, humans often choose actions that are satisfying (*i.e.*, actions that are above their pre-defined thresholds according to certain criteria) and relatively quick and easy to find. Simon [18] formulated such a decision-making strategy as bounded rationality. One

Preprint. Under review.

aspect of the bounded rationality is the cognitive hierarchy theory (CHT) [19, 6] which finds that humans cannot afford the complexity of infinite layers of strategic thinking in interactions but behave more like agents with finite levels of intelligence (rationality). Second, instead of optimizing the risk-neutral expected rewards, humans demonstrate strong tendency towards risk-sensitive measures when evaluating the outcomes of their actions under uncertainties. They are risk-seeking in terms of gains and risk-averse for losses. For example, people often prefer a choice with high reward but low probability over the one with low reward but high probability although the latter has a higher expected value. Humans also have stronger motivations to avoid losses than to obtain gains [20]. Such deviations make modeling real humans' interactive behaviors using classical MGs very difficult.

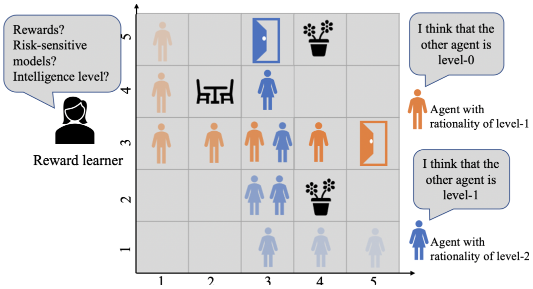

Rewards?
Risk-sensitive
models?
Intelligence level?
Reward learner
I think that the other agent is level-0
Agent with rationality of level-1
I think that the other agent is level-1
Agent with rationality of level-2
1
2
3
4
5
1
2
3
4
5

Figure 1: Modeling interactions between humans as bounded risk-sensitive Markov Games: two human agents (orange and blue) plan to exit the room through specified doors without collisions with obstacles and each other. We aim to answer two questions: 1) assuming both humans have bounded level of strategic reasoning and risk-sensitive performance measures, how will their optimal policies differ from that of classical MGs? and 2) what algorithms can recover the rewards and sensitivity parameters given their trajectories?

There have been extensive works trying to capture more realistic decision-making processes of humans. For instance, the cumulative prospect theory (CPT) [20] formulated a model that can well explain a substantial amount of human risk-sensitive behaviors that are beyond the scope of expected utility theory. Risk-sensitive strategy learning [12, 5, 10] and its inverse problems [13, 14] have also been investigated in single-agent settings. In the community of MGs, researchers have studied the influence of bounded rationality in the forward strategy design problem by either introducing additional costs to agents' actions such as in [2], [8] and [9], or developing bounded-level policies as in [15].

In this paper, we consider a general-sum two-player Markov Game with bounded rational agents and a risk-sensitive performance measure, *i.e.*, the bounded risk-sensitive Markov Game (BRSMG). Both the forward strategy learning/design problem and its inverse reward learning problem are studied, as shown in Fig. 1. In the forward strategy learning/design problem, rather than introducing additional computational costs to agents' actions, we investigate the impact of finite levels of strategic thinking as a special form of bounded computation resource. Moreover, to model the influence of humans' risk sensitivity, we study agents' optimal policies in terms of maximizing their cumulative prospects according to the CPT rather than the expected utilities. In the inverse reward learning problem, we develop an risk-sensitive inverse learning algorithm which can recover not only the rewards of agents but also the parameters in their risk measure from their demonstrated behaviors, without prior information on their intelligence levels. To our best knowledge, our work is the first to consider both bounded rationality and risk-sensitivity in general-sum MGs for both the forward the inverse problems.

2 Preliminaries

2.1 Classical Markov Game

In classical two-player MGs, each agent is represented by a Markov decision process (MDP). We denote a MG as $\mathcal{G} \triangleq \langle \mathcal{P}, \mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{T}, \tilde{\gamma} \rangle$, where $\mathcal{P}$=$\{1, 2\}$ is the set of agents in the game, $\mathcal{S}$=(S^1, S^2) and $\mathcal{A}$=(A^1, A^2) are, respectively, the joint state and action spaces of the two agents, $\mathcal{R}$=(R^1, R^2) is the set of agents' one-step reward functions with $R^i : \mathcal{S} \times A^i \times A^{-i} \rightarrow \mathbb{R}$, $\mathcal{T} : \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{S}$ represents the state transition of the game (we consider deterministic state transitions in this paper), $-i = \mathcal{P} \setminus \{i\}$ represents the opponent of agent i, and $\tilde{\gamma}$ is the reward discount factor that shared by both agents.

We let $\pi^i : \mathcal{S} \rightarrow A^i$ denote a deterministic policy of agent i. At step t, given current state s_t, each agent in classic MGs is trying to find the optimal action that maximizes his/her expected total discounted rewards. Namely, the optimal policy $\pi^{*,i}$ is given by $\pi^{*,i}$= $\arg\max_{\pi^i} V^{i,\pi^i}(s_t)$, where $V^{i,\pi^i}(s_t) = \mathbb{E}_{\pi^{-i}}\left[\sum_{\tau=0}^{\infty} \tilde{\gamma}^{\tau} R^i(s_{t+\tau}, a^i_{t+\tau}, a^{-i}_{t+\tau})\right]$ represents the value function at s_t, *i.e.*, the expected return starting from s_t under policy π^i with uncertainties on the estimate of the opponent's policy π^{-i}. The notations $a^{-i}_{t+\tau}$ and $s_{t+\tau}$, respectively, represent the predicted future action of the opponent and the corresponding state at step $t + \tau$. At the MPE, both agents achieve their optimal policies. Due to the mutual influence between the value functions of both agents, the solution of the MPE policies is typically of NP-hard.

2.2 Cognitive hierarchy theory

The cognitive hierarchy theory (CHT) is an alternative to the equilibrium solution concept in games [19, 4, 6]. Instead of assuming all agents to perform infinite levels of (circular) strategic reasoning, CHT considers the bounded rationality of humans. It models humans as agents who only do a finite iteration of strategic thinking. In fact, experiment results in the p-beauty contest game found that the average depth of thinking for human is 1.5, and over 80% people do, at most, two reasoning steps [19]. The iterative process of CHT starts with level-0 agents who do not perform any strategic reasoning and only make decisions based on their priors over the models of others. As shown in Fig. 1, a level-k agent with $k \in \mathbb{N}^+$ believes that all other agents in the game can be modeled as level-$(k-1)$ agents, and predicts their responses based on such an assumption. The CHT model has therefore reduced the complex circular strategic thinking in classical MGs to finite levels of iterative optimizations. With an anchoring level-0 policy, policies under different rationality levels, namely, $k = 1, 2, 3, \ldots$, can be sequentially and iteratively solved for each agent.

2.3 Cumulative prospect theory

The cumulative prospect theory (CPT) is a non-expected utility theory proposed by Kahneman and Tversky in [11] to describe the risk-sensitivity of humans' decision-making processes. It can explain many systematic biases of human behaviors deviating from risk-neutral decisions such as risk-avoiding, risk-seeking, and framing effects [11].

Definition 1 (**CPT value**). *For a random variable X, the CPT value of X is defined by*

1. *If X is continuous, then*
$$\mathbb{CPT}(X){=}\int_0^\infty w^+\left(\mathbb{P}\left(u^+(X{-}x^0)>y\right)\right)dy{-}\int_0^\infty w^-\left(\mathbb{P}\left(u^-(X{-}x^0)>y\right)\right)dy. \tag{1}$$

2. *If X is discrete satisfying $\sum_{i=-m}^{n}\mathbb{P}(X{=}x_i){=}1$, $x_i{\geq}x^0$ for $i{=}0,\cdots,n$, and $x_i{<}x^0$ for $i{=}{-}m,\cdots,-1$, then*
$$\mathbb{CPT}(X){=}\sum\nolimits_{i=0}^{n}\tilde{\rho}^+\left(P(X{=}x_i)\right)u^+(X-x^0)-\sum\nolimits_{i=-m}^{-1}\tilde{\rho}^-\left(P(X{=}x_i)\right)u^-(X-x^0), \tag{2a}$$
$$\tilde{\rho}^+\left(P(X{=}x_i)\right)=\left[w^+\left(\sum\nolimits_{j=i}^{n}\mathbb{P}(X{=}x_j)\right)-w^+\left(\sum\nolimits_{j=i+1}^{n}\mathbb{P}(X{=}x_j)\right)\right], \tag{2b}$$
$$\tilde{\rho}^-\left(P(X{=}x_j)\right)=\left[w^-\left(\sum\nolimits_{j=-m}^{i}\mathbb{P}(X{=}x_j)\right)-w^-\left(\sum\nolimits_{j=-m}^{i-1}\mathbb{P}(X{=}x_j)\right)\right]. \tag{2c}$$

The functions $w^+ : [0,1]{\to}[0,1]$ and $w^- : [0,1]{\to}[0,1]$ are two continuous non-decreasing functions which are referred as the probability decision weighting functions. They describe the characteristics of humans to deflate high probabilities and inflate low probabilities. The two functions $u^+ : \mathbb{R}{\to}\mathbb{R}^+$ and $u^- : \mathbb{R}{\to}\mathbb{R}^+$ are concave utility functions which are, respectively, monotonically non-decreasing and non-increasing. The notation x^0 denotes a "reference point" that separates the value X into gains ($X{\geq}x^0$) and losses ($X{<}x^0$). Handling gains and losses separately is a key feature of the CPT model and it captures the different preferences of humans towards gains and losses. Moreover, the slope of u^- is usually larger than that of u^+ to show that humans weigh losses more than gains. Without loss of generality, we set $x^0 = 0$ and omit x^0 in the rest of this paper. Note that when both the probability weighting functions and the utility functions take the identity function, *i.e.*, $w^+{=}w^-{=}u^+{=}u^-{=}\mathbb{1}$, the CPT value in (1) and (2) reduces to $\mathbb{E}[X^+]{-}\mathbb{E}[X^-]$, showing the connection to the expected value, i.e., the risk-neutral performance measure.

Many experimental studies have shown that representative functional forms for u and w are: $u^+(x){=}(x)^\alpha$ if $x{\geq}0$, and $u^+(x){=}0$ otherwise; $u^-(x){=}\lambda(-x)^\beta$ if $x{<}0$, and $u^-(x){=}0$ otherwise; $w^+(p){=}\frac{p^\gamma}{(p^\gamma+(1-p)^\gamma)^{1/\gamma}}$ and $w^-(p) = \frac{p^\delta}{(p^\delta+(1-p)^\delta)^{1/\delta}}$. The parameters $\alpha, \beta, \gamma, \delta{\in}(0,1]$ are model parameters. We adopt these two representative functions in this paper, and focus on the discrete-domain of CPT value in (2). Section 1 of the supplementary material illustrates the probability weighting functions and the utility functions.

3 Bounded Risk-Sensitive Markov Game

As discussed in Section 1, classical MGs implicitly assume: 1) all agents are risk-neutral, *i.e.*, they are maximizing their expected utilities under uncertainties, and 2) all agents are unbounded in terms of their intelligence levels and computation resources. In this section, we investigate the agents' policies in a new general-sum two-player MG, where each agent is bounded-rational with a risk-sensitive performance measure.

3.1 Bounded risk-sensitive policies

According to the cognitive hierarchy theory in Section 2.2, agents in the bounded risk-sensitive Markov Games are of finite-level intelligence. Specifically, if agent i is assumed to have intelligence level-k, $k \in \mathbb{N}^+$, then he/she believes that the opponent player is of level-$(k{-}1)$. To solve for the risk-sensitive level-k policy $\pi^{*,i,k}$, we start with defining a level-0 policy as the anchoring policy.

Definition 2 (The anchoring policy of level-0 agents)**.** *At time step t, given state s_t and the action a^{-i} from the opponent agent, the stochastic policy of a level-0 agent i is a pure-follower policy satisfying*

$$\pi^{i,0}(s,a^i,a^{-i}) = \frac{\exp\left(R^i(s,a^i,a^{-i})\right)}{\sum_{a'\in A^i}\exp\left(R^i(s,a',a^{-i})\right)}, \forall s\in\mathcal{S},\ \forall a^i\in A^i,\ \forall a^{-i}\in A^{-i}. \tag{3}$$

With the anchoring policy defined, we can start the iterative process to solve for the optimal policies under higher levels of intelligence.

First, based on the CPT model defined in (2), given current state s_t, a level-k agent i tries to maximize the following discounted future cumulative prospects:

$$\max_{\pi^{i,k}} J_{\pi^{i,k}}(s_t) = \max_{\pi^{i,k}} \mathbb{CPT}_{\pi^{*,-i,k-1}}\left[R^i(s_t,a^i_t,a^{-i}_t) + \tilde{\gamma}\mathbb{CPT}_{\pi^{*,-i,k-1}}\left[R^i(s_{t+1},a^i_{t+1},a^{-i}_{t+1}) + \dots\right]\right], \tag{4}$$

where $\pi^{*,-i,k-1}: \mathcal{S}\times A^{-i}\to[0,1]$ denotes the optimal risk-sensitive policy of agent $-i$, whose level of intelligence is believed to be $(k{-}1)$ from agent i's perspective. The action $a^{-i}_{t+\tau}$ denotes the predicted action of agent i sampled from $\pi^{*,-i,k-1}$ at time step $t{+}\tau$.

We let $\pi^{*,i,k}$ denote the optimal risk-sensitive level-k policy of agent i, and define $V^{*,i,k}(s_t) \triangleq J_{\pi^{*,i,k}}(s_t)$ in (4) as the optimal *CPT value* at s_t. It represents the optimal CPT value that agent i could collect in the future if he/she executes the optimal policy $\pi^{*,i,k}$. Therefore, similar to classical MGs, according to (4), the optimal CPT value at any $s\in\mathcal{S}$ satisfies [16, 12]:

$$V^{*,i,k}(s) = \max_{a^i\in A^i} \mathbb{CPT}_{\pi^{*,-i,k-1}}\left[R^i(s,a^i,a^{-i}) + \tilde{\gamma}V^{*,i,k}(s')\right], s' = \mathcal{T}_{a^{-i}\sim\pi^{*,-i,k-1}}(s,a^i,a^{-i}). \tag{5}$$

Also, at state s, we define the optimal value of agent i induced by taking action a^i as its optimal *CPT Q-value*, *i.e.*, $Q^{*,i,k}(s,a^i){=}\mathbb{CPT}_{\pi^{*,-i,k-1}}\left[R^i(s,a^i,a^{-i}){+}\tilde{\gamma}V^{*,i,k}(s')\right]$. Apparently, we have $V^{*,i,k}(s){=}\max_{a^i} Q^{*,i,k}(s,a^i)$, similar to classical MGs. Based on such definitions and the Boltzmann model [21], the optimal policy $\pi^{*,i,k}$ for all $k{\geq}1$ can be written as

$$\pi^{*,i,k}(s,a^i) = \frac{\exp\left(\beta Q^{*,i,k}(a^i,s)\right)}{\sum_{a'\in\mathcal{A}^i}\exp\left(\beta Q^{*,i,k}(a',s)\right)}, \forall s\in\mathcal{S}, \forall a^i\in A^i, \tag{6}$$

where $\beta{\geq}0$ defines the level of the agents conforming to the optimal strategy. We set $\beta{=}1$.

Next, we need to find the optimal level-k risk-sensitive policy $\pi^{*,i,k}$ by solving (5) and (6) for $i\in\mathcal{P}$ with $k=1,2,\dots$ in an iterative way.

3.2 The convergence of bounded risk-sensitive policies

Theorem 1. *Define the tuple $\langle s,a^i,a^{-i}\rangle{:=}c^{a^{-i}}_{s,a^i}$ and normalize the transformed probabilities $\tilde{\rho}^i(c^{a^{-i}}_{s,a^i}){:=}\tilde{\rho}^i(\mathbb{P}(a^{-i}|s,a^i))$ by*

$$\rho^i(c^{a^{-i}}_{s,a^i}) = \begin{cases}\tilde{\rho}^i(c^{a^{-i}}_{s,a^i})/\max_{a^i}\sum_{a^{-i}}\tilde{\rho}^i(c^{a^{-i}}_{s,a^i}), & \textit{if } k=1,\\ \tilde{\rho}^i(c^{a^{-i}}_{s,a^i})/\sum_{a^{-i}}\tilde{\rho}^i(c^{a^{-i}}_{s,a^i}), & \textit{otherwise.}\end{cases} \tag{7}$$

For an arbitrary agent $i{\in}\mathcal{P}$, if the one-step reward R^i is lower-bounded by R_{min} with $R_{min}\geq 1$, then $\forall s\in\mathcal{S}$ and all intelligence levels with $k{=}1,2,\cdots$, the dynamic programming problem in (5) can be solved by the following value iteration algorithm:

$$V^{i,k}_{m+1}(s) = \max_{a^i\in A^i}\sum\nolimits_{a^{-i}\in A^{-i}}\rho^i(c^{a^{-i}}_{s,a^i})u^i\left(R^i(s,a^i,a^{-i}){+}\tilde{\gamma}V^{i,k}_m(s')\right), \quad s'=\mathcal{T}(s,a^i,a^{-i}). \tag{8}$$

Moreover, as $m\to\infty$, $V^{i,k}_{m+1}$ converges to the optimal value function $V^{*,i,k}(s)$.

Proof. We prove the theorem by induction. For $k{=}1$, $\pi^{*,-i,k-1}{=}\pi^{-i,0}$ in Definition 2, and with (7), (5) reduces to a single-agent policy optimization problem where the value iteration algorithm can be proved for convergence. Then we prove that for any level $k{>}1$, $\pi^{*,-i,k-1}$ is known *a priori* and does not depend on a_i, which again reduces (5) into a single-agent policy optimization problem. More details are given in Section 2 of the supplementary material. ■

The algorithm in Theorem 1 is summarized in Algorithm 1, where the notation $\mathcal{B}$ denotes a CPT-based Bellman operator and is defined as: $\mathcal{B}V_m^{i,k} = V_{m+1}^{i,k}$.

4 The Inverse Reward Learning Problem

In Section 3, we investigate how the policies of agents are obtained in the BRSMGs. In this section, we consider the inverse problem, that is, inferring agents' reward and risk-sensitive parameters from their interactive behaviors in BRSMGs. In particular, we let $k_{\max} = 2$ since psychology studies found most humans do at most two layers of strategic thinking[19].

Algorithm 1: Bounded risk-sensitive policies

```
Input: Markov Game model $\mathcal{G}$, highest intelligence
  level $k_{\max}$, and the anchoring policy $\pi^0$.
Output: $\{\pi^{*,i,k}\}$, $i \in \mathcal{P}$ and $k = 1, \dots, k_{\max}$.
Initialize $k = 1$;
while $k \le k_{max}$ do
    for $i \in \mathcal{P}$ do
        Initialize $V^{i,k}(s), \forall s \in \mathcal{S}$;
        while $V^{i,k}$ not converged do
            for $s \in \mathcal{S}$ do
                $V^{i,k}(s) \leftarrow \mathcal{B}V^{i,k}(s)$;
            end for
        end while
        for $(s, a^i) \in \mathcal{S} \times A^i$ do
            Compute $\pi^{*,i,k}(s, a^i)$ based on (6);
        end for
    end for
    $k \leftarrow k + 1$;
end while
Return $\{\pi^{*,i,k}\}$, $i \in \mathcal{P}$ and $k \in \mathbb{K}$.
```

4.1 Problem formulation

We consider the situations where the two agents in a BRSMG that differ in their intelligence levels: one with $k{=}1$ and the other with $k{=}2$. But we do not know *a priori* the exact type of each agent. Moreover, we assume that $x^0{=}0$ and the value function $u^+(\cdot)$ in CPT model in (2) is an identity function, and consider only the influence from the weighting function $w^+(p){=}p^\gamma / \left(p^\gamma{+}(1{-}p)^\gamma\right)^{1/\gamma}$ specified via γ. Moreover, we assume that the one-step rewards for both agents can be linearly parameterized by a group of selected features: $\forall i{\in}\mathcal{P}$, $R^i(s, a^i, a^{-i}){=}(\omega^i)^\intercal\Phi^i(s, a^i, a^{-i})$, where $\Phi^i(s, a^i, a^{-i}){:}\mathcal{S}{\times}A^i{\times}A^{-i}{\to}\mathbb{R}^d$ is a known feature function that maps a game state s, an action of agent i, and an action of agent $-i$ to a d-dimensional feature vector, and $\omega^i{\in}\mathbb{R}^d$ is a d-dimensional reward parameter vector.

Under such circumstances, we define $\bar{\omega}{=}(\bar{\gamma}, \bar{\omega}^{\mathrm{r}})$, where $\bar{\gamma}{=}(\gamma^i, \gamma^{-i})$ and $\bar{\omega}^{\mathrm{r}}{=}(\omega^i, \omega^{-i})$, respectively, represent the parameters in the weighting functions and reward functions of both agents. Then, given a set of demonstrated trajectories from the two players in a BRSMG denoted by $\mathcal{D}{=}\{\xi_1, \cdots, \xi_M\}$ with $\xi{=}\{(s_0, \bar{a}_0), \dots, (s_{N-1}, \bar{a}_{N-1})\}$, $s_t{\in}\mathcal{S}$, and $\bar{a}_t{\in}\mathcal{A}$ $(t{=}0, \dots, N{-}1)$, we aim to retrieve the underlying reward parameters and risk-sensitive parameter, *i.e.*, $\bar{\omega}$, from $\mathcal{D}$. Based on the principle of Maximum Entropy as in [22] , the problem is equivalent to solving the following optimization problem:

$$\max_{\bar{\omega}} \sum\nolimits_{\xi\in\mathcal{D}} \log \mathbb{P}\left(\xi|\bar{\omega}\right) = \max_{\bar{\omega}} \sum\nolimits_{\xi\in\mathcal{D}} \log \prod\nolimits_{t=0}^{N-1} \mathbb{P}(\bar{a}_t|s_t, \bar{\omega}), \tag{9}$$

where $\mathbb{P}(\bar{a}_t|s_t, \bar{\omega})$ is the joint likelihood of agents' actions conditioned on states and parameters. We solve this optimization problem via gradient ascent. First, we need to find out its gradient.

4.2 Gradient of the log-likelihood of a demonstration

Since we assume that the two agents have different but unknown intelligence levels ($k{\in}\mathbb{K}{=}\{1, 2\}$), the log-likelihood of a joint trajectory ξ is given as follows:

$$\log \mathbb{P}(\xi|\bar{\omega}) = \sum\nolimits_{t=0}^{N-1} \log \sum\nolimits_{k\in\mathbb{K}} \pi_{\bar{\omega}}^{*,i,k}(s_t, a_t^i)\pi_{\bar{\omega}}^{*,-i,-k}(s_t, a_t^{-i})\mathbb{P}(k|\xi_{t-1}, \bar{\omega}), \tag{10}$$

where $\pi_{\bar{\omega}}^{*,i,k}$ and $\pi_{\bar{\omega}}^{*,-i,-k}$, respectively, represent the policies of the level-k agent i and another level-$(-k)$ agent $(-i)$ (note that $-k{\triangleq}\mathbb{K}\backslash\{k\}$) parameterized by $\bar{\omega}$. The probability $\mathbb{P}(k|\xi_{t-1}, \bar{\omega})$ is the posterior belief in agent i's intelligence level based on the joint trajectory history ξ_{t-1} upon time $t{-}1$, which can be updated recursively given an initial prior probability $\mathbb{P}(k)$:

$$\mathbb{P}(k|\xi_t,\bar{\omega}) = \frac{\pi_{\bar{\omega}}^{*,i,k}(s_t,a_t^i)\mathbb{P}(k|\xi_{t-1},\bar{\omega})}{\sum_{k'\in\mathbb{K}}\pi_{\bar{\omega}}^{*,i,k'}(s_t,a_t^i,)\mathbb{P}(k'|\xi_{t-1},\bar{\omega})}. \tag{11}$$

Intuitively, (10) and (11) mean that without prior knowledge on the intelligence level of each agent, we need to update the beliefs in the roles of both human agents simultaneously as we evaluate the likelihood of a joint trajectory, given parameter set $\bar{\omega}$.

From (10) and (11), we can see that the gradient $\partial\log\mathbb{P}(\xi|\bar{\omega})/\partial\bar{\omega}$ depends on two items (details are in Section 3 of the supplementary material): 1) the gradients of both agents' policies under arbitrary rationality level $k{\in}\mathbb{K}$ with respect to $\bar{\omega}$, *i.e.*, $\partial\pi_{\bar{\omega}}^{*,i,k}/\partial\bar{\omega}$, and 2) the gradient of the posterior belief of k with respect to $\bar{\omega}$, i.e., $\partial\log\mathbb{P}(k|\xi_{t-1},\bar{\omega})/\partial\bar{\omega}$.

4.2.1 Gradients of policies

Recalling (6), $\partial\pi_{\bar{\omega}}^{*,i,k}/\partial\bar{\omega}$, $\forall i{\in}\mathcal{P}$ and $k{\in}\mathbb{K}$, requires the gradient of the corresponding optimal Q function with respect to $\bar{\omega}$, i.e., $\partial Q_{\bar{\omega}}^{*,i,k}/\partial\bar{\omega}$. Due to the $\max$ operator in (5), direct differentiation is not feasible. Hence, we use a smooth approximation for the $\max$ function, that is, $\max(x_1,\cdots,x_{n_x}){\approx}\big(\sum_{i=1}^{n_x}(x_i)^\kappa\big)^{\frac{1}{\kappa}}$ with all $x_i{>}0$. The parameter $\kappa{>}0$ controls the approximation error, and when $\kappa{\to}\infty$, the approximation becomes exact. Therefore, (5) can be re-written as

$$V_{\bar{\omega}}^{*,i,k}(s) = \max_{a^i\in A^i}Q_{\bar{\omega}}^{*,i,k}(s,a^i) \approx \left(\sum_{a^i\in A^i}\left(Q_{\bar{\omega}}^{*,i,k}(s,a^i)\right)^\kappa\right)^{\frac{1}{\kappa}}. \tag{12}$$

Taking derivative of both sides of (12) with respect to $\bar{\omega}$ yields (note that $(\cdot)_{\bar{\omega}}^{'} := \frac{\partial(\cdot)_{\bar{\omega}}}{\partial\bar{\omega}}$)

$$V_{\bar{\omega}}^{',*,i,k}(s){\approx}\frac{1}{\kappa}\left(\sum_{a^i\in A^i}\left(Q_{\bar{\omega}}^{*,i,k}(s,a^i)\right)^\kappa\right)^{\frac{1-\kappa}{\kappa}}\sum_{a^i\in A^i}\left[\kappa\left(Q_{\bar{\omega}}^{*,i,k}(s,a^i)\right)^{\kappa-1}\cdot Q_{\bar{\omega}}^{',*,i,k}(s,a^i)\right], \tag{13a}$$

$$\begin{aligned} Q_{\bar{\omega}}^{',*,i,k}(s,a^i) = \sum_{a^{-i}\in A^{-i}}\Bigg(&\frac{\partial\rho_{\bar{\omega}}^i}{\partial\bar{\omega}}(c_{s,a^i}^{a^{-i}})\big(R_{\bar{\omega}}^i(s,a^i,a^{-i})+\tilde{\gamma}V_{\bar{\omega}}^{*,i,\mathrm{k}}(s')\big) \\ &+\rho_{\bar{\omega}}^i(c_{s,a^i}^{a^{-i}})\big(\frac{\partial R_{\bar{\omega}}^i}{\partial\bar{\omega}}(s,a^i,a^{-i})+\tilde{\gamma}V_{\bar{\omega}}^{',*,i,k}(s')\big)\Bigg). \end{aligned} \tag{13b}$$

We can see that in (13), $V_{\bar{\omega}}^{',*,i,k}$ is in a recursive format. Hence, we propose below a dynamic programming algorithm to solve for $V_{\bar{\omega}}^{',*,i,k}$ and $Q_{\bar{\omega}}^{',*,i,k}$ at all state and action pairs.

Theorem 2. *If the one-step reward R^i, $i\in\mathcal{P}$, is bounded by $R^i{\in}[R_{min},R_{max}]$ satisfying $\tilde{\gamma}\frac{R_{max}}{R_{min}}{<}1$, then $\partial V_{\bar{\omega}}^{*,i,k}/\partial\bar{\omega}$ can be found via the following value gradient iteration:*

$$V_{\bar{\omega},m+1}^{',i,k}(s) \approx \frac{1}{\kappa}\left(\sum_{a^i\in A^i}\left(Q_{\bar{\omega}}^{*,i,k}(s,a^i)\right)^\kappa\right)^{\frac{1-\kappa}{\kappa}}\sum_{a^i\in A^i}\left[\kappa\left(Q_{\bar{\omega}}^{*,i,k}(s,a^i)\right)^{\kappa-1}\cdot Q_{\bar{\omega},m}^{',i,k}(s,a^i)\right], \tag{14a}$$

$$\begin{aligned} Q_{\bar{\omega},m}^{',i,k}(s,a^i) = \sum_{a^{-i}\in A^{-i}}\Bigg(&\frac{\partial\rho_{\bar{\omega}}^i}{\partial\bar{\omega}}(c_{s,a^i}^{a^{-i}})\big(R^i(s,a^i,a^{-i})+\tilde{\gamma}V_(^{*,i,k}s')\big) \\ &+\rho_{\bar{\omega}}^i(c_{s,a^i}^{a^{-i}})\big(\frac{\partial R_{\bar{\omega}}^i}{\partial\bar{\omega}}(s,a^i,a^{-i})+\tilde{\gamma}V_{\bar{\omega},m}^{',i,k}(s')\big)\Bigg). \end{aligned} \tag{14b}$$

Moreover, the algorithm converges to $\partial V_{\bar{\omega}}^{*,i,k}/\partial\bar{\omega}$ as $m{\to}\infty$.

Proof. We first define $\nabla\mathcal{B}V_m^{',i,k} = V_{m+1}^{',i,k}$, and show that the operator $\nabla\mathcal{B}$ is a contraction under the given conditions (derivations of $\partial\rho_{\bar{\omega}}^i/\partial\bar{\omega}$ are shown in Section 3 of the supplementary material). Then, the statement is proved by induction similar to Theorem 1. More details are given in Section 4 of the supplementary material. ■

4.2.2 Gradient of the posterior belief

The second gradient that we need to compute is the gradient of the posterior belief of k with respect to $\bar{\omega}$, *i.e.*, $\partial\log\mathbb{P}(k|\xi_{t-1},\bar{\omega})/\partial\bar{\omega}$. Recalling the definition of the posterior belief over k in (11), we have $\partial\log\mathbb{P}(k|\xi_{t-1},\bar{\omega})/\partial\bar{\omega}$ depending on $\partial\pi_{\bar{\omega}}^{*,i,k}/\partial\bar{\omega}(s_{t-1},a_{t-1}^i)$ and $\partial\log\mathbb{P}(k|\xi_{t-2},\bar{\omega})/\partial\bar{\omega}$ for all $k{\in}\mathbb{K}$. Substituting the gradients of policies developed in Section 4.2.1 yields a recursive format from time 0 to time $t-1$ which can then be easily computed.

4.3 Parameter learning algorithm

We summarize the value iteration algorithm that computes the policy gradient in Algorithm 2. Based on this, the gradient ascent algorithm is used to find local optimal parameters $\bar{\omega}$ that maximize the log-likelihood of the demonstrated joint behaviors of agents in BRSMG. The algorithm is summarized in Algorithm 3.

Algorithm 2: Compute gradient of the bounded risk-sensitive policies

```
Input: Markov Game model $\mathcal{G}$, highest
  intelligence level $k_{\text{max}}$, and $\pi^{*,i,k}$, $i \in \mathcal{P}$
  and $k = 1, \ldots, k_{\text{max}}$.
Output:$\{\frac{\partial \pi_{\bar{\omega}}^{*,i,k}}{\partial \bar{\omega}}\}$, $i \in \mathcal{P}$ and $k \in \mathbb{K}$.
  Initialize $k = 1$;
while $k \leq k_{max}$ do
    for $i \in \mathcal{P}$ do
        Initialize $V_{\bar{\omega}}^{',i,k}(s), \forall s \in \mathcal{S}$;
        while $V^{',i,k}$ not converged do
            for $s \in \mathcal{S}$ do
                $V^{',i,k}(s) \leftarrow$
                  $\nabla \mathcal{B} V^{',i,k}(s)$;
            end for
        end while
        for $(s, a^i) \in \mathcal{S} \times A^i$ do
            Compute $\frac{\partial \pi_{\bar{\omega}}^{i,k}}{\partial \bar{\omega}}(s, a^i)$ by
              differentiating Eq. (6) with
              respect to $\omega$;
        end for
    end for
end while
Return $\{\frac{\partial \pi_{\bar{\omega}}^{i,k}}{\partial \bar{\omega}}\}$, $i \in \mathcal{P}$ and $k \in \mathbb{K}$.
```

5 Simulation

In this section, we utilize an indoor navigation example to verify the proposed algorithms in both the forward policy learning and inverse reward learning problems in BRSMG. The simulation setup is shown in Fig. 1. Two human agents in a room are required to exit the room through two different doors while avoiding the obstacles in the environment and potential collisions with each other. We assume that the two agents move simultaneously and there is no communication between them, but they can well observe the actions and states of each other at every time step.

5.1 Environment setup

We define the state as $s{=}(x^1, y^1, x^2, y^2)$, where x^i and y^i denote the coordinates of the human agent i, $i{\in}\mathcal{P}$. The two agents share a same action set $A{=}\{(a_x, a_y)\}$ with $a_x{\in}[-1, 0, 1]$ representing the steps to move along x direction, and similarly $a_y{\in}[-1, 0, 1]$ for the direction of y. For each agent, we do not allow it to move at two directions simultaneously. The state transition of the Markov Game is given by $x^i_{t+1}{=}x^i_t{+}a^{i,x}$ and $y^i_{t+1}{=}y^i_t{+}a^{i,y}$.

At each state, the reward of agent i includes two elements: a navigation reward and a safety reward which reflect, respectively, how close the agent is to the target door and how safe it is with respect to collisions with obstacles or the other agent. We restrict all rewards to be positive, satisfying $R_{\text{min}}{=}1$, namely, if a collision happens, an agent will collect only a fixed reward of 1. If there is no collision, then agents can receive rewards greater than 1 according to their distances to the doors. Such ground-truth navigation rewards are shown in Fig. 2 where the highest rewards will be collected if the agents reach the doors. In the forward policy learning problem, we set the parameters in the CPT model as $\gamma^{1,2}{=}0.5$ and $\alpha^{1,2}{=}0.7$. In the inverse reward learning problem, the CPT parameters reduce to $\gamma^{1,2}{=}0.5$ and $\alpha^{1,2}{=}1$ since only the influences from the decision weighting function will be considered. Moreover, in the inverse problem, we directly learn the values of the navigation rewards and the CPT parameter γ, *i.e.*, $\bar{\omega}{=}(\gamma, (\omega^1, \omega^2))$ with $\omega^i \in \mathbb{R}^{25}$ for $i{=}1, 2$.

5.2 Interactions in Bounded Risk-Sensitive Markov Games

In this section, we investigate the influence of the risk-sensitive performance measure to agents' policies in Markov Game by comparing the interactive behaviors of agents under risk-neutral and risk-sensitive policies. We consider three cases: Case 1 - both agents are level-1 (L1-L1); Case 2 - both agents are level-2 (L2-L2); and Case 3 - one agent is level-1 and the other is level-2 (L1-L2). If both agents exit the environment without collisions and dead-locks, we call it a success. We compare the rate of success (RS) of each case under risk-neutral and risk-sensitive policies in 100 simulations.

First, let us see how a risk-neural agent behaves under different levels of intelligence. Based on the anchoring policy (level-0) in Definition 2 in Section 3.1, a risk-neural level-1 agent will behave quite aggressively since it believes that the other agent is a pure-follower. On the contrary, a risk-neutral level-2 agent will perform more conservatively because it believes that the other agent is aggressively executing a level-1 policy. Indeed, as shown in Fig. 3(b), in Case 1, both agents behaved aggressively and leads to the lowest RS. Case 3 achieved the highest RS since the level-2 agent was aware of the aggressiveness of the level-1 agent which acted exactly as expected. The RS in Case 2 sits in the middle because though both agents behaved conservatively, the wrong beliefs over the other's model lead to higher collisions compared to Case 2.

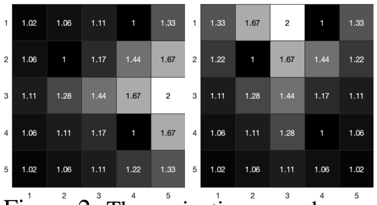
Next, we will see how the risk-sensitive CPT model impacts such risk-neutral behaviors. As shown in Fig. 3(b), in Case 1, the risk-sensitive policies help significantly improve the RS of interactions between two level-1 agents, *i.e.*, they performed less aggressively compared to the risk-neutral case. This is because that the CPT model makes the level-1 agents underestimate the possibilities of "yielding" from the pure-follower, and thus leads to more conservative behaviors with higher RS. Such observations can also be verified by the example trajectories shown in Fig. 3(a). We can see that compared to the risk-neutral case (dashed line), under the risk-sensitive policy, the blue agent decided to yield to the orange one at the fourth decision. At the same time, in Cases 2 and 3, the risk-sensitivity measure makes the level-2 agents overestimate the possibilities of "yielding" from level-1 agents and generate more aggressive behaviors. Hence, the RS for both Case 2 and Case 3 are reduced compared to the risk-neutral scenarios.

5.3 The Inverse Learning in Bounded Risk-Sensitive Markov Games

We randomized the initial conditions of the indoor navigation scenario and collected $M{=}100$ expert demonstrations to learn the rewards and parameters in the CPT risk model.

Algorithm 3: The inverse learning algorithm

```
Input: A demonstration set $\mathcal{D}$ and learning rate $\eta$
Output: Learned parameters $\bar{\omega}$.
Initialize $\bar{\omega}$.
while not converged do
    Run Algorithm 1, Algorithm 2
    Compute demonstration gradient following:
      $\nabla_{\bar{\omega}} = \sum_{\xi\in\mathcal{D}} \frac{\partial \log\left(\mathbb{P}(\xi|\bar{\omega})\right)}{\partial\bar{\omega}}$;
    Update the parameters following: $\bar{\omega} = \bar{\omega} + \eta\nabla_{\bar{\omega}}$;
end while
Return: $\bar{\omega}$
```



	1	2	3	4	5
1	1.02	1.06	1.11	1	1.33
2	1.06	1	1.17	1.44	1.67
3	1.11	1.28	1.44	1.67	2
4	1.06	1.11	1.17	1	1.67
5	1.02	1.06	1.11	1.22	1.33

	1	2	3	4	5
1	1.33	1.67	2	1	1.33
2	1.22	1	1.67	1.44	1.22
3	1.11	1.28	1.44	1.17	1.11
4	1.06	1.11	1.28	1	1.06
5	1.02	1.06	1.11	1.06	1.02

Figure 2: The navigation reward maps (left: agent 1; right: agent 2).

We evaluate the learning performance via two metrics: 1) the parameter percentage error (PPE), and 2) the policy loss (PL). The PPE of learned parameters $\bar{\omega}^i$ is defined as $|\bar{\omega}^i{-}\bar{\omega}^{*,i}|/|\bar{\omega}^{*,i}|$ with $\bar{\omega}^{*,i}$ being the ground-truth value. The PL is defined as $\frac{1}{|\mathbb{K}\times\mathcal{S}\times A^i|}\sum_{(k,s,a^i)\in\mathbb{K}\times\mathcal{S}\times A^i}|\pi^{*,i,k}_{\bar{\omega}}(s,a) - \pi^{*,i,k}_{\bar{\omega}^*}(s,a)|$ where $\pi^{*,i,k}_{\bar{\omega}}$ and $\pi^{*,i,k}_{\bar{\omega}^*}$ are, respectively, the policy of agent i under the learned parameter vector $\bar{\omega}$ and the true vector $\bar{\omega}^*$.

Figure 3(d) and Fig. 3(e) show, respectively, the history of averaged PPE over all parameters and the PL during learning. The solid lines represent the means from 25 trials and the shaded areas are the 95% confidence intervals. The PPEs of all parameters are given in Fig. 3(c). We can see that the proposed inverse learning algorithm can effectively recover both the rewards and the parameter γ in the CPT model for both agents, with all PPEs smaller than 15%.

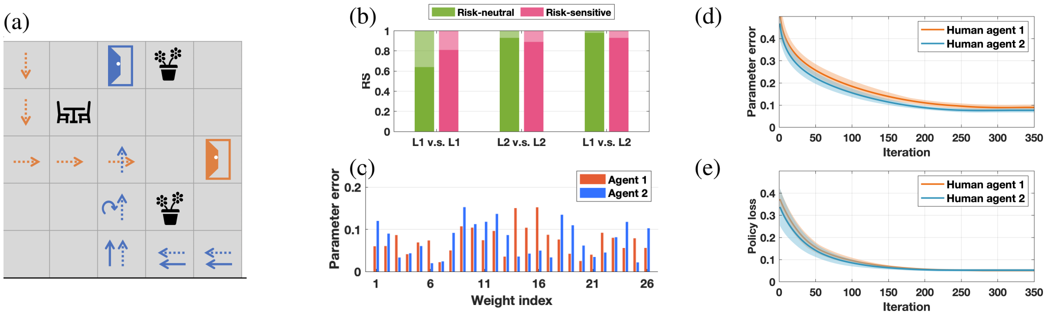


Figure 3: (a) An example pair of interactive trajectories between two level-1 agents under risk-neutral policies (dashed line) and risk-sensitive policies (solid line, circular arrow denotes the action "stay"). (b) Performance comparison between the bounded risk-neutral policies and the risk-sensitive policies. (c) PPE of learned parameters. (d-e) Averaged PPE and PL w.r.t. training epochs.

6 Conclusion

In this paper, we investigated a new type of Markov Game, *i.e.*, the bounded risk-sensitive Markov Game (BRSMG), and its influences on agents' interactive behaviors. Both the forward policy design and the inverse reward learning problems have been addressed with not only theoretical proofs but also simulation verification. We found that the risk-sensitive CPT measure made aggressive agents less aggressive, and conservative agents less conservative compared to the cases in classical Markov Games with expected utility theory. Such findings are important for us to establish a more realistic game-theoretic model to describe human behaviors in interactions, and build better human-centered intelligent agents that can understand and assist humans more efficiently.

7 Broader Impact

Humans always interact with each other and make decisions under uncertainties. To better assist humans, a better model to capture their interactive behaviors has attracted a great amount of research efforts in multiple disciplines such as psychology, cognitive science, economics and computer science. The proposed bounded risk-sensitive Markov Game (BRSMG) framework extends classical Markov Game to consider more realistic behaviors of humans such as bounded rationality and risk-sensitive tendency. We investigate how the optimal policies of the BRSMG differ from that of classical Games. Such a framework aims to provide a better modeling of humans' interactive behaviors, so that we can help them make better and more rational investment decisions in markets, we can design more human-like robots to assist humans in a more efficient and friendly manner, and we can help the governments to propose better guidance towards public policies (such as the "shelter-in-place" order and the re-opening guidance for the COVID-19 pandemic) by reasoning about the possible responses from people.

Similar to all other technologies, a better descriptive model for humans' interactive behaviors also face ethical risks. One possible such risk is that the proposed work might make humans more vulnerable to malicious attacks. For example, someone might be able to more accurately infer your personal preferences (rewards and risk models) by observing your behaviors and take advantage of it. Another risk is that though more human-like artificial agents are preferable in terms of efficiency and less tedious work for humans, they might cause more people lose their jobs in more societal sectors.

References

[1] Rabah Amir. Stochastic games in economics and related fields: an overview. In *Stochastic games and applications*, pages 455–470. Springer, 2003.

[2] Eli Ben-Sasson, Ehud Kalai, and Adam Kalai. An approach to bounded rationality. In *Advances in Neural Information Processing Systems*, pages 145–152, 2007.

[3] Lucian Bu, Robert Babu, Bart De Schutter, et al. A comprehensive survey of multiagent reinforcement learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38(2):156–172, 2008.

[4] Colin F Camerer, Teck-Hua Ho, and Juin-Kuan Chong. A cognitive hierarchy model of games. *The Quarterly Journal of Economics*, 119(3):861–898, 2004.

[5] Yinlam Chow, Aviv Tamar, Shie Mannor, and Marco Pavone. Risk-sensitive and robust decision-making: a cvar optimization approach. In *Advances in Neural Information Processing Systems*, pages 1522–1530, 2015.

[6] Miguel A. Costa-Gomes and Vincent P. Crawford. Cognition and behavior in two-person guessing games: An experimental study. *American Economic Review*, 96(5):1737–1768, Dec. 2006.

[7] Jaime F Fisac, Eli Bronstein, Elis Stefansson, Dorsa Sadigh, S Shankar Sastry, and Anca D Dragan. Hierarchical game-theoretic planning for autonomous vehicles. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 9590–9596. IEEE, 2019.

[8] Joseph Y Halpern. Beyond nash equilibrium: Solution concepts for the 21st century. In *Proceedings of the twenty-seventh ACM symposium on Principles of distributed computing*, pages 1–10, 2008.

[9] Joseph Y Halpern and Rafael Pass. Algorithmic rationality: Game theory with costly computation. *Journal of Economic Theory*, 156:246–268, 2015.

[10] Cheng Jie, LA Prashanth, Michael Fu, Steve Marcus, and Csaba Szepesvári. Stochastic optimization in a cumulative prospect theory framework. *IEEE Transactions on Automatic Control*, 63(9):2867–2882, 2018.

[11] Daniel Kahneman and Amos Tversky. Prospect theory: An analysis of decision under risk. In *Handbook of the fundamentals of financial decision making: Part I*, pages 99–127. World Scientific, 2013.

[12] Kun Lin and Steven I Marcus. Dynamic programming with non-convex risk-sensitive measures. In *2013 American Control Conference*, pages 6778–6783. IEEE, 2013.

[13] Anirudha Majumdar, Sumeet Singh, Ajay Mandlekar, and Marco Pavone. Risk-sensitive inverse reinforcement learning via coherent risk models. In *Robotics: Science and Systems*, 2017.

[14] Lillian J Ratliff and Eric Mazumdar. Inverse risk-sensitive reinforcement learning. *IEEE Transactions on Automatic Control*, 2019.

[15] Debajyoti Ray, Brooks King-Casas, P Read Montague, and Peter Dayan. Bayesian model of behaviour in economic games. In *Advances in neural information processing systems*, pages 1345–1352, 2009.

[16] Andrzej Ruszczyński. Risk-averse dynamic programming for markov decision processes. *Mathematical programming*, 125(2):235–261, 2010.

[17] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *Nature*, 550(7676):354–359, 2017.

[18] Herbert A Simon. From substantive to procedural rationality. In *25 years of economic theory*, pages 65–86. Springer, 1976.

[19] Dale O Stahl and Paul W Wilson. On players' models of other players: Theory and experimental evidence. *Games and Economic Behavior*, 10(1):218–254, 1995.

[20] Amos Tversky and Daniel Kahneman. Advances in prospect theory: Cumulative representation of uncertainty. *Journal of Risk and uncertainty*, 5(4):297–323, 1992.

[21] John Von Neumann and Oskar Morgenstern. *Theory of games and economic behavior (commemorative edition)*. Princeton university press, 2007.

[22] Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, and Anind K Dey. Maximum entropy inverse reinforcement learning. In *Aaai*, volume 8, pages 1433–1438. Chicago, IL, USA, 2008.

Supplementary Material

A Cumulative Prospect Theory

The cumulative prospect theory (CPT) is a non-expected utility theory that describes the risk-sensitivity of humans' decision-making processes. In this section, we illustrate the probability weighting function and the utility function, specifically when they are using the following functional forms:

$$w^+(p) = \frac{p^\gamma}{(p^\gamma + (1-p)^\gamma)^{1/\gamma}}, w^-(p) = \frac{p^\delta}{(p^\delta + (1-p)^\delta)^{1/\delta}}, \tag{1}$$

$$u(x) = \begin{cases} (x)^\alpha, & \text{if } x \geq 0, \\ \lambda(-x)^\beta, & \text{otherwise.} \end{cases} \tag{2}$$

In Fig. 4 (a), we show an example of the probability weighting function w^+, which describes the characteristics of humans to deflate high probabilities and inflate low probabilities. In Fig. 4 (b), we show an example of the utility function u with $x^0 = 0$ as the reference point.

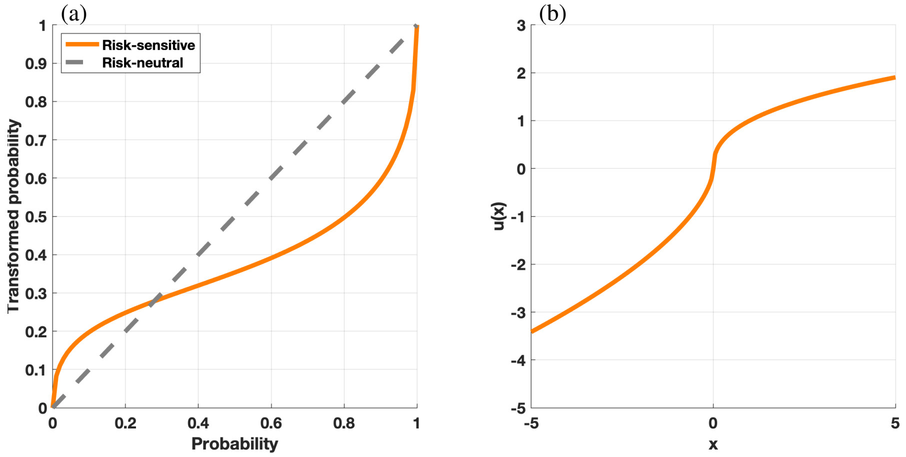


Figure 4: (a) Probability weighting function w^+ with $\gamma = 0.7$. (b) Utility function with $\alpha = 0.4$, $\beta = 0.6$, λ = -1.3.

B The convergence of bounded risk-sensitive policies

In this section, we show the proof of Theorem 1. To begin with, we show two lemmas that facilitate the proof.

Lemma 1. *If $a \geq 1$, $b \geq 1$, and $\alpha \in (0, 1]$, then $|a^\alpha - b^\alpha| \leq |a - b|$.*

Proof. First, it is clear that the above argument holds when $a = b$. Then, without loss of generality, we assume that $a > b$. We define a differentiable function $f(x) : \mathbb{R} \to \mathbb{R}$ and $f(x) = x^\alpha$. Then, following the mean value theorem, we can have $f(a) - f(b) = (a - b)f'(c)$, where $c \in (b, a)$. Note that $f'(c) = \alpha c^{\alpha-1} \leq 1$ since $\alpha \in (0, 1]$ and $c > 1$. Then we have $f(a) - f(b) \leq (a - b)$, and thus $a^\alpha - b^\alpha \leq a - b$ holds. Similarly, we have $b^\alpha - a^\alpha \leq b - a$ if $a < b$. ■

Lemma 2. *Assume that $\sum_{a^{-i}\in A^{-i}} \rho^i(c^{a^{-i}}_{s,a^i}) \leq 1$. Then, the Bellman operator $\mathcal{B}V^{i,k}_m(s) = \max_{a^i\in A^i} \sum_{a^{-i}\in A^{-i}} \rho^i(c^{a^{-i}}_{s,a^i}) u^i\big(R^i(s, a^i, a^{-i}) + \tilde{\gamma} V^{i,k}_m(s')\big)$ defined in (8) in Section 3.2 of the submitted manuscript is a $\tilde{\gamma}$-contraction mapping when R_{min} satisfies $R_{min} \geq 1$. That is, for any two value function estimates $V^{i,k}_1$ and $V^{i,k}_2$, we have*

$$\max_{s\in\mathcal{S}} \left| \mathcal{B}V^{i,k}_1(s) - \mathcal{B}V^{i,k}_2(s) \right| \leq \tilde{\gamma} \max_{s\in\mathcal{S}} \left| V^{i,k}_1(s) - V^{i,k}_2(s) \right|. \tag{3}$$

Proof. Define $r^i_{1,2}(c^{a^{-i}}_{s,a^i}) = R^i(s, a^i, a^{-i}) + \tilde{\gamma} V^{i,k}_{1,2}(s')$, then, we can write

$$\begin{aligned}
\left|\mathcal{B}V_1^{i,k}(s) - \mathcal{B}V_2^{i,k}(s)\right| &= \left| \max_{a^i \in A^i} \sum_{a^{-i} \in A^{-i}} \rho^i(c^{a^{-i}}_{s,a^i}) u^i\big(r^i(c^{a^{-i}}_{s,a^i})\big) - \max_{a^i \in A^i} \sum_{a^{-i} \in A^{-i}} \rho^i(c^{a^{-i}}_{s,a^i}) u^i\big(r^i(c^{a^{-i}}_{s,a^i})\big) \right| \\
&\leq \max_{a^i \in A^i} \left| \sum_{a^{-i} \in A^{-i}} \rho^i(c^{a^{-i}}_{s,a^i}) u^i(r^i(c^{a^{-i}}_{s,a^i})) - \sum_{a^{-i} \in A^{-i}} \rho^i(c^{a^{-i}}_{s,a^i}) u^i\big(r^i(c^{a^{-i}}_{s,a^i})\big) \right| \qquad \text{(4a)} \\
&\leq \max_{a^i \in A^i} \sum_{a^{-i} \in A^{-i}} \rho^i(c^{a^{-i}}_{s,a^i}) \left| u^i\big(r^i_1(c^{a^{-i}}_{s,a^i})\big) - u^i\big(r^i_2(c^{a^{-i}}_{s,a^i})\big) \right| \qquad \text{(4b)} \\
&\leq \max_{a^i \in A^i} \sum_{a^{-i} \in A^{-i}} \rho^i(c^{a^{-i}}_{s,a^i}) \left| r^i_1(c^{a^{-i}}_{s,a^i}) - r^i_2(c^{a^{-i}}_{s,a^i}) \right| \qquad \text{(4c)} \\
&\leq \max_{a^i \in A^i} \sum_{a^{-i} \in A^{-i}} \rho^i(c^{a^{-i}}_{s,a^i}) \left| \tilde{\gamma} V_1^{i,k}(s') - \tilde{\gamma} V_2^{i,k}(s') \right| \qquad \text{(4d)} \\
&= \max_{a^i \in A^i} \tilde{\gamma} \sum_{a^{-i} \in A^{-i}} \rho^i(c^{a^{-i}}_{s,a^i}) \left| V_1^{i,k}(s') - V_2^{i,k}(s') \right|. \qquad \text{(4e)}
\end{aligned}$$

Note that the inequality (4)(c) holds based on the definition of u^i defined in Section 2.3 of the submitted manuscript, namely, $u^i(x) = x^\alpha, x \geq 0, \alpha \in (0, 1]$. Therefore, we have $r^i_{(1,2)}(c^{a^{-i}}_{s,a^i}) = R^i(s, a^i, a^{-i,n}) + \tilde{\gamma} V^{i,k}_{(1,2)}(s') > R_{\min} \geq 1$. With Lemma 1, we have (4)(c). Hence,

$$\begin{aligned}
\max_{s \in \mathcal{S}} \left| \mathcal{B}V_1^{i,k}(s) - \mathcal{B}V_2^{i,k}(s) \right| &\leq \max_{s \in \mathcal{S}} \max_{a^i \in A^i} \tilde{\gamma} \sum_{a^{-i} \in A^{-i}} \rho^i(c^{a^{-i}}_{s,a^i}) \left| V_1^{i,k}(s') - V_2^{i,k}(s') \right| \\
&\leq \max_{s \in \mathcal{S}} \max_{a^i \in A^i} \tilde{\gamma} \sum_{a^{-i} \in A^{-i}} \rho^i(c^{a^{-i}}_{s,a^i}) \max_{s''} \left| V_1^{i,k}(s'') - V_2^{i,k}(s'') \right| \\
&= \tilde{\gamma} \max_{s''} \left| V_1^{i,k}(s'') - V_2^{i,k}(s'') \right| \max_{s \in \mathcal{S}} \max_{a^i \in A^i} \sum_{a^{-i} \in A^{-i}} \rho^i(c^{a^{-i}}_{s,a^i}) \\
&\leq \tilde{\gamma} \max_{s \in \mathcal{S}} \left| V_1^{i,k}(s) - V_2^{i,k}(s) \right|. \qquad \text{(5a)}
\end{aligned}$$

Note that the inequality (5a) holds since $\sum_{a^{-i} \in A^{-i}} \rho^i(c^{a^{-i}}_{s,a^i}) \leq 1$ as defined in the condition. Proceeding in this way, we conclude that the operator $\mathcal{B}$ is a $\tilde{\gamma}$-contraction mapping. ■

Now, we restate Theorem 1 in the submitted manuscript and show its proof.

Theorem 1. *Define the tuple* $\langle s, a^i, a^{-i} \rangle := c^{a^{-i}}_{s,a^i}$ *and normalize the transformed probabilities* $\tilde{\rho}^i(c^{a^{-i}}_{s,a^i}) := \tilde{\rho}^i(\mathbb{P}(a^{-i}|s, a^i))$ *by*

$$\rho^i(c^{a^{-i}}_{s,a^i}) = \begin{cases} \tilde{\rho}^i(c^{a^{-i}}_{s,a^i}) / \max_{a^i} \sum_{a^{-i}} \tilde{\rho}^i(c^{a^{-i}}_{s,a^i}), & \textit{if } k = 1, \\ \tilde{\rho}^i(c^{a^{-i}}_{s,a^i}) / \sum_{a^{-i}} \tilde{\rho}^i(c^{a^{-i}}_{s,a^i}), & \textit{otherwise}. \end{cases} \qquad (6)$$

For an arbitrary agent $i \in \mathcal{P}$*, if the one-step reward* R^i *is lower-bounded by* R_{min} *with* $R_{min} \geq 1$*, then* $\forall s \in \mathcal{S}$ *and all intelligence levels with* $k = 1, 2, \cdots$*, the dynamic programming problem in (5) of the submitted manuscript can be solved by the following value iteration algorithm:*

$$V^{i,k}_{m+1}(s) = \max_{a^i \in A^i} \sum\nolimits_{a^{-i} \in A^{-i}} \rho^i(c^{a^{-i}}_{s,a^i}) u^i\big(R^i(s, a^i, a^{-i}) + \tilde{\gamma} V^{i,k}_m(s')\big)\big), \quad s' = \mathcal{T}(s, a^i, a^{-i}). \qquad (7)$$

Moreover, as $m \to \infty$*,* $V^{i,k}_{m+1}$ *converges to the optimal value function* $V^{*,i,k}(s)$*.*

Proof. We prove the theorem by induction.

When k=1, $\pi^{*,-i,k-1}$=$\pi^{-i,0}$, which is defined in (3) (Definition 2) in the submitted manuscript. Hence, the dynamic programming problem defined in (5) in the submitted manuscript reduces to a single-agent policy optimization problem since the anchoring policy is known and (5) can be expressed as $V^{*,i,1}(s) = \mathcal{B}V^{*,i,1}(s)$. Moreover, (6) guarantees that the assumption $\sum_{a^{-i}\in\mathcal{A}^{-i}} \rho^i(c^{a^{-i}}_{s,a^i}) \leq 1$ in Lemma 2 holds. Hence, according to Lemma 2, we have

$$\begin{aligned}
\max_{s\in\mathcal{S}} \left| V^{i,1}_{m+1}(s) - V^{*,i,1}(s) \right| &= \max_{s\in\mathcal{S}} \left| \mathcal{B}V^{i,1}_{m}(s) - \mathcal{B}V^{*,i,1}(s) \right| \\
&\leq \tilde{\gamma} \max_{s\in\mathcal{S}} \left| V^{i,1}_{m}(s) - V^{*,i,1}(s) \right| \\
&= \tilde{\gamma} \max_{s\in\mathcal{S}} \left| \mathcal{B}V^{i,1}_{m-1}(s) - \mathcal{B}V^{*,i,1}(s) \right| \\
&\leq \tilde{\gamma}^2 \max_{s\in\mathcal{S}} \left| V^{i,1}_{m-1}(s) - V^{*,i,1}(s) \right| \\
&\;\;\vdots \\
&\leq \tilde{\gamma}^m \max_{s\in\mathcal{S}} \left| V^{i,1}_{1}(s) - V^{*,i,k}(s) \right|, \qquad (8)
\end{aligned}$$

and it is clear that $V^{i,1}_m \to V^{*,i,1}$ as $m \to \infty$. Hence, when $k = 1$, the algorithm in (7) can solve for the optimal CPT value and the policy $\pi^{*,i,1}$ can be obtained for all $i \in \mathcal{P}$. Note that $\pi^{*,i,1}$ depends on i's intelligence level.

Next, we will show that for any $k' \in \mathbb{N}^+$ and $k' > 1$, assuming the convergence of $V^{*,i,k'-1}$ is proved and the policy $\pi^{*,i,k'-1}$ is obtained for all $i \in \mathcal{P}$, then, similar to (8), we have $V^{i,k'}_m \to V^{*,i,k'}$ as $m \to \infty$.

Again, with the above assumption on $V^{*,i,k'-1}$ and $\pi^{*,i,k'-1}$, we can see that the dynamic programming problem defined in (5) in the submitted manuscript has been reduced to a single-agent optimal policy optimization problem since the opponent's policy $\pi^{*,-i,k'-1}$ is already solved for and thus only depends on agent $-i$'s intelligence level. Moreover, (6) assures that $\sum_{a^{-i}\in\mathcal{A}^{-i}} \rho^i(c^{a^{-i}}_{s,a^i})$=1 for $k' > 1$, satisfying the assumption in Lemma 2. Hence, via the conclusion from Lemma 2 and (8), we can see that $V^{*,i,k'}$ can be solved by the algorithm in (7). The policy $\pi^{*,i,k'}$ can also be obtained correspondingly for all $i \in \mathcal{P}$.

Hence, we have proved that argument in Theorem 1 holds. ■

C Detailed Derivations in the Inverse Learning Problem

In this section, we show some detailed derivations that needed to compute the gradient of the objective function in (9) in the submitted manuscript.

C.1 Derivation of the gradient of the log-likelihood of a demonstration

In this subsection, we show the derivation of the gradient of the log-likelihood of a demonstration. Recall (10) in the submitted manuscript, we can write

$$\frac{\partial \log\big(\mathbb{P}(\xi|\bar{\omega})\big)}{\partial\bar{\omega}} = \sum_{t=0}^{N-1} \frac{1}{\mathfrak{P}_t}\frac{\partial\mathfrak{P}_t}{\partial\bar{\omega}}, \qquad (9a)$$

$$\mathfrak{P}_t := \sum_{k\in\mathbb{K}} \pi^{*,i,k}_{\bar{\omega}}(s_t, a^i_t)\pi^{*,-i,-k}_{\bar{\omega}}(s_t, a^{-i}_t)\mathbb{P}(k|\xi_{t-1}, \bar{\omega}), \qquad (9b)$$

$$\begin{aligned}
\frac{\partial\mathfrak{P}_t}{\partial\bar{\omega}} = \sum_{k\in\mathbb{K}} \Big(&\frac{\partial\pi^{*,i,k}_{\bar{\omega}}}{\partial\bar{\omega}}(s_t, a^i_t)\pi^{*,-i,-k}_{\bar{\omega}}(s_t, a^{-i}_t)\mathbb{P}(k|\xi_{t-1}, \bar{\omega}) \\
&+ \pi^{*,i,k}_{\bar{\omega}}(s_t, a^i_t)\frac{\partial\pi^{*,-i,-k}_{\bar{\omega}}}{\partial\bar{\omega}}(s_t, a^{-i}_t)\mathbb{P}(k|\xi_{t-1}, \bar{\omega}) \\
&+ \pi^{*,i,k}_{\bar{\omega}}(s_t, a^i_t)\pi^{*,-i,-k}_{\bar{\omega}}(s_t, a^{-i}_t)\frac{\partial\mathbb{P}(k|\xi_{t-1}, \bar{\omega})}{\partial\bar{\omega}}\Big). \qquad (9c)
\end{aligned}$$

C.2 Supporting derivations for the gradient of policies in Section 4.2.1 in the submitted manuscript

In this subsection, we show the derivation of $\frac{\partial \rho^i_{\bar{\omega}}}{\partial \bar{\omega}}(c^{a^{-i}}_{s,a^i})$ which is required in Theorem 2 in the submitted manuscript to compute the gradient of policies.

Recall (7) in the submitted manuscript, we can compute $\frac{\partial \rho^i_{\bar{\omega}}}{\partial \bar{\omega}}(c^{a^{-i}}_{s,a^i})$ as follows:

$$\frac{\partial \rho^i_{\bar{\omega}}}{\partial \bar{\omega}}(c^{a^{-i}}_{s,a^i}) = \begin{cases} \frac{\frac{\partial \tilde{\rho}^i_{\bar{\omega}}}{\partial \bar{\omega}}(c^{a^{-i}}_{s,a^i}) \max_{a^i} \sum_{a^{-i}} \tilde{\rho}^i(c^{a^{-i}}_{s,a^i}) - \tilde{\rho}^i_{\bar{\omega}}(c^{a^{-i}}_{s,a^i}) \frac{\partial \max_{a^i} \sum_{a^{-i}} \tilde{\rho}^i(c^{a^{-i}}_{s,a^i})}{\partial \omega}}{\left(\max_{a^i} \sum_{a^{-i}} \tilde{\rho}^i(c^{a^{-i}}_{s,a^i})\right)^2}, & \text{if } k = 1, \\ \frac{\frac{\partial \tilde{\rho}^i_{\bar{\omega}}}{\partial \bar{\omega}}(c^{a^{-i}}_{s,a^i}) \sum_{a^{-i}} \tilde{\rho}^i(c^{a^{-i}}_{s,a^i}) - \tilde{\rho}^i_{\bar{\omega}}(c^{a^{-i}}_{s,a^i}) \frac{\partial \sum_{a^{-i}} \tilde{\rho}^i(c^{a^{-i}}_{s,a^i})}{\partial \omega}}{\left(\sum_{a^{-i}} \tilde{\rho}^i(c^{a^{-i}}_{s,a^i})\right)^2}, & \text{if } k > 1. \end{cases} \tag{10}$$

It can be observed that (10) only depends on $\frac{\partial \tilde{\rho}^i_{\bar{\omega}}}{\partial \bar{\omega}}(c^{a^{-i}}_{s,a^i})$, and the treatment for the $\max$ operator follows the smooth approximation method used in (12) in Section 4.2.1 of the submitted manuscript. Next, we will show how to compute $\frac{\partial \tilde{\rho}^i_{\bar{\omega}}}{\partial \bar{\omega}}(c^{a^{-i}}_{s,a^i})$.

Note that based on the CPT model defined in (2) in the submitted manuscript, $\tilde{\rho}^i_{\bar{\omega}}(c^{a^{-i}}_{s,a^i})$ is a transform of the probability that agent $-i$ takes the action a^{-i} given current state s (and the action a^i from agent i if $k = 1$, *i.e.*, $\pi^{*,-i,0}(s, a^{-i}, a^i)$, since the anchoring policy depends on the actions from both agents). Without loss of generality, we assume that all $N_A = |A^{-i}|$ utilities induced by agent $-i$'s possible actions are ordered in increasing order, *i.e.*, $0 \leq r^i(c^{a^{-i}_1}_{s,a^i}) \leq \cdots \leq r^i(c^{a^{-i}_{N_A}}_{s,a^i})$, where $r^i(c^{a^{-i}}_{s,a^i}) = u^i\big(R^i(s, a^i, a^{-i}) + \tilde{\gamma} V^{*,i,k}(s')\big)\big)$. Then recall (2b) in the submitted manuscript (since all rewards are positive), for any $g \in \{1, \ldots, N_A\}$, we define

$$p^1(c^{a^{-i}_g}_{s,a^i}) = \begin{cases} \sum_{j=g}^{N_A} \pi^{*,-i,k-1}_{\bar{\omega}}(s, a^{-i}_j, a^i), & k = 1 \\ \sum_{j=g}^{N_A} \pi^{*,-i,k-1}_{\bar{\omega}}(s, a^{-i}_j), & k > 1 \end{cases}, \tag{11a}$$

$$p^2(c^{a^{-i}_g}_{s,a^i}) = \begin{cases} \sum_{j=g+1}^{N_A} \pi^{*,-i,k-1}_{\bar{\omega}}(s, a^{-i}_j, a^{-i}_j), & k = 1 \\ \sum_{j=g+1}^{N_A} \pi^{*,-i,k-1}_{\bar{\omega}}(s, a^{-i}_j), & k > 1 \end{cases}, \tag{11b}$$

then we have $\tilde{\rho}^i_{\bar{\omega}}(c^{a^{-i}_g}_{s,a^i}) = w^{i,+}(p^1) - w^{i,+}(p^2)$.

Note that both $w^{i,+}$, p^1 and p^2 depend on the parameter γ^i since $\gamma^i \in \bar{\omega}$, but only p^1 and p^2 depend on the parameter $\bar{\omega}^{-\gamma^i}$ (note that $\bar{\omega}^{-\gamma^i} \triangleq \bar{\omega} \setminus \{\gamma^i\}$), thus we compute $\frac{\partial \tilde{\rho}^i_{\bar{\omega}}}{\partial \gamma^i}$ and $\frac{\partial \tilde{\rho}^{i,n}_{\bar{\omega}}}{\partial \bar{\omega}^{-\gamma^i}}$ separately:

$$\frac{\partial \tilde{\rho}^i_{\bar{\omega}}}{\partial \gamma^i}(c^{a^{-i}_g}_{s,a^i}) = \Phi^1 - \Phi^2, \tag{12a}$$

$$\begin{aligned} \Phi^j = {} & w^{i,+}(p^j)\Big(\log(p^j) + \frac{\gamma^i}{p^j}\frac{\partial p^j}{\partial \gamma^i}(c^{a^{-i}_g}_{s,a^i}) + \frac{\log\left((p^j)^{\gamma^i} + (1-p^j)^{\gamma^i}\right)}{(\gamma^i)^2} \\ & - \frac{(p^j)^{\gamma^i}\left(\log(p^j) + \frac{\gamma^i}{p^j}\frac{\partial p^j}{\partial \gamma^i}(c^{a^{-i}_g}_{s,a^i})\right) + (1-p^j)^{\gamma^i}\left(\log(1-p^j) - \frac{\gamma^i}{1-p^j}\frac{\partial p^j}{\partial \gamma^i}(c^{a^{-i}_g}_{s,a^i})\right)}{\gamma^i\left((p^j)^{\gamma^i} + (1-p^j)^{\gamma^i}\right)}\Big), \\ & j = 1, 2, \end{aligned} \tag{12b}$$

$$\frac{\partial \tilde{\rho}^{i,n}_{\bar{\omega}}}{\partial \bar{\omega}^{-\gamma^i}}(c^{a^{-i}_g}_{s,a^i}) = w'^{(+)}_p(p^1)\frac{\partial p^1}{\partial \bar{\omega}^{-\gamma^i}}(c^{a^{-i}_g}_{s,a^i}) - w'^{(+)}_p(p^2)\frac{\partial p^2}{\partial \bar{\omega}^{-\gamma^i}}(c^{a^{-i}_g}_{s,a^i}), \tag{12c}$$

where $\frac{\partial p^{(1,2)}}{\partial \bar{\omega}}(c^{a^{-i}_g}_{s,a^i}) = \begin{cases} \sum_{j=(g,g+1)}^{N_A} \frac{\partial \pi^{*,-i,0}_{\bar{\omega}}}{\partial \bar{\omega}}(s, a^{-i}_g, a^i), & k = 1 \\ \sum_{j=(g,g+1)}^{N_A} \frac{\partial \pi^{*,-i,k-1}_{\bar{\omega}}}{\partial \bar{\omega}}(s, a^{-i}_g), & k > 1 \end{cases}$, $w'^{(+)}_p$ is the partial derivative with respect to the variables p and can be computed straightforwardly based on the functional form $w^+(p)$ defined in Section 2.3 of the submitted manuscript (or (1) in this supplementary material). $\frac{\partial \pi^{*,-i,0}_{\bar{\omega}}}{\partial \bar{\omega}}$ can be computed straightforwardly based on the definition of the anchoring policy ((3) in the submitted manuscript). The item $\frac{\partial \pi_{\bar{\omega}}^{*,-i,k-1}}{\partial \bar{\omega}}$ for $k > 1$ is already known, since we compute $\frac{\partial \pi_{\bar{\omega}}^{*,i,k}}{\partial \bar{\omega}}$ iteratively and sequentially for $k = 1, 2, \ldots, k_{\text{max}}$ and for $i \in \mathcal{P}$ as shown in Algorithm 2 in the submitted manuscript.

D The Convergence of Value Gradient

In this section, we show the proof of Theorem 2. To begin with, we first restate Theorem 2 as follows:

Theorem 2. *If the one-step reward R^i, $i \in \mathcal{P}$, is bounded by $R^i {\in} [R_{min}, R_{max}]$ satisfying $\tilde{\gamma} \frac{R_{max}}{R_{min}} {<} 1$, then $\partial V_{\bar{\omega}}^{*,i,k} / \partial \bar{\omega}$ can be found via the following value gradient iteration:*

$$V_{\bar{\omega},m+1}^{\prime,i,k}(s) \approx \frac{1}{\kappa} \left(\sum\nolimits_{a^i \in A^i} \left(Q_{\bar{\omega}}^{*,i,k}(s, a^i) \right)^{\kappa} \right)^{\frac{1-\kappa}{\kappa}} \sum\nolimits_{a^i \in A^i} \left[\kappa \left(Q_{\bar{\omega}}^{*,i,k}(s, a^i) \right)^{\kappa - 1} \cdot Q_{\bar{\omega},m}^{\prime,i,k}(s, a^i) \right], \tag{13a}$$

$$\begin{aligned} Q_{\bar{\omega},m}^{\prime,i,k}(s, a^i) = \sum\nolimits_{a^{-i} \in A^{-i}} \Bigg(& \frac{\partial \rho_{\bar{\omega}}^i}{\partial \bar{\omega}} (c_{s,a^i}^{a^{-i}}) \big(R^i(s, a^i, a^{-i}) + \tilde{\gamma} V_{\bar{\omega}}^{*,i,k}(s') \big) \\ & + \rho_{\bar{\omega}}^i (c_{s,a^i}^{a^{-i}}) \big(\frac{\partial R_{\bar{\omega}}^i}{\partial \bar{\omega}} (s, a^i, a^{-i}) + \tilde{\gamma} V_{\bar{\omega},m}^{\prime,i,k}(s') \big) \Bigg). \end{aligned} \tag{13b}$$

Moreover, the algorithm converges to $\partial V_{\bar{\omega}}^{*,i,k} / \partial \bar{\omega}$ as $m {\to} \infty$.

To prove Theorem 2, we begin with several lemmas that facilitate the proof.

Lemma 3. *When $u^+(x) = x^\alpha$, $\mathbb{CPT}(\epsilon x) = u^+(\epsilon) \mathbb{CPT}(x), \forall \epsilon > 0, \forall x \geq 0$.*

Proof. Based on the definition of the $\mathbb{CPT}$ measure ((1) in the submitted manuscript), we can write (note that we only consider u^+ since we consider positive rewards)

$$\mathbb{CPT}(\epsilon x) = \int_0^\infty w^+ \left(\mathbb{P} \left(u^+(\epsilon x) > y \right) \right) dy = \int_0^\infty w^+ \left(\mathbb{P} \left(u^+(x) > \frac{y}{u^+(\epsilon)} \right) \right) dy. \tag{14}$$

We let $z := \frac{y}{u^+(\epsilon)}$, then we have $dy = u^+(\epsilon) dz$, and

$$\mathbb{CPT}(\epsilon x) = u^+(\epsilon) \int_0^\infty w^+ \left(\mathbb{P} \left(u^+(x) > z \right) \right) dz = u^+(\epsilon) \mathbb{CPT}(x). \tag{15}$$

■

Lemma 4. *For an arbitrary agent $i \in \mathcal{P}$, if i's one-step reward is lower-bounded by R_{min} and upper-bounded by R_{max}, then $\forall k \in \mathbb{N}^+$, we have $V_{max}^{i,k} \leq \frac{R_{max}}{R_{min}} V_{min}^{i,k}$.*

Proof. We define $\theta = \frac{R_{\text{max}}}{R_{\text{min}}}$, then according to (4) in the submitted manuscript, $V_{\text{max}}^{i,k}$ can only be achieved if agent i collects the maximum one-step reward at every step. Similarly, $V_{\text{min}}^{i,k}$ can only be achieved if agent i collects the minimum one-step reward at every step. Hence, we have

$$\begin{aligned} V_{\text{max}}^{i,k} &= \mathbb{CPT}_{\pi^{*,-i,k-1}} \left[R_{\text{max}} + \tilde{\gamma} \mathbb{CPT}_{\pi^{*,-i,k-1}} \left[R_{\text{max}} + \ldots \right] \right] \\ &= \mathbb{CPT}_{\pi^{*,-i,k-1}} \left[\theta R_{\text{min}} + \tilde{\gamma} \mathbb{CPT}_{\pi^{*,-i,k-1}} \left[\theta R_{\text{min}} + \ldots \right] \right]. \end{aligned} \tag{16}$$

Since $\mathbb{CPT}_{\pi^{*,-i,k-1}}\Big[\theta R_{\text{min}}\Big] = u^i(\theta)\mathbb{CPT}_{\pi^{*,-i,k-1}}\Big[R_{\text{min}}\Big] \leq \theta\mathbb{CPT}_{\pi^{*,-i,k-1}}\Big[R_{\text{min}}\Big]$ based on Lemma 3 and the fact that $u^i(\theta) = u^+(\theta) = \theta^\alpha \leq \theta$, then we can have

$$\begin{aligned}
V^{i,k}_{\text{max}} &= \mathbb{CPT}_{\pi^{*,-i,k-1}}\Bigg[\theta R_{\text{min}} + \tilde{\gamma}\mathbb{CPT}_{\pi^{*,-i,k-1}}\Big[\theta R_{\text{min}} + \dots\Big]\Bigg] \\
&\leq \mathbb{CPT}_{\pi^{*,-i,k-1}}\Bigg[\theta R_{\text{min}} + \theta\tilde{\gamma}\mathbb{CPT}_{\pi^{*,-i,k-1}}\Big[R_{\text{min}} + \dots\Big]\Bigg] \\
&\leq \theta\mathbb{CPT}_{\pi^{*,-i,k-1}}\Bigg[R_{\text{min}} + \tilde{\gamma}\mathbb{CPT}_{\pi^{*,-i,k-1}}\Big[R_{\text{min}} + \dots\Big]\Bigg] \\
&= \frac{R_{\text{max}}}{R_{\text{min}}} V^{i,k}_{\text{min}}. \qquad (17)
\end{aligned}$$

■

Lemma 5. *Recall* (13)*(a) in Theorem 2 in this supporting material, we define an operator* $\nabla\mathcal{B}V^{',i,k}_{m} = V^{',i,k}_{m+1}$, $\forall i, \in \mathcal{P}$, $\forall k \in \mathbb{N}^+$. *Then, the operator* $\nabla\mathcal{B}$ *is a* $\bar{\gamma}$*-contraction mapping if the one-step reward* R^i *is bounded by* $R^i \in [R_{min}, R_{max}]$ *satisfying* $\bar{\gamma} = \tilde{\gamma}\frac{R_{max}}{R_{min}} {<} 1$, *that is, for any value function gradient estimates* $V^{',i,k}_{\bar{\omega},1}$ *and* $V^{',i,k}_{\bar{\omega},2}$, *we have*

$$\max_{s\in\mathcal{S}} \Bigg| \nabla\mathcal{B}V^{',i,k}_{\bar{\omega},1}(s) - \nabla\mathcal{B}V^{',i,k}_{\bar{\omega},2}(s) \Bigg| \leq \bar{\gamma} \max_{s\in\mathcal{S}} \Bigg| V^{',i,k}_{\bar{\omega},1}(s) - V^{',i,k}_{\bar{\omega},2}(s) \Bigg|. \qquad (18)$$

Proof. Recall (13)(a) in this supporting material, we can write

$$\begin{aligned}
&\Bigg|\nabla\mathcal{B}V^{',i,k}_{\bar{\omega},1}(s) - \nabla\mathcal{B}V^{',i,k}_{\bar{\omega},2}(s)\Bigg| \\
&= \Bigg|\frac{1}{\kappa}\Bigg(\sum_{a^i\in A^i}\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa}\Bigg)^{\frac{1-\kappa}{\kappa}} \sum_{a^i\in A^i}\Bigg[\kappa\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa-1}\Big(Q^{',i,k}_{\bar{\omega},1}(s,a^i) - Q^{',i,k}_{\bar{\omega},2}(s,a^i)\Big)\Bigg]\Bigg| \\
&= \Bigg|\sum_{a^i\in A^i}\Bigg[\frac{1}{\kappa}\Bigg(\sum_{a^i\in A^i}\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa}\Bigg)^{\frac{1-\kappa}{\kappa}} \kappa\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa-1}\Big(Q^{',i,k}_{\bar{\omega},1}(s,a^i) - Q^{',i,k}_{\bar{\omega},2}(s,a^i)\Big)\Bigg]\Bigg| \\
&= \Bigg|\sum_{a^i\in A^i}\Bigg[\Bigg(\sum_{a^i\in A^i}\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa}\Bigg)^{\frac{1}{\kappa}} \frac{\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa-1}}{\sum_{a^i\in A^i}\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa}}\Big(Q^{',i,k}_{\bar{\omega},1}(s,a^i) - Q^{',i,k}_{\bar{\omega},2}(s,a^i)\Big)\Bigg]\Bigg| \\
&= \Bigg|\sum_{a^i\in A^i}\Bigg[\frac{V^{*,i,k}(s)\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa-1}}{\sum_{a^i\in A^i}\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa}}\Big(Q^{',i,k}_{\bar{\omega},1}(s,a^i) - Q^{',i,k}_{\bar{\omega},2}(s,a^i)\Big)\Bigg]\Bigg| \\
&\leq \sum_{a^i\in A^i}\Bigg[\frac{V^{*,i,k}(s)\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa-1}}{\sum_{a^i\in A^i}\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa}}\Bigg|Q^{',i,k}_{\bar{\omega},1}(s,a^i) - Q^{',i,k}_{\bar{\omega},2}(s,a^i)\Bigg|\Bigg]. \qquad (19)
\end{aligned}$$

Recall (13)(b) in this supplementary material, we can have

$$\left|Q'^{,i,k}_{\bar{\omega},1}(s,a^i) - Q'^{,i,k}_{\bar{\omega},2}(s,a^i)\right| = \left|\sum_{a^{-i}\in A^{-i}} \left(\rho^i_{\bar{\omega}}(c^{a^{-i}}_{s,a^i})\tilde{\gamma}\Big(V'^{,i,k}_{\bar{\omega},1}(s') - V'^{,i,k}_{\bar{\omega},2}(s')\Big)\right)\right|$$

$$\leq \tilde{\gamma} \sum_{a^{-i}\in A^{-i}} \left(\rho^i_{\bar{\omega}}(c^{a^{-i}}_{s,a^i})\left|V'^{,i,k}_{\bar{\omega},1}(s') - V'^{,i,k}_{\bar{\omega},2}(s')\right|\right) \tag{20a}$$

$$\leq \max_{s''\in\mathcal{S}} \tilde{\gamma}\left|V'^{,i,k}_{\bar{\omega},1}(s'') - V'^{,i,k}_{\bar{\omega},2}(s'')\right| \sum_{a^{-i}\in A^{-i}} \rho^i_{\bar{\omega}}(c^{a^{-i}}_{s,a^i}) \tag{20b}$$

$$\leq \max_{s''\in\mathcal{S}} \tilde{\gamma}\left|V'^{,i,k}_{\bar{\omega},1}(s'') - V'^{,i,k}_{\bar{\omega},2}(s'')\right|. \tag{20c}$$

Note that the inequality (20)(c) holds since $\sum_{a^{-i}\in A^{-i}} \rho^i_{\bar{\omega}}(c^{a^{-i}}_{s,a^i}) \leq 1$, which is governed by (7) in the submitted manuscript. We substitute (20) into (19), then we have

$$\begin{aligned}\left|\nabla\mathcal{B}V'^{,i,k}_{\bar{\omega},1}(s) - \nabla\mathcal{B}V'^{,i,k}_{\bar{\omega},2}(s)\right| &\leq \sum_{a^i\in A^i}\left[\frac{V^{*,i,k}(s)\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa-1}}{\sum_{a^i\in A^i}\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa}} \max_{s''\in\mathcal{S}}\tilde{\gamma}\left|V'^{,i,k}_{\bar{\omega},1}(s'') - V'^{,i,k}_{\bar{\omega},2}(s'')\right|\right] \\ &= \max_{s''\in\mathcal{S}}\tilde{\gamma}\left|V'^{,i,k}_{\bar{\omega},1}(s'') - V'^{,i,k}_{\bar{\omega},2}(s'')\right| \sum_{a^i\in A^i}\frac{V^{*,i,k}(s)\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa-1}}{\sum_{a^i\in A^i}\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa}}.\end{aligned} \tag{21}$$

Also note that

$$\sum_{a^i\in A^i}\frac{V^{*,i,k}(s)\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa-1}}{\sum_{a^i\in A^i}\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa}} \leq \sum_{a^i\in A^i}\frac{V^{*,i,k}_{\text{max}}\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa-1}}{\sum_{a^i\in A^i}\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa}} \tag{22a}$$

$$\leq \frac{\sum_{a^i\in A^i}\frac{R_{\text{max}}}{R_{\text{min}}}V^{*,i,k}_{\text{min}}\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa-1}}{\sum_{a^i\in A^i}\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa}} \tag{22b}$$

$$= \frac{R_{\text{max}}}{R_{\text{min}}}\frac{\sum_{a^i\in A^i}V^{*,i,k}_{\text{min}}\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa-1}}{\sum_{a^i\in A^i}\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa}} \tag{22c}$$

$$\leq \frac{R_{\text{max}}}{R_{\text{min}}}. \tag{22d}$$

Note that the inequality (22) (b) holds based on Lemma 4. Moreover, inequality (22)(d) holds since

$$\sum_{a^i\in A^i}V^{*,i,k}_{\text{min}}\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa-1} - \sum_{a^i\in A^i}\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa}$$

$$= \sum_{a^i\in A^i}\Big(Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big)^{\kappa-1}\Big(V^{*,i,k}_{\text{min}} - Q^{*,i,k}_{\bar{\omega}}(s,a^i)\Big) \tag{23}$$

$$\leq 0. \tag{24}$$

Now, we substitute (22) into (21), and then we can write

$$\begin{aligned}\max_{s\in\mathcal{S}}\left|\nabla\mathcal{B}V'^{,i,k}_{\bar{\omega},1}(s) - \nabla\mathcal{B}V'^{,i,k}_{\bar{\omega},2}(s)\right| &\leq \max_{s''\in\mathcal{S}}\tilde{\gamma}\left|V'^{,i,k}_{\bar{\omega},1}(s'') - V'^{,i,k}_{\bar{\omega},2}(s'')\right| \max_{s\in\mathcal{S}}\frac{R_{\text{max}}}{R_{\text{min}}} \\ &= \frac{R_{\text{max}}}{R_{\text{min}}}\tilde{\gamma}\max_{s\in\mathcal{S}}\left|V'^{,i,k}_{\bar{\omega},1}(s) - V'^{,i,k}_{\bar{\omega},2}(s)\right|.\end{aligned} \tag{25}$$

Proceeding in this way, we conclude that the operator $\nabla\mathcal{B}$ is a $\bar{\gamma}$-contraction mapping, where $\bar{\gamma}{=}\tilde{\gamma}\frac{R_{\text{max}}}{R_{\text{min}}}{<}1$. ■

Now, we show the proof of Theorem 2.

Proof. We first define $\nabla\mathcal{B}V_m^{',i,k} = V_{m+1}^{',i,k}$, and then Lemma 5 shows that the operator $\nabla\mathcal{B}$ is a contraction under the given conditions. Then, the statement is proved by induction in a similar way as for Theorem 1, and thus is omitted. ■

Remark 1. *In the inverse reward learning problem, the u^+ function is reduced to identify function, but Lemma 5 and Theorem 2 both hold for the general form $u^+(x) = x^\alpha$ with $\alpha \in (0, 1]$.*