Modeling Human Behavior Without Humans – Bringing Prospect Theory to Multi-Agent Reinforcement Learning

Khush Gupta, Sheyan Lalmohammed, Alok Shah
University of Pennsylvania

Abstract

We apply CPT-MADDPG, a novel extension of the Multi-Agent Deep Deterministic Policy Gradient algorithm that embeds full Cumulative Prospect Theory (CPT) value and probability weighting transforms into both critic and actor updates. By replacing expected return maximization with rank-dependent Choquet integrals over gains and losses, CPT-MADDPG endows agents with tunable risk profiles—ranging from exploratory, risk-seeking to conservative, loss-averse behaviors—without human intervention. We further propose two extensions: an *observability adjustment*, which aggregates cross-agent subjective utilities in the Bellman backup when agents share CPT parameters, and *adaptive behavioral parameters*, where CPT hyperparameters are learned online via a secondary loss. Across competitive pursuit (Simple Tag), cooperative coverage (Simple Spread), and strategic bidding (first-price auctions), we show that risk-seeking parameterized CPT speeds early learning, extreme risk-averse parameterized CPT enforces prudence at a performance cost, transparent utility sharing preserves coordination under heterogeneity, and naive dynamic adaptation destabilizes convergence. In auction settings, learned CPT policies replicate the overbidding phenomenon documented by Josheski and Delcev, yielding short-term gains followed by long-term losses. Our work demonstrates a principled, differentiable framework for integrating human-like risk attitudes into multi-agent RL.

1 Introduction

Multi-agent reinforcement learning (MARL) has achieved remarkable success in domains ranging from autonomous driving to strategic game playing by training agents to maximize expected cumulative rewards Lowe et al. [2017]. Yet, such agents implicitly assume classical rationality, neglecting systematic human decision biases under risk. Decades of behavioral economics research have shown that real humans deviate from expected-utility theory in predictable ways—exhibiting loss aversion, reference dependence, and probability weighting—captured by Prospect Theory [Kahneman and Tversky, 1979] and its extension, Cumulative Prospect Theory (CPT) [Tversky and Kahneman, 1992].

In this work, we bridge the gap between rational MARL agents and human-like risk-sensitive behavior by embedding full CPT value and probability transformations into the Multi-Agent Deep Deterministic Policy Gradient (MADDPG) framework. Unlike prior risk-sensitive RL approaches that focus on single-agent settings, CPT-MADDPG applies rank-dependent weighting directly to cumulative returns in both critic and actor updates, enabling agents to exhibit calibrated risk-seeking or risk-averse behaviors without humans in the loop.

We further explore two novel extensions:

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

- **Observability Adjustment:** Allowing agents to access each other's subjective CPT-adjusted utilities, and deriving a cross-agent valuation aggregation that modifies the Bellman backup.
- **Adaptive Behavioral Parameters:** Treating CPT parameters (α, β, λ) as learnable variables, optimized alongside network weights to adapt risk profiles dynamically during training.

We evaluate CPT-MADDPG in two multi-partical environments (MPE's): Lowe et al. [2017] environments: competitive Simple Tag, cooperative Simple Spread, and a first-price auction with mixed CPT and non-CPT bidders. Our experiments demonstrate that (1) moderate risk-seeking parameterized CPT values yield exploratory, risk-seeking dynamics, (2) extreme risk-averse parameters induces conservative, low-variance strategies, (3) transparency of utilities from rewards preserves coordination, and (4) adaptive behavioral parameter dynamics can destabilize learning if updated too frequently.

Our contributions are as follows:

1. We apply CPT-MADDPG, integrating full CPT value and probability transforms into a multi-agent actor–critic algorithm.
2. We derive and implement observability-adjusted CPT updates that aggregate cross-agent utilities in the critic target.
3. We propose a secondary optimization of CPT hyperparameters to enable adaptive risk profiling during training.
4. We provide extensive empirical validation across competitive, cooperative, and auction tasks, highlighting the behavioral and performance trade-offs of CPT integration.

Despite its promise, naively inserting CPT into multi-agent actor–critic frameworks poses several challenges. First, the non-linear Choquet integrals in CPT introduce nonconvexity into the objective, destabilizing standard gradient updates. Second, the probability-weighting step requires empirical estimation of tail probabilities over returns, demanding careful batch-based approximations to avoid bias. Third, heterogeneous risk profiles across agents can yield non-stationary dynamics, complicating convergence. In this paper, we address these issues by (1) designing a minibatch-based CPT integral approximation that is fully differentiable, (2) integrating rank-dependent weighting inside the MADDPG critic and actor updates to maintain stability, and (3) extending the approach with observability and adaptive-parameter modules that we show preserve coordination and control non-stationarity. Our results across competitive pursuit, cooperative landmark coverage, and first-price auctions demonstrate how human-like risk biases can be systematically tuned to enhance exploration, enforce safety, or predictably modulate strategic behavior in richly interactive settings.

2 Related Work

Our work builds on four strands of literature: the foundations of Prospect Theory and its cumulative extension, risk-sensitive reinforcement learning, recent efforts to bring human-like risk preferences into multi-agent settings, and existing multi-agent actor critic methods in the context of risk-sensitive learning.

Prospect Theory and Cumulative Prospect Theory. Prospect Theory (PT) was introduced by Kahneman and Tversky [Kahneman and Tversky, 1979] to explain systematic deviations from expected-utility theory, notably loss aversion, reference dependence, and probability weighting. Cumulative Prospect Theory (CPT) extends PT to multi-outcome gambles by applying rank-dependent weighting to cumulative probabilities, which corrects several anomalies of the original formulation and enables tractable aggregation of outcomes [Tversky and Kahneman, 1992].

Risk-Sensitive Reinforcement Learning. In single-agent RL, risk-sensitive objectives have been studied extensively. Conditional Value at Risk (CVaR) criteria have been incorporated into MDPs to control downside risk [Bäuerle and Ott, 2014, Tamar et al., 2015]. Utility-based approaches, leveraging exponential or power utilities, provide an alternate route for encoding risk attitudes [García and Fernández, 2015]. These frameworks demonstrate that modifying the reward aggregation can systematically steer agent behavior toward risk-averse or risk-seeking policies.

Multi-Agent Risk-Aware Learning and CPT Integration. Extending risk sensitivity to multi-agent environments yields additional complexity due to strategic interactions. CVaR-based objectives have been applied in cooperative MARL tasks to mitigate collective downside [Fan et al., 2019], and entropy-regularized actor–critic methods have been proposed for risk-sensitive policy updates [Nachum et al., 2017]. Chen et al. [Chen et al., 2020] analyze network-aggregative games under risk awareness. Efforts to incorporate CPT directly into multi-agent actor–critic frameworks have recently emerged demonstrating that rank-dependent value transformations can induce human-like biases in both competitive and cooperative scenarios [Ewerhart and Leisen, 2010].

Multi-Agent Actor–Critic Methods Lowe et al. [Lowe et al., 2017] introduce Multi-Agent Deep Deterministic Policy Gradient (MADDPG), an actor–critic framework tailored for mixed cooperative–competitive environments. By employing centralized critics with access to all agents' observations and decentralized actors for scalable execution, MADDPG achieves stabilized learning and effective coordination. This paradigm directly informs our CPT-MADDPG design, where CPT-driven value transformations are embedded within each agent's critic update to capture risk preferences across strategic interactions. Building on the MADDPG paradigm, Lepel and Barakat Lepel and Barakat [2024] propose CPT-MADDPG, which embeds cumulative prospect theoretic transformations into each agent's critic and policy updates. By incorporating full Choquet integrals over gains and losses and learnable probability-weighting functions, CPT-MADDPG endows agents with tunable risk profiles—ranging from risk-averse to risk-seeking—while preserving the centralized training, decentralized execution framework. Their empirical results on continuous control benchmarks demonstrate improved performance under risk-sensitive objectives and offer insights into the computational trade-offs of CPT-driven multi-agent learning.

3 Preliminaries

Simple Tag (Competitive Multi-agent Particle Environment) The Simple Tag environment is a predator-prey scenario within the Multi-agent Particle Environment (MPE) framework. The setting involves $N_p = 1$ predator agents (adversaries) attempting to capture a single prey agent within a bounded two-dimensional arena populated with stationary obstacles. At each discrete timestep t, each predator i selects an action $a_t^i \in \mathbb{R}^2$ based on observations o_t^i of relative agent positions and velocities. Similarly, the prey agent independently selects actions aimed at evasion. The environment dynamics are deterministic given these actions, with added small Gaussian noise.

Rewards are structured as follows:

$$r_t^i = \begin{cases} +10, & \text{if predator } i \text{ successfully tags the prey at time } t, \\ 0, & \text{otherwise,} \end{cases} \qquad r_t^{\text{prey}} = \begin{cases} -10, & \text{if prey is tagged at time } t, \\ 0, & \text{otherwise.} \end{cases}$$

Additionally, to discourage escape from the bounded area, the prey receives a continuous penalty as it approaches the boundaries, defined by:

$$\text{bound}(x) = \begin{cases} 0, & x < 0.9, \\ 10(x - 0.9), & 0.9 \le x < 1.0, \\ \min(\exp(2x - 2), 10), & x \ge 1.0. \end{cases}$$

Agents receive observations comprising their own velocity and position, relative positions to obstacles, and relative positions and velocities of other agents.

Simple Spread (Cooperative MPE) In Simple Spread, N agents must cover M fixed landmarks. Each agent's observation o_t^i includes its own position and those of landmarks and other agents. At each step, agent i receives

$$r_t^i = \underbrace{\sum_{m=1}^{M} \mathbb{I}[\|x_t^i - \ell_m\| < d_{\text{cov}}]}_{\text{covered landmarks}} - \underbrace{\sum_{j \neq i} \mathbb{I}[\|x_t^i - x_t^j\| < d_{\text{coll}}]}_{\text{collision penalty}}.$$

Here d_{cov} is the coverage radius and d_{coll} the collision threshold. All agents share identical reward functions to encourage cooperation.

First-Price Auction In the first-price auction, each of N agents receives a private valuation $v_i \sim \text{Uniform}(0, 100)$. Agents simultaneously submit bids $b_i \in [0, 100]$. The highest bidder wins and pays their bid; all others pay nothing. Agent i's reward is

$$r_i = \begin{cases} v_i - b_i, & \text{if } b_i = \max_j b_j \text{ (tie-broken uniformly)}, \\ 0, & \text{otherwise.} \end{cases}$$

In competitive mode, each agent maximizes its own payoff; in cooperative mode, the group reward is $\sum_i r_i$ and is shared equally.

Markov Decision Processes (MDPs) We formulate each learning problem as a Markov Decision Process (MDP), defined by the tuple $(\mathcal{S}, \mathcal{A}, P, R, \gamma)$, where:

- $\mathcal{S}$ is the state space, with $s_t \in \mathcal{S}$ denoting the environment state at time t.
- $\mathcal{A}$ is the action space, with $a_t \in \mathcal{A}$ representing the agent's decision.
- $P(s_{t+1} \mid s_t, a_t)$ is the transition probability kernel.
- $R(s_t, a_t)$ is the (possibly stochastic) immediate reward received after taking action a_t in state s_t.
- $\gamma \in [0, 1)$ is the discount factor weighting future rewards.

A (stochastic) policy $\pi_\theta(a \mid s)$, parameterized by θ, induces trajectories $(s_0, a_0, r_0, \dots)$ whose return is $G_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k}$. The goal is to find θ^* maximizing the value function $J(\theta) = \mathbb{E}\pi\theta[G_0]$.

Policy Gradient Algorithms Policy Gradient methods directly optimize the policy parameters by estimating $\nabla_\theta J(\theta)$ and performing gradient ascent:

$$J(\theta) = \mathbb{E}_{\tau \sim p_\theta(\tau)}\big[R(\tau)\big] = \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)\Big],$$

where $\tau = (s_0, a_0, s_1, \dots)$, $p_\theta(\tau) = \rho_0(s_0) \prod_t \pi_\theta(a_t \mid s_t) P(s_{t+1} \mid s_t, a_t)$, and $0 < \gamma < 1$ is the discount factor.

The Policy Gradient Theorem states

$$\nabla_\theta J(\theta) = \mathbb{E}_{s \sim d^\pi,\, a \sim \pi_\theta}\big[\nabla_\theta \log \pi_\theta(a \mid s)\, Q^\pi(s, a)\big],$$

where $d^\pi(s) \propto \sum_{t=0}^{\infty} \gamma^t P(s_t = s \mid \pi)$ is the discounted state-visitation distribution and

$$Q^\pi(s, a) = \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \mid s_0 = s, a_0 = a\Big].$$

To reduce variance, one replaces $Q^\pi(s, a)$ with the advantage function

$$A^\pi(s, a) = Q^\pi(s, a) - V^\pi(s), \quad V^\pi(s) = \mathbb{E}_{a \sim \pi_\theta}\big[Q^\pi(s, a)\big].$$

Actor–Critic methods maintain

- an *actor* $\pi_\theta(a \mid s)$, updated via $\theta \leftarrow \theta + \alpha\, \mathbb{E}[\nabla_\theta \log \pi_\theta(a \mid s)\, A^\pi(s, a)]$,
- a *critic*, $V_w(s)$ or $Q_w(s, a)$, trained (e.g. by temporal-difference learning) to approximate V^π or Q^π.

This coupling yields low-variance, on-policy gradient estimates while retaining exploration.

Multi-Agent Deep Deterministic Policy Gradient (MADDPG) The Multi-Agent Deep Deterministic Policy Gradient (MADDPG) Lowe et al. [2017] algorithm extends the Deep Deterministic Policy Gradient (DDPG) framework to multi-agent settings, particularly those involving mixed cooperative-competitive interactions. A core tenet of MADDPG is *centralized training with decentralized execution*. During execution, each agent i acts based on its own local observation o_i using its actor policy $\mu_i : \mathcal{O}_i \to \mathcal{A}_i$, parameterized by θ_i^μ, which outputs a deterministic action $a_i = \mu_i(o_i|\theta_i^\mu)$.

For training, MADDPG introduces a separate centralized critic $Q_i(x, a_1, \ldots, a_N|\theta_i^Q)$ for each agent i. This critic is parameterized by θ_i^Q and takes as input some representation of the global state x (e.g., the concatenation of all agents' observations $(o_1, \ldots, o_N)$ and potentially other state information) and the actions of *all* N agents $a_1, \ldots, a_N$. It outputs an estimate of the expected return for agent i. The critic Q_i for each agent i is updated by minimizing the loss:

$$\mathcal{L}(\theta_i^Q) = \mathbb{E}_{(x,\mathbf{a},\mathbf{r},x')\sim\mathcal{D}}\left[\left(Q_i(x, a_1, \ldots, a_N|\theta_i^Q) - y_i\right)^2\right], \tag{1}$$

where $\mathbf{a} = (a_1, \ldots, a_N)$, $\mathbf{r} = (r_1, \ldots, r_N)$, and the target value y_i is computed as:

$$y_i = r_i + \gamma Q_i'(x', a_1', \ldots, a_N'|\theta_i^{Q'})\big|_{a_j'=\mu_j'(o_j'|\theta_j^{\mu'})}. \tag{2}$$

Here, $\mathcal{D}$ is an experience replay buffer storing tuples $(x, \mathbf{a}, \mathbf{r}, x')$. Q_i' and μ_j' are target networks with parameters $\theta_i^{Q'}$ and $\theta_j^{\mu'}$, which are typically updated via soft updates (Polyak averaging) from their respective online network parameters.

The actor policy μ_i for each agent i is updated using the deterministic policy gradient, derived from the expected return $J(\theta_i^\mu) = \mathbb{E}[R_i]$:

$$\nabla_{\theta_i^\mu} J(\theta_i^\mu) = \mathbb{E}_{x,\mathbf{a}\sim\mathcal{D}}\left[\nabla_{\theta_i^\mu}\mu_i(o_i|\theta_i^\mu)\nabla_{a_i}Q_i(x, a_1, \ldots, a_N|\theta_i^Q)\big|_{a_i=\mu_i(o_i|\theta_i^\mu)}\right]. \tag{3}$$

By conditioning the critic on the actions of all agents, the environment becomes stationary from the perspective of each agent's learning process, even as other agents' policies change. The use of separate critics for each agent allows MADDPG to be applied in scenarios with differing reward functions, including competitive or mixed settings. Optionally, if true policies of other agents are unknown during training, they can be inferred from observations.

4 Methods

4.1 Problem Formulation

We consider a multi-agent system modeled as an N-agent Markov game, defined by the tuple $(\mathcal{S}, \{\mathcal{A}_i\}_{i=1}^N, P, \{r_i\}_{i=1}^N, \gamma)$. Here, $\mathcal{S}$ represents the global state space, $\mathcal{A}_i$ is the action space for agent i, $P(s' \mid s, \mathbf{a})$ is the state transition probability kernel given the current state $s \in \mathcal{S}$ and joint action $\mathbf{a} = (a_1, \ldots, a_N)$ where $a_i \in \mathcal{A}_i$. The function $r_i(s, \mathbf{a})$ denotes the reward received by agent i, and $\gamma \in [0, 1)$ is the discount factor. Each agent i employs a stochastic policy $\pi_{\theta_i}(a_i \mid o_i)$, parameterized by θ_i, which maps its local observation o_i (derived from s) to a distribution over its actions. In standard Multi-Agent Reinforcement Learning (MARL), the objective for each agent i is to maximize its expected cumulative discounted return:

$$J(\theta_i) = \mathbb{E}_{\tau\sim\Pi_\theta}\left[\sum_{t=0}^{T}\gamma^t r_i(s_t, \mathbf{a}_t)\right], \tag{4}$$

where $\tau = (s_0, \mathbf{a}_0, r_0, \ldots, s_T, \mathbf{a}_T, r_T)$ is a trajectory of states, joint actions, and rewards, and $\Pi_\theta = \prod_{j=1}^N \pi_{\theta_j}$ is the joint policy of all agents.

4.2 Cumulative Prospect Theory Adjustments

To model decision-making biases and risk attitudes observed in human behavior, we incorporate Cumulative Prospect Theory (CPT) Tversky and Kahneman [1992] into the agent's objective. Instead

of directly maximizing the expected return $R_i = \sum_{t=0}^{T} \gamma^t r_i(s_t, \mathbf{a}_t)$, agents aim to maximize the CPT value of this return. The CPT value functional, $C(R_i)$, is defined by Choquet integrals over gains and losses relative to a reference point (assumed to be zero in this work):

$$C(R_i) = \int_0^\infty w^+\big(P(u(R_i) > z)\big)\, dz \; + \; \int_{-\infty}^0 w^-\big(P(u(R_i) > z) - 1\big)\, dz, \tag{5}$$

where $u(x)$ is a subjective utility function that captures diminishing sensitivity and loss aversion. We use the common power-law utility function:

$$u(x) = \begin{cases} x^\alpha, & \text{if } x \geq 0, \\ -\lambda(-x)^\beta, & \text{if } x < 0. \end{cases} \tag{6}$$

Here, $\alpha, \beta \in (0, 1]$ are parameters governing the curvature for gains and losses, respectively, and $\lambda \geq 1$ is the loss aversion coefficient. The functions $w^+(p)$ and $w^-(p)$ are non-linear probability weighting functions that transform objective probabilities p into subjective decision weights. These functions typically exhibit an inverse S-shape, overweighting small probabilities and underweighting large probabilities. In our implementation, we approximate $w^\pm$ using a 6-segment piecewise-linear function, with parameters $L^\pm$ fitted to a stretched-sigmoid form for efficient gradient estimation during training. (Note, in our results, a crude linear version of the probability weighting function is used for the MPE tasks rather than the approximation detailed above. However, in reviewing the results with the improved approximation, there is no change to the observed results of behavior.) The approximation is implemented by our w_approx(L, p) function.

The CPT-adjusted objective for agent i thus becomes:

$$J_{\mathrm{CPT}}(\theta_i) = \mathbb{E}_{\tau \sim \Pi_\theta}\left[C(R_i)\right]. \tag{7}$$

4.3 Approximation of the CPT Integral

Directly computing the CPT value functional $C(R_i)$ as defined by the Choquet integrals (Eq. 5) is challenging in an RL setting. The probability distribution of the cumulative return, $P(u(R_i) > z)$, is induced by the agent's evolving policy π_{θ_i} and the complex environment dynamics, and is typically not available in closed form. Estimating these probabilities and computing the integrals for every state-action pair or for every sampled trajectory during training would be computationally prohibitive.

To make CPT practical within our RL framework, we employ an approximation based on empirical estimates from a batch of sampled trajectory returns. Given a batch of B trajectories sampled from the replay buffer, yielding cumulative returns $\{R_i^{(k)}\}_{k=1}^B$ for agent i, the CPT integral $C(R_i)$ is approximated via the compute_cpt_integral‘ function as follows:

1. **Utility Transformation**: Each sampled cumulative return $R_i^{(k)}$ is first transformed into its subjective utility $u_k = u(R_i^{(k)})$ using the utility function defined in Eq. 6.
2. **Empirical Probability Estimation**: The probabilities $P(u(R_i) > z)$ for gains and $P(u(R_i) \leq z)$ for losses (or equivalently, $P(-u(R_i) \geq -z)$ for $z < 0$) are estimated empirically from the batch of B transformed utilities $\{u_k\}_{k=1}^B$. For a given threshold z, the empirical estimate of the tail probability for gains is $\hat{P}(u(R_i) > z) \approx \frac{1}{B} \sum_{k=1}^B \mathbb{I}(u_k > z)$, where $\mathbb{I}(\cdot)$ is the indicator function. A similar estimation is performed for losses.
3. **Piecewise Linear Weighting Function Application**: The probability weighting functions $w^+(p)$ and $w^-(p)$ are approximated using pre-defined piecewise linear functions, implemented by our w_approx(L, p) function.
4. **Numerical Integration via Summation**: The CPT value is then computed by numerically approximating the Choquet integrals. This involves sorting the unique observed utilities $\{u_k\}$ (along with 0) to define integration segments. For gains ($z \geq 0$), the integral is approximated as $\sum_j w^+\left(\hat{P}(u(R_i) > z_j)\right)(z_{j+1} - z_j)$, where z_j are the sorted unique positive utility values from the batch. A corresponding summation is performed for losses ($z < 0$) using $w^-(\hat{P}(u(R_i) \leq z_j))$ and the differences between sorted unique negative utility values. This process yields an estimate $\hat{C}(R_i)$ of the CPT value based on the empirical distribution of utilities from the sampled batch.

Algorithm 1 CPT-MADDPG Algorithm

Initialize actors μ_{θ_i}, critics Q_{ϕ_i}, and target networks
Initialize replay buffer $\mathcal{D}$
for episode = $1, \ldots, M$ **do**
    Collect trajectory using policies $\{\mu_{\theta_i}\}$ and store in $\mathcal{D}$
    for update step = $1, \ldots, K$ **do**
        Sample minibatch from $\mathcal{D}$
        Compute CPT values $C(R_i)$ using `compute_cpt_integral`
        Update critics $\phi_i \leftarrow \phi_i - \eta_\phi \nabla_{\phi_i} L(\phi_i)$
        Update actors $\theta_i \leftarrow \theta_i - \eta_\theta \nabla_{\theta_i} J_{\mathrm{CPT}}(\theta_i)$
        Soft-update target networks
    end for
end for

This batch-based empirical approximation allows for an estimate of $C(R_i)$ and, crucially, its gradient with respect to the input returns $R_i^{(k)}$, rendering it amenable for integration into gradient-based RL algorithms. While this approach relies on the representativeness of the sampled batch, it provides a tractable method for incorporating CPT-based risk preferences.

4.4 CPT-MADDPG Algorithm

We propose CPT-MADDPG, an extension of the Multi-Agent Deep Deterministic Policy Gradient (MADDPG) algorithm [Lowe et al., 2017], to incorporate CPT-based risk preferences. MADDPG employs a centralized training with decentralized execution paradigm. Each agent i learns an actor policy $\mu_{\theta_i}(o_i)$ that maps its local observation o_i to a deterministic action a_i, and a centralized critic $Q_{\phi_i}(s, \mathbf{a})$ that estimates the value of the joint action $\mathbf{a}$ in state s.

In CPT-MADDPG, the critic $Q_{\phi_i}(s, \mathbf{a})$ is trained to estimate the standard expected cumulative return. The actor, however, is updated to maximize a CPT-informed objective. The critic $Q_{\phi_i}(s, \mathbf{a})$ for agent i, parameterized by ϕ_i, is updated by minimizing the standard TD-error loss:

$$L(\phi_i) = \mathbb{E}_{(s,\mathbf{a},\mathbf{r},s')\sim\mathcal{D}} \left[\left(Q_{\phi_i}(s, \mathbf{a}) - y_i \right)^2 \right], \tag{8}$$

where $\mathcal{D}$ is a replay buffer, and the target value y_i is:

$$y_i = r_i + \gamma Q'_{\bar{\phi}_i}(s', \mathbf{a}') \Big|_{\mathbf{a}'_j = \mu'_{\bar{\theta}_j}(o'_j)}, \tag{9}$$

with $Q'_{\bar{\phi}_i}$ and $\mu'_{\bar{\theta}_j}$ being target networks with slowly updated parameters $\bar{\phi}_i$ and $\bar{\theta}_j$.

The actor $\mu_{\theta_i}(o_i)$ for agent i, parameterized by θ_i, is updated using a modified policy gradient. We introduce a CPT-based scaling factor, Φ_i, derived from the CPT value of empirically observed terminal returns. Specifically, from a batch of experiences, we identify trajectories that terminate at the next state s'. Let $\{R_{i,\text{final}}^{(k)}\}_{k=1}^{B'}$ be the set of cumulative episode rewards for these B' terminal trajectories. The CPT scaling factor is then $\Phi_i = \hat{C}(\{R_{i,\text{final}}^{(k)}\})$, computed using our compute_cpt_integral function on these terminal returns. The actor loss is then formulated to incorporate this CPT sensitivity:

$$L(\theta_i) = -\mathbb{E}_{s\sim\mathcal{D}, \mathbf{a}_{-i}\sim\mu_{\theta_{-i}}} \left[\exp(\nu \cdot \Phi_i) \cdot Q_{\phi_i}(s, \mu_{\theta_i}(o_i), \mathbf{a}_{-i}) \right], \tag{10}$$

where ν is a hyperparameter controlling the influence of the CPT scaling. The gradient is thus:

$$\nabla_{\theta_i} J_{\mathrm{CPT}}(\theta_i) \approx \mathbb{E}_{s\sim\mathcal{D}, \mathbf{a}_{-i}\sim\mu_{\theta_{-i}}} \left[\exp(\nu \cdot \Phi_i) \cdot \nabla_{\theta_i} \mu_{\theta_i}(o_i) \nabla_{a_i} Q_{\phi_i}(s, \mathbf{a}) \big|_{a_i = \mu_{\theta_i}(o_i)} \right]. \tag{11}$$

This formulation implies that actions leading to higher (CPT-valued) terminal returns are more strongly reinforced if $\Phi_i > 0$, and actions leading to lower CPT-valued terminal returns are less reinforced or more strongly penalized if $\Phi_i < 0$. If insufficient terminal returns are available in a batch to reliably compute Φ_i, it defaults to a neutral value (e.g., $\Phi_i = 0$, resulting in $\exp(0) = 1$). The overall algorithm is presented in Algorithm 1.

4.5 Observability-Adjusted CPT Transformation

When an agent is granted access to other agents' utility functions, we replace the single-agent CPT transform $C(R)$ with a cross-agent aggregation. Let $\mathcal{A}$ be the set of agents whose parameters are visible, and for each agent $j \in \mathcal{A}$ let

$$u_j^+(x) = x^{\alpha_j}, \qquad u_j^-(x) = \lambda_j\,(-x)^{\alpha_j},$$

and define constant weights $w'_{j,+}, w'_{j,-}$. For a (flattened) return R, we compute

$$\phi_{\text{cross}}(R) = \frac{1}{|\mathcal{A}|} \sum_{j \in \mathcal{A}} \begin{cases} w'_{j,+}\, u_j^+(R), & R \geq 0, \\ -\,w'_{j,-}\, u_j^-(R), & R < 0. \end{cases}$$

This replaces the usual CPT wrapper in the critic target:

$$y_i \;=\; u_i(r_i) \;+\; \gamma\, \mathbb{E}_{\mathbf{a}' \sim \mu_{\bar{\theta}}}\big[\phi_{\text{cross}}(R'_i)\big]\,.$$

By averaging each visible agent's subjective valuation, the update incorporates observed risk biases directly into the Bellman backup.

4.6 Adaptive Behavioral Parameter Dynamics

To allow each agent's CPT parameters to evolve during training, we parameterize α_i, λ_i, γ_i^+, and γ_i^- as learnable variables and optimize them via a secondary loss. Let $\Theta_i = \{\alpha_i, \lambda_i, \gamma_i^+, \gamma_i^-\}$ and write the usual CPT-adjusted target for agent i as

$$y_i^{\text{CPT}} \;=\; u_i(r_i) \;+\; \gamma\, \mathbb{E}_{\mathbf{a}' \sim \mu_{\bar{\theta}}}\big[C_{\Theta_i}(R'_i)\big],$$

and the standard Bellman target as y_i. We define the *base loss* for the adaptive parameters as

$$L_b^{(t)} = \mathbb{E}_{(s,\mathbf{a},r,s') \sim \mathcal{D}}\big[\,\big(y_i^{\text{CPT}} - y_i\big)^2\big].$$

To make the parameter updates sensitive to recent changes in this loss, we introduce a *dynamic scaling factor*

$$d^{(t)} = 1 + \big|\, L_b^{(t)} - L_b^{(t-1)}\big|.$$

We also include an ℓ_2 *regularization* term that penalizes deviation from the initial parameter values $\Theta_{i,0}$:

$$L_r = \sum_{p \in \Theta_i} (p - p_{i,0})^2.$$

Putting these together, the total loss for the adaptive parameters at iteration t is

$$L_{\text{adapt}}^{(t)} = d^{(t)}\, \kappa\, L_b^{(t)} \;+\; \rho\, L_r,$$

where $\kappa = 10^{-3}$ (scale_factor) and $\rho = 10^{-3}$ (reg_lambda). We then update Θ_i by gradient descent:

$$\Theta_i \;\leftarrow\; \Theta_i - \eta_{\text{adapt}}\, \nabla_{\Theta_i} L_{\text{adapt}}^{(t)}.$$

In practice, we freeze these updates for the first 20 iterations and then apply them every 10 iterations thereafter. This scheme allows the CPT parameters to adapt to the evolving reward landscape while avoiding instability from overly rapid changes.

4.7 Implementation Details

We implement CPT-MADDPG in PyTorch using the TorchRL and Vmas libraries. Each actor and critic is a 3-layer MLP (hidden sizes 128–128), with ReLU activations and tanh outputs on actions. Key hyperparameters are:

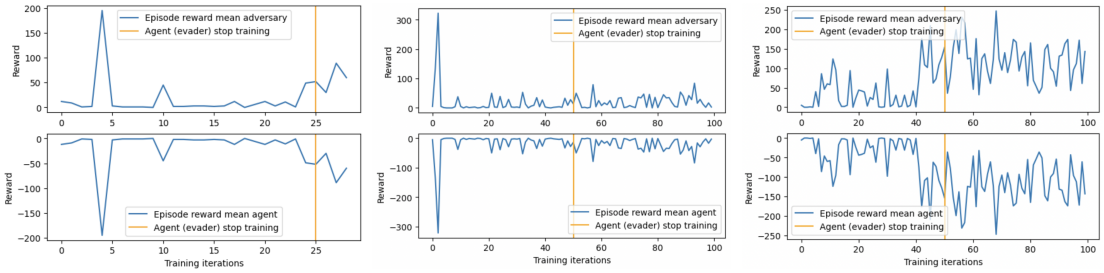
$$\eta_\theta = 1 \times 10^{-4},\;\; \eta_\phi = 1 \times 10^{-3},\;\; \gamma = 0.99,\;\; \tau = 0.01,\;\; \alpha = \beta = 0.88,\;\; \lambda = 2.25$$

CPT components (`u_plus`, `u_minus`, `w_approx`) and the `compute_cpt_integral` routine are implemented as differentiable PyTorch modules, enabling end-to-end training with Adam.

5 Results

5.1 Competitive Environment: Simple Tag

In the Simple Tag predator–prey scenario, our goal is to evaluate how wrapping cumulative returns in CPT transforms influences pursuit strategies in a continuous 2D action space; by comparing moderate (risk-seeking) and extreme (risk-averse) CPT settings against the risk-neutral baseline, we gain insight into how risk preferences trade off exploration versus safety. Figure 1 shows that moderate CPT induces intermittent spikes in episodic predator reward—reflecting willingness to risk zero payoff for potential large gains—whereas extreme CPT dramatically suppresses variance, delays convergence, and lowers mean reward relative to baseline, illustrating pronounced loss aversion and diminished risk tolerance.

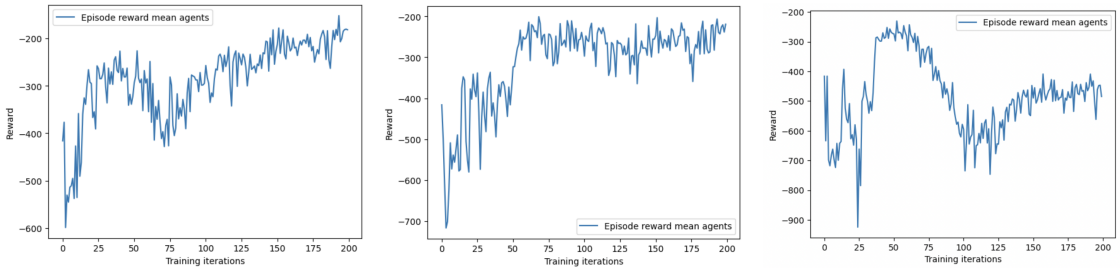


(a) Baseline MADDPG (b) Moderate CPT (c) Extreme CPT

Figure 1: Predator average episodic rewards in Simple Tag under baseline, moderate, and extreme CPT.

5.2 Cooperative Environment: Simple Spread

In the Simple Spread cooperative landmark-coverage task, we investigate whether CPT-based risk sensitivity can accelerate coordination without sacrificing stability; by applying moderate and extreme CPT to joint rewards, we gain an understanding of how agents hedge collision risk against coverage gains. As depicted in Figure 2, moderate (risk-seeking) CPT hyperparameter choices accelerate early convergence—agents explore varied positions to balance coverage and collision avoidance—yet stabilizes at similar asymptotic coverage to the baseline, while extreme CPT's strong loss-aversion leads to overly cautious movements and a large reduction in final coverage, highlighting the performance cost of excessive loss sensitivity.



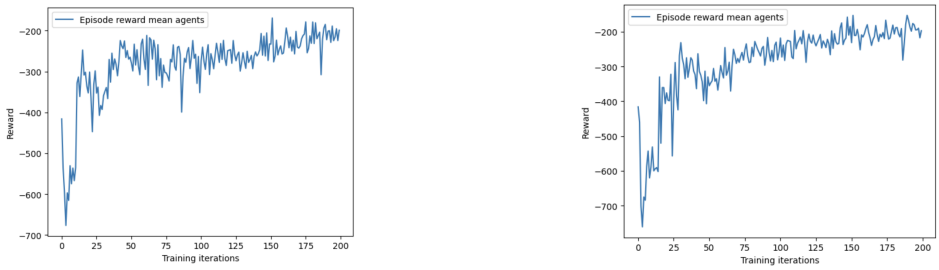
(a) Baseline (b) Moderate CPT (c) Extreme CPT

Figure 2: Landmark coverage rewards in Simple Spread under baseline, moderate, and extreme CPT.

5.3 Transparent Utility Sharing

Extending Simple Spread with full visibility of peers' CPT utilities, we ask whether transparency of subjective evaluations aligns expectations and preserves cooperative equilibria; this allows us to gain insight into the interplay between heterogeneous risk profiles under shared information. Figure 3 shows that both purely risk-averse pairs and mixed risk-averse/risk-seeking teams converge to the

same landmark-coverage trajectory as in opaque training, indicating that observing each other's utility functions mitigates strategic uncertainty without disrupting coordination.

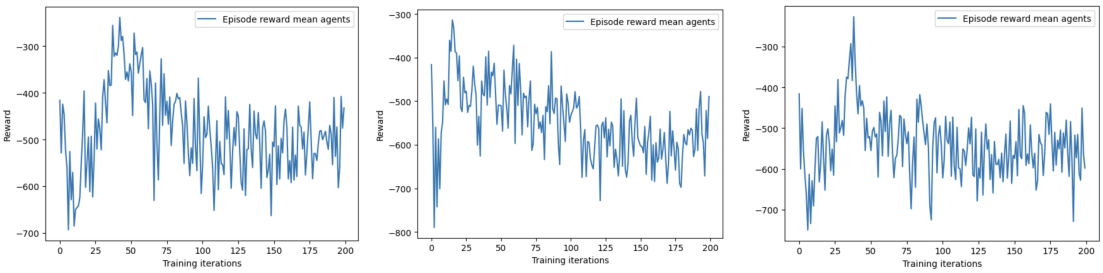


(a) Risk-Averse Seeing (b) Mixed Risk Profiles Seeing

Figure 3: Coverage rewards when agents observe each other's CPT utilities.

5.4 Dynamic CPT Parameters

As an extension to the ability of beeing able to observe their cooperative agents utility, we enable agents to update their CPT hyperparameters every ten episodes in the Simple Spread task, aiming to learn whether dynamic profiling can improve performance or introduce instability. Figure 4 illustrates that dynamic risk-seeking, moderate, and high-aversion schedules all produce large oscillations in coverage reward and fail to converge—reward variance exceeds the baseline by over 50%—demonstrating that rapidly shifting risk parameters destabilize multi-agent learning.



(a) Dynamic Risk Seeking (b) Dynamic Moderate CPT (c) Dynamic Extreme CPT

Figure 4: Reward trajectories under dynamic CPT hyperparameter scheduling.

5.5 First-Price Auction

Building on the empirical findings of Josheski and Delcev [Josheski and Delcev, 2022], who demonstrate that CPT-modeled bidders systematically overbid in first-price auctions, we evaluate whether our CPT-MADDPG agents replicate this behavior and its payoff consequences. Each agent's private valuation v_i is drawn uniformly from $[0, 100]$, and we compare three CPT-trained bidders against three risk-neutral agents.

Figure 5 overlays the empirical bid distributions after convergence. Consistent with Josheski and Delcev's results, the CPT histogram is clearly shifted to the right: modal bids for CPT agents lie around the maximum possible, whereas non-CPT bids cluster either between the lower end and the upper end of the possibel bid spectrum. This shift visually confirms the overbidding effect driven by loss aversion and probability weighting under CPT.

Figure 6 shows the average reward over iterations. CPT agents begin with higher payoffs—reflecting frequent wins from aggressive bids—but rewards decline below zero over time as the cost of overbidding outweighs gains. This turnaround closely matches the long-run loss effects described in Josheski and Delcev, illustrating the CPT trade-off between short-term advantage and eventual negative returns.

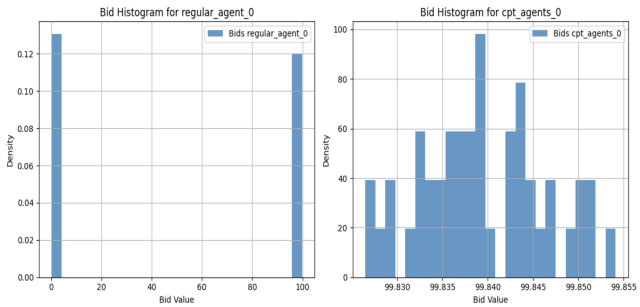


Figure 5: Empirical bid distributions for CPT vs. non-CPT agents (valuations uniform in [0,100]).

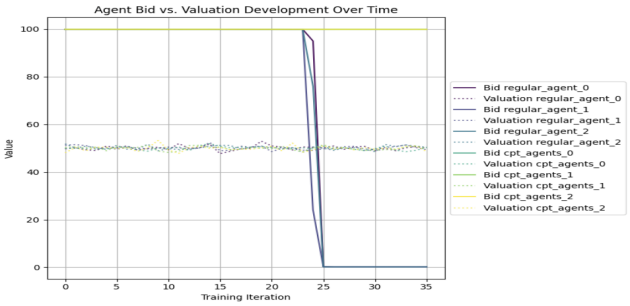


Figure 6: Average episodic reward trajectories for CPT vs. non-CPT agents in the first-price auction.

5.6 Summary of Findings

Across competitive, cooperative, transparency, dynamic, and auction settings, CPT-MADDPG produces rich, parameter-dependent behaviors: moderate (risk-seeking) CPT enhances exploratory learning, extreme (risk-averse) CPT enforces prudence at a performance cost, shared utilities preserve coordination despite risk heterogeneity, and adaptive parameter updates destabilize convergence. Auction results validate that CPT imparts human-like risk biases, granting strategic advantage through overbidding. Detailed hyperparameters for each variant are listed in Table 1 (Appendix A).

6 Discussion

Our empirical evaluation of CPT-MADDPG across competitive, cooperative, and auction domains demonstrates that embedding human-like risk preferences into multi-agent learning yields rich and interpretable behavioral variations. In Simple Tag, risk-seeking CPT drives predators to adopt more aggressive, exploratory tactics—risk-seeking "spikes" in reward—whereas risk-averse driven CPT produces conservative strategies marked by delayed convergence and lower overall payoff. In Simple Spread, moderate risk sensitivity accelerates early coordination at the cost of increased fluctuation, while extreme loss aversion impairs final coverage due to overly cautious movement. Allowing agents to observe each other's CPT utility rewards, assuming that they would adjust their own outcomes in the context of the rewards, shows that transparency of subjective evaluations preserves cooperative equilibria even under heterogeneous risk profiles. Conversely, dynamically adapting CPT parameters on the fly destabilizes learning, suggesting that introducing non-stationarity in risk attitudes undermines policy convergence. Finally, in first-price auctions, CPT-trained bidders systematically overbid—right-shifted bid distributions and an initial reward advantage followed by long-term losses— replicating the expected overbidding phenomenon documented by Josheski and Delcev [Josheski and Delcev, 2022].

These results highlight several key insights. First, rank-dependent probability weighting and loss aversion can be effectively integrated into actor–critic updates, endowing agents with tunable risk profiles that mirror human decision biases. Second, while moderate levels of CPT hyperparameters in a risk-seeking setting can enhance exploration and initial learning speed, extreme aversion or overly frequent adaptation of risk parameters imposes tangible performance penalties. Third, transparency

of risk preferences between agents need not harm coordination; indeed, shared utility information can align expectations and stabilize behavior. Fourth, CPT-induced overbidding confers short-term auction success but risks eventual negative returns, illustrating the classic "prospect" trade-off in learned policies.

However, our approach has limitations. The empirical approximation of the CPT integral relies on batch-based estimates of tail probabilities, which may introduce bias when returns are sparse or highly skewed. Dynamic hyperparameter adaptation, while conceptually appealing, proved difficult to stabilize and would benefit from more principled schedules or meta-learning frameworks. There may also be an alternative form of providing this hyperparameter adaptation in settings with asymmetric or perfect information. Computational overhead from computing CPT integrals in large multi-agent systems remains non-trivial, suggesting the need for more efficient approximation or hierarchical risk modeling.

Looking forward, several avenues for future work arise. Extending CPT-MADDPG to high-dimensional, continuous control tasks and real-world domains (e.g., autonomous driving or energy management) could validate scalability and practical utility. Incorporating theory-guided meta-learning to tune CPT parameters online may overcome the instability of naïve dynamic schedules. Finally, integrating CPT-based agents with large language models or other human-interactive systems offers a promising path toward more psychologically realistic and interpretable AI agents, capable of anticipating and adapting to human risk behavior in complex multi-agent environments.

In sum, CPT-MADDPG offers a principled framework for endowing reinforcement learners with human-aligned risk attitudes, opening new opportunities for interpretable, risk-aware multi-agent systems that bridge the gap between rational optimization and realistic decision-making under uncertainty.

References

Nicole Bäuerle and Johanna Ott. Markov decision processes with average value-at-risk criteria. *Mathematical Methods of Operations Research*, 80(2):281–298, 2014.

Liyun Chen, Liping Wang, Satinder Singh, and Fei Wang. Risk-sensitive multi-agent reinforcement learning in network aggregative games. *IEEE Transactions on Neural Networks and Learning Systems*, 31(5):1615–1627, 2020.

Christian Ewerhart and Dirk Leisen. Cumulative prospect theory and first price auctions. *Journal of Risk and Uncertainty*, 40(2):157–177, 2010.

Ziyang Fan, Ming Liu, Jun Wang, and Trevor Darrell. Conditional value-at-risk optimization for cooperative multi-agent reinforcement learning. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*, pages 321–329, 2019.

Javier García and Fernando Fernández. A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research*, 16:1437–1480, 2015.

Dushko Josheski and Goce Delcev. The cumulative prospect theory and first price auctions: an explanation of overbidding. *Journal of Economic Behavior & Organization*, 200:123–134, 2022. doi: 10.1016/j.jebo.2022.09.001.

Daniel Kahneman and Amos Tversky. Prospect theory: An analysis of decision under risk. *Econometrica*, 47(2):263–291, 1979.

Olivier Lepel and Anas Barakat. Beyond expected returns: A policy gradient algorithm for cumulative prospect theoretic reinforcement learning, 2024. URL `https://arxiv.org/abs/2410.02605`.

Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. *arXiv preprint arXiv:1706.02275*, 2017. URL `https://arxiv.org/abs/1706.02275`.

Ofir Nachum, Mohammad Norouzi, Kelvin Xu, and Dale Schuurmans. Bridging the gap between value and policy based reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 30, pages 2775–2785, 2017.

Aviv Tamar, Yishay Glassner, and Shie Mannor. Policy gradient for coherent risk measures. In *Advances in Neural Information Processing Systems*, volume 28, pages 1468–1476, 2015.

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

A CPT Function Hyperparameters

Table 1: Hyperparameters for the CPT value and probability-weighting functions across model variants.

Environment	Variant	α	β	λ	γ	δ	$(w^+)'$	$(w^-)'$
Simple Tag	Baseline	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	Moderate CPT (risk-seeking)	0.9	0.6	1.5	0.69	0.61	0.8	0.2
	Extreme CPT (risk-averse)	0.88	0.88	2.25	0.61	0.69	0.2	0.8
Simple Spread	Baseline	N/A	N/A	N/A	N/A			
	Moderate CPT (risk-averse)	0.88	0.88	2.25	0.61	0.69	0.2	0.8
	Extreme CPT (risk-averse)	0.7	0.95	2.5	0.61	0.69	0.2	0.8
	Observability CPT (Seeing - RS Agent)	0.7	0.7	0.8	0.61	0.69	0.8	0.2
	Observability CPT (Seeing - RA Agent)	0.65	0.65	2.8	0.61	0.69	0.25	0.75
	Dynamic (Agent 1)	0.7	0.7	2.5	0.61	0.69	0.8	0.2
	Dynamic (Agent 2)	0.65	0.65	2.8	0.61	0.69	0.8	0.2
	Dynamic Moderate (Agent 1)	0.6	0.6	1	0.5	0.55	0.2	0.8
	Dynamic Moderate (Agent 2)	0.3	0.3	1.5	0.5	0.55	0.2	0.8
	Dynamic Extreme (Agent 1)	1.2	1.2	1.2	0.5	0.69	0.2	0.8
	Dynamic Extreme (Agent 2)	0.3	0.3	1.5	0.5	0.69	0.2	0.8
Auction	CPT Agents	0.88	0.88	2.25	0.61	0.69	N/A	N/A
	Non-CPT Agents	N/A	N/A	N/A	N/A	N/A	N/A	N/A