DRL-ORA: Distributional Reinforcement Learning with Online Epistemic Risk Adaptation

Yupeng Wu[1] **Wenyun Li**[2] **Wenjie Huang**[2] **Chin Pang Ho**[3]

[1]London Business School
[2]The University of Hong Kong
[3]City University of Hong Kong

Abstract

One of the main challenges in reinforcement learning (RL) is that the agent has to make decisions that would influence the future performance without having complete knowledge of the environment. Dynamically adjusting the level of epistemic risk during the learning process can help to achieve reliable policies in safety-critical settings with better efficiency. In this work, we propose a new framework, *Distributional RL with Online Epistemic Risk Adaptation* (DRL-ORA). This framework quantifies both *epistemic* and implicit *aleatory* uncertainties in a unified manner and dynamically adjusts the epistemic risk levels by solving a total variation minimization problem *online*. The framework generalizes the existing variants of risk adaptation approaches with better explainability and flexibility. The selection of risk levels is performed efficiently via a Follow-The-Leader-type algorithm.We show that DRL-ORA outperforms existing methods that rely on fixed risk levels or manually designed risk level adaptation in multiple classes of tasks.

1 INTRODUCTION

Reinforcement learning (RL) algorithms have shown great success in games and simulated environments [Mnih et al., 2015], and have drawn interest for real-world and industrial applications [Mahmood et al., 2018, Zhang et al., 2020]. Taking a step further, distributional reinforcement learning (DRL) [Bellemare et al., 2017] considers the intrinsic distribution of the returns rather than their expectation, as considered in standard RL settings. This approach allows agents to quantify the risk of their policies and enables the development of risk-aware policies by considering the distributional information of their performance [Dabney et al., 2018b]. Quantifying both aleatory and epistemic uncertainties of RL crucial for developing risk-aware policies. The former expresses the inherent *randomness* in the problem, and the latter represents *a lack of knowledge* about the environment [Mavrin et al., 2019, Clements et al., 2019, Charpentier et al., 2022]. However, most existing research has focused on risk-aware RL with a *fixed risk level*, such as investigating risk awareness in the return distribution [Lim and Malik, 2022] and addressing both aleatoric and epistemic uncertainties [Eriksson and Dimitrakakis, 2020, Eriksson et al., 2022].

In this paper, we investigate *dynamic risk-awareness* strategy with respect to epistemic uncertainty, which reflects the attitude toward exploring unknown environments. Modeling risk-awareness of epistemic uncertainty is particularly relevant for applications such as autonomous driving, where the systems are almost deterministic, and the primary source of uncertainty is the lack of information about the environment. A higher risk-aversion level can avoid the negative consequences of excessive exploration that may lead to unsafe decisions in real-life (pessimism-under-uncertainty). On the other hand, a lower risk-aversion level (including risk-seeking) encourages exploring and collecting more data in unexplored regions (optimism-under-uncertainty). Although standard DRL can learn a policy under different levels of risk-awareness and adapt to various environments that require different degrees of caution, a specific risk-awareness level typically needs to be pre-specified and fixed for each environment prior to deployment. Studies have shown that optimism and pessimism-under-uncertainty settings outperform each other dependent on the task at hand, highlighting the need for an adaptive risk-awareness strategy that can balance these seemingly opposing perspectives [Moskovitz et al., 2021, Wang et al., 2024, 2025].

In addition, maintaining a fixed level of risk-awareness throughout the learning period is often practically suboptimal, as the level of risk concerned varies at different stages. An intelligent agent should be able to identify the appropriate level of risk required for effective decision-making at

Proceedings of the 42nd Conference on Uncertainty in Artificial Intelligence (UAI 2026), PMLR 337:7484–7503.

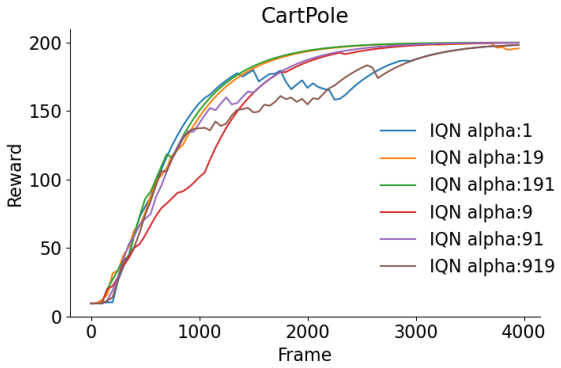


Figure 1: "IQN alpha:1" represents that α is fixed at 0.1 throughout all episodes. "IQN alpha:191" means that α is manually adjusted over the episodes, linearly increasing from 0.1 to 0.9 and then linearly decreasing back to 0.1. Same interpretation applies to the other settings.

different stages, rather than simply following a fixed risk-aware policy [Schubert et al., 2021, Liu et al., 2023]. In the context of RL, problems where uncertainties are expected to diminish over time (e.g., epistemic uncertainty) should be assigned a decreasing risk-awareness level to avoid excessive conservativeness [Eriksson and Dimitrakakis, 2020]. We illustrate this point further using a CartPole example.

Example 1. *The CartPole task was trained through Implicit Quantile Network (IQN) with left-tail Conditional Value-at-Risk (CVaR) at level $\alpha \in (0, 1]$ as its risk measure on the return (IQN will be explicitly described in Section 3). Over the episodes, the risk parameter α could either be fixed or manually adjusted. Figure 1 shows that "IQN alpha:191" outperforms all other settings, including IQN with fixed risk levels. The intuition is: at early stage of training, set a high risk-awareness level ($\alpha = 0.1$) to hedge the uncertain from the unknown environment, then move to a low risk-awareness level ($\alpha = 0.9$) to acquire high rewards as more experience from the environment is gained (and the epistemic uncertainty is reduced), and finally set back to a value in $[0, 1]$ that acts optimally in hindsight. This result highlights the advantage of using varying risk levels and motivates the development of more advanced methods that can automatically adjust the risk level and achieve even better performance.*

In this paper, we propose a DRL framework that is capable of automatically adjusting the risk-awareness level *on the fly* (i.e., online) by the epistemic uncertainty quantification in each period. The proposed framework does not rely on any pre-specified risk-awareness level and it can be incorporated with a wide range of risk measures based on IQN. Our framework generalizes and improves the existing adaptive risk-awareness level approach [Moskovitz et al., 2021, Liu et al., 2023, Wang et al., 2024, 2025]. Dynamic risk level selection methods are beneficial for RL algorithms because predetermining a suitable risk-awareness level is difficult, if not impossible, for a new domain task without any prior knowledge. Our experiments demonstrate that the proposed approach, which requires only moderate extensions to standard RL algorithms, achieves superior performance across various classes of practical problems compared to risk-neutral methods, fixed risk-awareness level methods, and existing adaptive risk-awareness level methods.

2 RELATED WORK

The literature on DRL has significantly expanded in recent years [Morimura et al., 2010, Bellemare et al., 2017, Barth-Maron et al., 2018, Dabney et al., 2018a, Yang et al., 2019, Singh et al., 2020]. This approach has been used to enhance exploration in RL [Mavrin et al., 2019] and in risk-aware applications [Bernhard et al., 2019]. Many different types of risk measures have been used in risk-aware DRL including CVaR [Lim and Malik, 2022, Choi et al., 2021, Keramati et al., 2020, Yang et al., 2022], power-law risk measure [Choi et al., 2021], distortion risk measure [Dabney et al., 2018a], entropic risk measure [Hai et al., 2022] and cumulative prospect theory [Prashanth et al., 2016]. Research has also been conducted on quantifying the risk of epistemic uncertainty, which demonstrates the attitude towards unexplored areas [Clements et al., 2019, Eriksson and Dimitrakakis, 2020, Tang and Agrawal, 2018, O'Donoghue, 2023], and on establishing comprehensive composite risk, which quantifies the joint effect of aleatory and epistemic risk during the learning process [Eriksson et al., 2022, Cho et al., 2023].

The above literature focuses on *fixed* risk-awareness levels. Recently, there has been growing evidence supporting the *adaptive* risk-awareness algorithms, such as risk scheduling approaches that explore risk levels and optimistic behaviors from a risk perspective [Oh et al., 2022]. These methods are not situation-dependent, however, risk levels decay linearly from high to specific low levels during the learning process. In addition, frameworks for value estimation, such as Tactical Optimism and Pessimism (TOP) and Quantile Option Architecture (QUOTA) [Moskovitz et al., 2021, Wang et al., 2024, 2025, Zhang and Yao, 2019], have been developed to balance the degree of optimistic and pessimistic uncertainty quantification. Many of these frameworks formulate the problem as a multi-armed bandit problem and utilize the Exponentially Weighted Average Forecasting (EWAF) algorithm. To implement this method, a pre-specified degree level should be set based on expert recommendations. A similar EWAF algorithm is used for adaptive risk tendency (ART) of CVaR in an IQN framework for a Nano Drone navigation application [Liu et al., 2023], and it achieves superior reward accumulation and efficient computational performance. The drawback of this method appears dur-

ing evaluation in environments with high obstacle density, possibly because it relies solely on the current observation without utilizing any memory. Besides, EWAF is not intuitive , and tuning the learning rate for weight forecasting still requires extra effort. In another work [Schubert et al., 2021], risk levels are adjusted based on a random network distillation (RND) error, which provides an estimate of how frequently a state has been visited during training. While the performance heavily depends on the original RND architecture, it is possible that a more conservative risk setting could hinder exploration.

3 PRELIMINARIES

Distributional Reinforcement Learning (DRL) Consider a Markov Decision Process (MDP) [Puterman, 2014] defined as a tuple $(\mathbb{S}, \mathbb{A}, R, P, \gamma)$, where $\mathbb{S}$ is the state space, $\mathbb{A}$ the action space, R is the (state and action-dependent) reward function, $P(\cdot|s,\, a)$ is the transition probability associated with state-action pair $(s,\, a)$, and $\gamma \in (0, 1]$ is the discounted factor. A policy $\pi(\cdot|s)$ maps the state s to a probability distribution over actions.

DRL [Bellemare et al., 2017] introduces a new perspective other than the standard RL setting, by learning the entire return distribution of the action-value function $Q(s,\, a)$ directly instead of estimating only the expected return $Q(s,\, a)$. This approach allows the agent to capture more detailed information beyond a single expectation value. Given a policy π, the distributional Bellman equation is defined as

$$Z(s,\, a) :\overset{D}{=} R(s,\, a) + \gamma Z\left(s',\, a'\right),$$

for $s' \sim P(\cdot \mid s,\, a)$, $a' \sim \pi(\cdot|s')$, where Z is the distributional value function and $\overset{D}{=}$ represents the equality in distribution. Its corresponding optimality equation is defined as

$$Z(s,\, a) :\overset{D}{=} R(s,\, a) + \gamma Z\left(s',\, \arg\max_{a' \in \mathbb{A}} \mathbb{E}\left[Z\left(s',\, a'\right)\right]\right),$$

for $s' \sim P(\cdot \mid s,\, a)$. It is proved by Bellemare et al. [2017] that the distributional Bellman operator is a contraction in a maximal form of the Wasserstein metric. There are approaches [Bellemare et al., 2017, Dabney et al., 2018b,a, Yang et al., 2019, Rowland et al., 2019] to estimate the return distribution, most of which are based on Quantile Regression [Koenker and Bassett Jr, 1978].

The Implicit Quantile Network (IQN) proposed by Dabney et al. [2018a] provides an effective way to learn an implicit representation of the return distribution, yielding a powerful function approximator for a Deep Q-Network agent. IQN can re-parameterize samples from a based distribution to the respective quantile value of a target distribution, which produces risk-aware policies. Let $\beta : [0,\, 1] \to [0,\, 1]$ be a *distortion* function, and choosing β as the identity function corresponds to risk-neutrality. Let $Z_q(s,\, a)$ denote the q-quantile function (inverse of cumulative density function) of the return distribution, then the distorted expectation of $Z(s,\, a)$ under β and its risk-aware greedy policy are denoted as

$$\begin{aligned} Q_\beta(s,\, a) &:= \mathbb{E}_{q \sim U[0,1]}[Z_{\beta(q)}(s,\, a)], \\ \pi_\beta(s) &= \arg\max_{a \in \mathbb{A}} Q_\beta(s,\, a). \end{aligned} \tag{1}$$

For brevity, the IQN loss function and its detailed training process are summarized in Appendix Section A.

Adaptive Risk-awareness (Risk Tendency) The proposed adaptive risk-awareness approach treats the selection of an exploration strategy (risk awareness level) as a multi-armed bandit problem [Moskovitz et al., 2021, Wang et al., 2024, 2025], where each bandit arm corresponds to a specific choice of distortion function β. Define a discrete set of D distortion functions $\mathcal{D} = \{\beta^1, ..., \beta^D\}$ recommended by experts. After sampling $d_n \in \{1, ..., D\}$ at epsiode n, we form a belief distribution $p_n \in \Delta_{\mathcal{D}}$ ($\Delta_{\mathcal{D}}$ is the probability simplex on $\mathcal{D}$) of the form $p_n(d) \propto \exp(w_n(d))$ for all $d \in \mathcal{D}$. The distribution is updated using the Exponentially Weighted Average Forecasting (EWAF) algorithm based on a received feedback signal in each episode. After receiving a feedback signal $g_n := R_n - R_{n-1}$ where R_n is the cumulative reward obtained in episode n, the weight w_n is updated by:

$$w_{n+1}(d) = \begin{cases} w_n(d) + \eta \cdot (g_n / p_n(d)), & d = d_n, \\ w_n(d), & \text{otherwise}, \end{cases}$$

where $\eta > 0$ is the step-size. In each episode n, the β in Eq.(1) is chosen by random sampling from distribution p_n.

Another variants of constructing the feedback signal is the right-truncated variance difference of reward distributions between two successive periods (time-steps) [Liu et al., 2023], where $g_t := \text{RTV}_t - \text{RTV}_{t-1}$ and

$$\text{RTV}_t := \frac{2}{N} \sum_{i=1}^{N/2} (F_{Z_t}^{-1}(q_i) - F_{Z_t}^{-1}(q_{N/2}))^2,$$

in which q_i is the i/Nth quantile levels and N is a sample size. The EWAF algorithm is applied to update the weight w_t in periodic frequency.

4 METHODOLOGY: DRL-ORA

We propose DRL-ORA, the first distributional reinforcement learning framework capable of online, state-action-level adaptation of epistemic risk attitudes without requiring pre-specified risk levels or manual scheduling. At its core, DRL-ORA introduces a unified uncertainty quantification scheme that explicitly disentangles epistemic from

aleatory uncertainty via ensemble networks, and formulates risk adaptation as a non-convex online learning problem with a well-defined regret objective. This marks a fundamental departure from prior approaches—such as EWAF-based bandit selection—which lack explainability, rely on predefined discrete risk sets, and fail to leverage full distributional information. In contrast, DRL-ORA enables fine-grained, per-transition risk adaptation, grounded in a total variation minimization criterion that is both theoretically tractable and computationally efficient. The framework is agnostic to the choice of distortion risk measures, supports both CVaR and quantile-based variants, and establishes a novel connection to satisficing measures in decision theory—revealing that the offline oracle in our formulation corresponds to a quasi-concave satisficing objective. Together, these innovations yield a flexible, interpretable, and high-performing risk adaptation paradigm that consistently outperforms fixed-risk and existing adaptive baselines across diverse tasks.

A Generalized Non-convex Learning Perspective In the existing EWAF approach, the major shortcoming is its explainability. The theoretical foundation/objective of the approach in updating the risk tendency is unclear. The most similar framework to the approach is online optimization on simplex [Bubeck, 2011], where the signals must be an explicit loss function in the decision variable, rather than simple numerical outputs. There are also several other shortcomings. In this approach, epistemic uncertainty is not disentangled from the reward distribution, which means that the agent cannot specify the risk-awareness level towards exploration as it will further lead to inaccurate adaptation. Besides, it relies on truncated variance or cumulative reward, rather than leveraging the full information from the epistemic uncertainty distribution, to construct the feedback signal. Furthermore, this approach requires a predefined discrete finite set of distortion functions (e.g., only two functions are used in [Liu et al., 2023]).

Instead, we propose a non-convex learning perspective that constructs the loss function as the feedback signal, generalizing the variants of the existing approaches, and offering greater explainability, flexibility and improved accuracy. Given T as the total periods (time-steps) of training, we then define a loss function $l_t : \mathcal{A} \to \mathbb{R}$ for all $t = 1, \ldots, T$. Our feedback signal at period t is:

$$\begin{aligned} &l_t(\alpha(s,\, a)) \\ :=&\big|\rho_{\alpha(s,\,a)}(X_t(s,\, a)) - \rho_{\alpha(s,\,a)}(X_{t+1}(s,\, a))\big|, \end{aligned} \tag{2}$$

where $X_t(s,\, a)$ represents the epistemic uncertainty associated with the state-action pair $(s,\, a)$ at period t. Here we define the parametric risk measure on the epistemic uncertainty: $\rho_\alpha : \mathcal{L} \to \mathbb{R}$, where $\mathcal{L}$ represents the space of bounded random variables, $\alpha \in \mathcal{A}$ is a parameter that controls the level of risk-awareness, and $\mathcal{A} \subset \mathbb{R}$ is a compact set containing all possible risk parameters. Explicitly, $\alpha(s,\, a) \in \mathcal{A}$ denotes the risk parameter chosen corresponding to the state-action pair $(s,\, a)$.

The risk measure $\rho_\alpha(\cdot)$ is Lipschitz continuous but not necessarily convex with respect to α. Distortion risk measures [Wang, 1996, Dabney et al., 2018b] fall under our framework and are well-suited for the IQN framework. Specifically, for $X \in \mathcal{L}$, we have

$$\rho_\alpha(X) := \int_{-\infty}^{0} [g_\alpha(S_X(x)) - 1]dx + \int_0^{\infty} g_\alpha(S_X(x))dx,$$

where g_α is the distortion function parameterized by α, and $S_X(x)$ is the survival distribution function of random variable X. Suppose g_α is a bounded and Lipschitz continuous function with respect to α. Typically, when X represents a random *loss*, a larger risk parameter α indicates a lower level of risk-awareness. Below, we provide several examples along with their corresponding g_α and $\mathcal{A}$.

Example 2 ([Acerbi and Tasche, 2002])**.** *For the right-tail CVaR, CVaR$_\alpha(X) = \mathbb{E}[X|X \geq F^{-1}(1-\alpha)]$, where F^{-1} denotes the inverse CDF of X. The corresponding distortion function is $g_\alpha(x) = \max\{-x/\alpha + 1,\, 0\}$, and $\mathcal{A} = [\alpha_{\min},\, 1]$ for some $\alpha_{\min} \in (0, 1]$.*

Example 3 ([Dhaene et al., 2012])**.** *For the right-tail α-quantile,* i.e.*, $\rho_\alpha(X) = F_X^{-1}(1-\alpha)$, the distortion function is $g_\alpha(x) = \mathbb{I}\{x \leq \alpha\}$, and $\mathcal{A} = [\alpha_{\min},\, 1]$ for some $\alpha_{\min} \in (0, 1]$. Note that g_α here is not continuous in α; the Lipschitz condition assumed in our analysis serves as a sufficient condition for the general case, and the quantile distortion requires a mild additional discretization argument, which we provide in Appendix Section C.*

Our construction of the feedback signal (2) has several features and advantages:

(i) Our feedback signal generalizes the existing approach by utilizing the full information of the epistemic uncertainty distribution. Also, the feedback signal is defined as an explicit function of the risk parameter, so the risk adaptation problem can be formally written as an online learning problem with clear objective and decision foundation.

(ii) Our approach provides higher flexibility, as the risk levels associated with different state-action pairs can be updated independently and simultaneously. Furthermore, the adaptation is performed in every period/transition rather than episodically, as in [Moskovitz et al., 2021, Wang et al., 2024, 2025]. These will be validated later by experiments, demonstrating improved training performance during early periods and episodes.

(iii) The risk measure $\rho_\alpha(\cdot)$ focuses on the right α-level quantile of a distribution, capturing the risk of left-truncated epistemic uncertainty (reflecting upper-tail

variability), which makes it comparable with the setting of reward distribution RTV in [Liu et al., 2023]. It is inspired by setting a contrary to [Mavrin et al., 2019], which argues that the upper-tail variability of reward distribution reflects optimism of exploration in the face of uncertainty, particularly when the distribution is asymmetric.

The offline objective, in hindsight, is formulated for each state-action pair $(s,\, a)$ as

$$\min_{\alpha(s,\,a)\in\mathcal{A}} \sum_{t=1}^{T-1} l_t(\alpha(s,\, a)), \tag{3}$$

which can be interpreted as finding a state-action-dependent risk parameter that minimizes the *Total Variation* [Chambolle, 2004, Huang et al., 2008, Li et al., 2013] (i.e., the sum of feedback signals) of the epistemic uncertainty risk.

The entire learning period under hindsight reveals the absolute level of risk awareness associated with a particular risk parameter that stabilizes the negative impact of epistemic uncertainty variation. However, the relative levels of risk awareness during the process cannot be explicitly elicited. At any period $t < T$, only limited information $(X_1, \ldots, X_{t+1})$ is available, allowing the best risk parameter to be forecasted on the fly using an appropriate online learning method. The "regret" of the online learning algorithm under the outputs $\alpha_1, \ldots, \alpha_T$ is defined as

$$\mathfrak{R}_T := \sum_{t=1}^{T} l_t(\alpha_t) - \min_{\alpha\in\mathcal{A}} \sum_{t=1}^{T} l_t(\alpha),$$

where the dependence of α and X_t on $(s,\, a)$ is omitted only for simplicity, as will be done in the following discussions.

Ensemble Networks for Epistemic Uncertainty Quantification Building on the principles of Bayesian inference [MacKay, 2003], the *ensemble network* is a straightforward approach for modeling epistemic uncertainty and is commonly used in prior work [Dimitrakakis, 2006, Osband et al., 2016, Agarwal et al., 2020]. Recall that in the regular RL framework, a neural network θ is used to estimate the Q-function $Q^\theta(s, a)$, where the distribution of θ reflects the current epistemic uncertainty during training [Eriksson et al., 2022]. While modeling the distribution of θ is challenging, its corresponding network outputs $Q^\theta(s,\, a)$ can be directly observed.

Thus, ensemble networks create K network heads during training, each initialized with different parameters θ, denoted by $\{\theta_i\}_{i=1}^K$. For each state-action pair $(s,\, a)$, these K networks output K different values $Q^{\theta_k}(s,\, a)$ for $k = 1, \ldots, K$. The distribution of $Q^{\theta_k}(s,\, a)$ is used to approximate the corresponding distribution of θ, which represents the epistemic uncertainty at $(s,\, a)$. Formally, for each $(s,\, a)$, the ensemble networks generate $Q^{\theta_k}(s,\, a)$, and a distribution Y maps $(s,\, a)$ to a uniform probability distribution supported on $\{Q^{\theta_k}(s,\, a)\}$, defined as:

$$Y(s,\, a) = 1/K \sum_{k=1}^{K} \delta\{Q^{\theta_k}(s,\, a)\},$$

where $\delta\{z\}$ is the Dirac delta function at $z \in \mathbb{R}$. Thus, the epistemic uncertainty associated with $(s,\, a)$ is represented by the distribution $Y(s,\, a)$. At time step $t-1$ in the RL process, given the explored state-action pair $(s_{t-1},\, a_{t-1})$ and the epistemic uncertainty distribution $X_{t-1}(s,\, a)$ for each $(s,\, a)$, the epistemic uncertainty distribution $X_t(s,\, a)$ is updated asynchronously as

$$\begin{array}{ll} X_t(s,\, a) \leftarrow Y(s_{t-1},\, a_{t-1}), & \text{if } (s,\, a) = (s_{t-1},\, a_{t-1}), \\ X_t(s,\, a) \leftarrow X_{t-1}(s,\, a), & \text{otherwise,} \end{array}$$

where $Y(s_{t-1},\, a_{t-1})$ is the distribution generated by the ensemble networks. This distribution is supported by the Q-functions of each network, where each Q-value is computed as the quantile expectation over the return distribution (aleatoric uncertainty). It is important to note that the return distribution in each network is updated based on the output of its corresponding ensemble network. This ensures that the epistemic uncertainty is properly quantified and disentangled from the overall uncertainties. A potential concern is that ensemble networks may collapse to similar predictions as training progresses, driving the epistemic uncertainty signal to zero and undermining the adaptation mechanism. We address this in two ways. First, we maintain ensemble diversity through independent initialization with distinct random seeds and independent experience replay sampling, following well-established practices [Lakshminarayanan et al., 2017, Osband et al., 2016]. Second, even if ensemble variance decreases in well-explored regions, this is epistemically correct behavior: low epistemic uncertainty should drive ORA toward risk-neutral policies, which is the intended adaptation dynamics. Empirically, the epistemic signal remains non-degenerate throughout training; see Appendix D for supporting evidence. Appendix Figure 6 shows the tests on different ensemble size K. It demonstrates that the increase of ensemble network size does not exhibit a significant correlation with performance improvement. Both the time and memory overhead grow slightly faster than linear growth with K. With the default value of K, this overhead is not significant.

We finally wrap-up the proposed algorithm – *Distributional RL with Online Epistemic Risk Adaptation (DRL-ORA)* in Algorithm 1. In each period t, the epistemic uncertainty is estimated using ensemble networks, where the return distribution is updated based on the output of each ensemble network (Line 9). The historical epistemic uncertainties are then used to dynamically update the risk parameter through the proposed online non-convex learning method (Line 10-11). The composite risk framework quantifies the combined

Algorithm 1 DRL-ORA (on Top of IQN Objective)

1: **Input:** Aleatory uncertainty distortion function β; epistemic uncertainty distortion function g_α with risk parameter $\alpha \in [\alpha_{\min}, \alpha_{\max}] \subset [0, 1]$ (the risk parameter is associated with each state-action pair) and corresponding risk measure ρ_α; parameter of exponential distribution η, arbitrarily chosen $\alpha_1 \in \mathcal{A}'$, hyperparameters: $M > 0$, $K > 0$, discounted factor $\gamma > 0$; Training period $T > 0$.

2: **Initialize:** The return distribution Z^k for each ensemble network $k = 1, ..., K$, risk parameter $\alpha(s, a) = \alpha_{\max}$ for all (s, a), initial state s_0 and initial action a_0, initial estimated epistemic uncertainty distribution $X_1(s, a)$ for all (s, a).

3: **for** $t = 1$ **to** T **do**

4: Set $\bar{\alpha}_t(s) = \alpha_t(s, a_{t-1})$.

5: Choose an action by

$$a_t \in \arg\min_{a \in \mathcal{A}} \rho_{\bar{\alpha}_t(s_t)}(X_t(s_t, a)).$$

Observe return r_t and state transition s_{t+1}.

6: **for** $k = 1$ **to** K **do**

7: Compute $Q^{\theta_k}(s, a) = \frac{1}{M} \sum_{m=1}^{M} Z^k_{\beta(\tilde{q}_m)}(s, a)$ by M samples of $\tilde{q}_m \sim U([0, 1])$, where the return distribution Z^k is updated based on the kth ensemble network's output.

8: **end for**

9: Update the estimated epistemic uncertainty distribution $X_{t+1}(s, a)$ for all (s, a) asynchronously.

10: Generate randomly $\sigma_{t+1} \overset{\text{i.i.d}}{\sim} \exp(\eta)$;

11: Compute state-action-level update (simplified notation)

$$\alpha_{t+1} \in \arg\min_{\alpha \in \mathcal{A}'} \left\{ \sum_{i=1}^{t} l_i(\alpha) - \sigma_{t+1}\alpha \right\}.$$

12: **end for**

effects of aleatory and epistemic risks by applying separate risk-awareness levels to the epistemic uncertainty and the return distribution (which implicitly captures aleatory uncertainty) using distortion functions g_α and β, respectively (Line 5 & 7). The action is chosen to minimize the epistemic uncertainty risk under the latest estimation of the parameter (Line 5).

Regret Minimization and Analysis Given that our loss function is not necessarily convex in the risk parameter α, standard online convex optimization algorithms cannot guarantee sublinear regret. The EWAF algorithm also fails in this setting, as our goal is to identify a specific risk parameter through continuous optimization rather than updating logit weights over a predefined discrete set. The Follow the Perturbed Leader (FTPL) algorithm [Agarwal et al., 2019] achieves sublinear regret complexity $O(T^{1/2})$ for the expected regret $\mathbb{E}[\mathfrak{R}_T]$ when the perturbation $\{\sigma_t\}_{t \geq 0}$ is sampled from an exponential distribution with parameter η. Theorem 4 demonstrates that a proper discretization of $\mathcal{A}$ into a finite set $\mathcal{A}' \subseteq \mathcal{A}$, combined with grid search over $\mathcal{A}'$, can efficiently solve the offline oracle problem while preserving the $O(T^{1/2})$ expected regret complexity. This method is particularly suitable when gradient or subgradient information is difficult to obtain. We note that this regret analysis concerns only the online optimization of α, assuming given epistemic uncertainty estimates. It does not imply end-to-end convergence for the full RL algorithm, which remains an open problem that we leave for future work.

Theorem 4. *For an arbitrarily small $\epsilon > 0$, the set $\mathcal{A}$ can be properly discretized as $\mathcal{A}'$, such that the Hausdorff distance between the two sets, i.e.,*

$$\max\left\{ \sup_{\alpha \in \mathcal{A}} \inf_{\alpha' \in \mathcal{A}'} |\alpha - \alpha'|, \inf_{\alpha \in \mathcal{A}} \sup_{\alpha' \in \mathcal{A}'} |\alpha - \alpha'| \right\} \leq \epsilon.$$

In addition, by choosing $\epsilon = O(T^{-1/2})$, Algorithm 1 can achieve $O(T^{1/2})$ expected regret complexity.

Extension: Relation to Satisficing Measure Now we define a *recursive* loss function $l'_t : \mathcal{A} \times \mathcal{A} \to \mathbb{R}$ for all $t = 1, \ldots, T$, where the first argument represents the decision variable α, and the second argument is the parameter corresponding to the solution from the previous period. Specifically, at period $t - 1$, we define

$$l'_{t-1}(\alpha, \alpha_{t-1}) := |\rho_\alpha(X_{t-1}) - \rho_{\alpha_{t-1}}(X_t)|.$$

It can be shown that the cumulative regret under the above recursive loss function is equivalent to the cumulative regret under the regular loss function; that is,

$$\mathfrak{R}'_T := \sum_{t=1}^{T} l'_t(\alpha_t, \alpha_t) - \min_{\alpha \in \mathcal{A}} \sum_{t=1}^{T} l'_t(\alpha, \alpha) = \mathfrak{R}_T.$$

We then modify and implement Algorithm 1 with the offline oracle model (Line 11) as

$$\alpha_t \in \arg\min_{\alpha \in \mathcal{A}'} l'_{t-1}(\alpha, \alpha_{t-1}),$$

where the random perturbation term is omitted. Using this recursive loss function requires significantly less storage space compared to the regular loss, but the tradeoff is that the modified Algorithm 1 incurs an $O(T)$ expected regret complexity for both $\mathfrak{R}_T$ and $\mathfrak{R}'_T$, which implies that, at least, a local optimum of the hindsight problem is attained.

The connection between this algorithm's offline oracle model and the *satisficing measure* [Brown and Sim, 2009] from the decision analysis literature, can be established. The satisficing measure evaluates the extent to which a random

outcome fails to meet a fixed target. To formalize this connection. Let $\tau_t := \rho_{\alpha_{t-1}}(X_t)$ denote the "target" at period t. Given Ω as the scenario space, if $\min_{\omega\in\Omega} X_{t-1}(\omega) \geq \tau_t$, then $\alpha_t = \alpha_{\min}$ (which could be 0, depending on the chosen risk measure). Conversely, if $\max_{\omega\in\Omega} X_{t-1}(\omega) \leq \tau_t$, then $\alpha_t = \alpha_{\max}$ (which could be 1, depending on the chosen risk measure). Otherwise, the offline oracle problem of Algorithm 1 becomes equivalent to solving the following quasi-concave optimization problem:

$$\max\{1-\alpha \,:\, \rho_\alpha(X_{t-1}) \leq \tau_t,\, \alpha \in \mathcal{A}\}, \qquad (4)$$

where the optimal solution provides the level of non-attainability of X_{t-1} relative to τ_t. Theorem 5, based on the analysis in Section 3.4 of Brown and Sim [2009] and Theorem 3.6 of Huang [2020], demonstrates that when ρ_α is chosen as CVaR, Problem (4) can be reformulated as a stochastic convex program.

Theorem 5. *Suppose $\rho_\alpha = CVaR_\alpha$. We have*

$$\alpha_t = \min_{b\geq 0} \mathbb{E}\left[(b(X_{t-1}-\tau_t)+1)_+\right],$$

with $\tau_t = \rho_{\alpha_{t-1}}(X_t)$.

After applying sample average approximation for the expectation, Problem (4) can be further reformulated as a linear programming (LP) problem. The number of decision variables and constraints are both $O(K)$, and the best-known LP algorithm has a complexity of $O(K^{2.5})$ [Vaidya, 1989]. However, we develop a specialized search algorithm, illustrated in Appendix Section B, which can determine α_t with significantly lower computational complexity.

Theorem 6. *The search algorithm (Appendix Section B: Algorithm 2) computes the optimal risk parameter in time $O(K \log K)$.*

5 EXPERIMENTS AND APPLICATIONS

We conduct experiments on three classes of tasks to evaluate the performance of DRL-ORA. In all experiments, CVaR is used in ORA-based methods with $\alpha_{\min} = 0.1$, resulting in $\mathcal{A} = [0.1, 1]$. Additionally, we set $\beta(q) = q$, making the agent risk-neutral with respect to the implicit aleatory uncertainties of the tasks. We also implement the two existing risk aware adaptation approaches: ART [Liu et al., 2023] and TOP [Moskovitz et al., 2021, Wang et al., 2024, 2025] as benchmarks.

Atari Games We first demonstrate the performance of DRL-ORA on CartPole and the Atari games: Hero, MsPacman and SpaceInvaders. Here we analyze the results of CartPole task in addition to Example 1. Figure 7 in Appendix, built upon Figure 1, shows that DRL-ORA outperforms all other methods, with a noticeable reward advantage at the early stages of the episodes. To statistically validate the observed performance differences, we employed the Mann-Whitney U test [Mann and Whitney, 1947]. The resulting Rank-Biserial correlation effect sizes (0.990 against ART and 0.787 against TOP, $p < 0.001$) confirm a substantial advantage of ORA over both the ART and TOP baselines in the CartPole environment. Furthermore, the 90% confidence interval of DRL-ORA highlights its stability. To evaluate the robustness of our approach, we compared the ORA using quantile ("ORA(quantile)") against the default CVaR-based version ("ORA(CVaR)"). The results show that despite a slight performance reduction, ORA(quantile) maintains considerable risk adaptation capability. The subsequent experiments further demonstrate the robustness of ORA across different risk measures. Due to limited spaces, please refer to Appendix Section D for the additional experiment results on other Atari games.

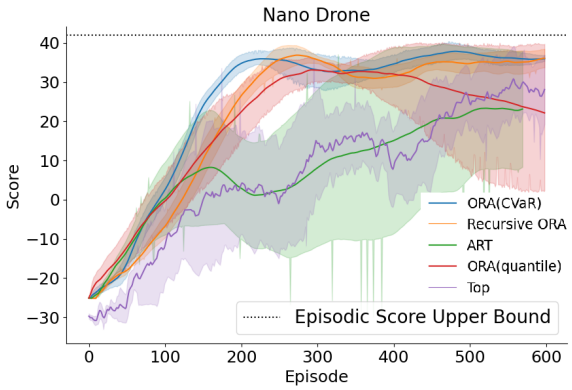


Figure 2: Average episodic scores in Nano Drone navigation task. The shaded area represents a 90% confidence interval.

Nano Drone Navigation We evaluate DRL-ORA on an open-source Nano Drone navigation task [Liu et al., 2023], where ART has shown superior performance compared to IQN. In this partially observable environment, the drone aims to maximize its score by reaching the target while avoiding obstacles. It gains points for success but loses points for time and collisions, and the agent's level of risk-awareness directly influences its exploration strategy and overall performance. We record the complete training and testing results for ORA with the regular loss function ("Recursive ORA" in short), ORA with the recursive loss function, ART, TOP and IQN under various fixed risk parameters. All training parameters, including learning rate and random seeds, are consistent with the default settings in [Liu et al., 2023], see Appendix Table 6.

As shown in Figure 2, ORA not only shows the best training performance and fastest convergence but also significantly outperforms both baselines ($p < 0.001$), achieving Rank-Biserial effect sizes of 0.319 over ART and 0.309 over TOP.

Table 1: Success and collision rate percentage

Metric (%)	Density 2		Density 6		Density 12	
	Suc.	Col.	Suc.	Col.	Suc.	Col.
IQN α=0.5	88	5	70	15	48	32
IQN α=1.0	88	4	74	10	52	25
ART	84	9	73	15	54	25
Recursive ORA	93	5	75	16	54	26
TOP	90	1	78	5	60	26
ORA(CVaR)	92	6	**81**	13	**64**	28
ORA(quantile)	**94**	6	75	16	52	23

Table 1 also presents the average testing results of each algorithm over 500 evaluation runs in environments with varying obstacle densities (i.e., the number of obstacles). Testing is particularly important in this navigation task, as excessive exploration during training may result in lower rewards but ultimately lead to an effective policy with strong testing performance. The testing results show that ORA outperforms all other algorithms, especially in environments with higher uncertainty (*Density 6 & 12*). In contrast, in the task with lower environmental uncertainty (*Density 2*), Recursive ORA slightly outperforms ORA while offering more efficient computation and requiring less storage for historical epistemic uncertainties.

Knapsack We assess DRL-ORA by applying it to a classic operations research (OR) problem, the Knapsack problem, on OR-gym [Hubbs et al., 2020, Balaji et al., 2019], a widely utilized RL library for OR problems. The Knapsack problem is a combinatorial optimization problem aiming at maximizing the total value of items placed in a knapsack without exceeding its maximum capacity. The agent's reward is the value of items placed in the knapsack, and an episode terminates once the capacity is full. There is no aleatory uncertainty in the Knapsack problem. We set *50 items* in the Binary Knapsack [Hubbs et al., 2020] for a lower computational load while keeping other settings aligned with the default. All training hyper-parameters are provided in Appendix Table 5. In terms of testing results in Figure 3, ORA achieves a higher average reward than IQN, demonstrating that by precisely capturing epistemic uncertainty, our adaptive algorithm maintains its effectiveness in such scenarios. Our ORA method achieves perfect separation from TOP (effect size = 1.000) and substantial advantages over DQN (0.699) and ART (0.544), all with $p <$ 0.001. Note that the testing result should be normally higher than the final training reward because a fixed ϵ probability for random action selection is used in the whole training, while there is no random action selection in testing.

In Figure 4, IQN with $\alpha = 1.0$ demonstrates a clear advantage over IQN with $\alpha = 0.1$, while regular Deep Q-Network (DQN) [Mnih et al., 2015]. fails to perform effectively. This

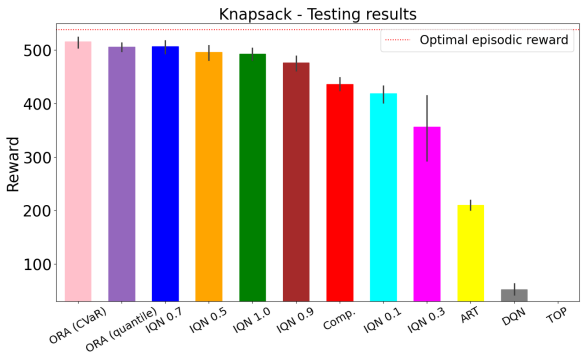


Figure 3: Testing results with 90% confidence interval. "Comp." means Composite IQN. The "Optimal episodic reward" is the benchmark solved via DP.

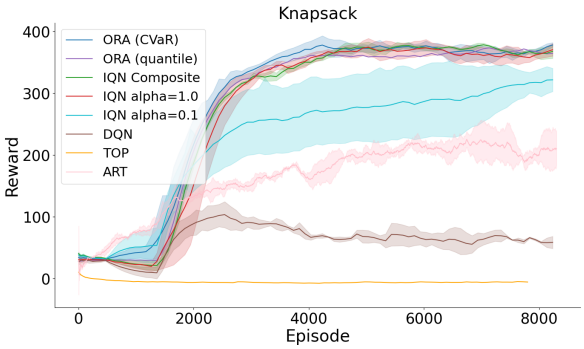


Figure 4: Reward lines on Knapsack.

suggests that lower risk awareness is more appropriate in tasks without aleatory uncertainty. Once again, ORA outperforms the best IQN, TOP and ART, particularly during the early and intermediate stages of training. Additionally, we conduct an ablation experiment comparing ORA with the IQN Composite method [Hai et al., 2022]. The IQN Composite algorithm is derived from Algorithm 1 by fixing $\alpha_t = 1$ for all $t = 1, \ldots, T$ and for all state-action pairs, making it risk-neutral with respect to epistemic uncertainty. This method replaces the expectation imposed on the return distribution with an expectation imposed solely on epistemic uncertainties, as there is no aleatory uncertainty in this task. Figure 4 shows that IQN Composite fails to achieve similar performance to ORA in the early training episodes, indicating that ORA's superior performance stems from its adaptive mechanism for risk level selection.

Further Remarks Table 3 in Appendix, reports, for ORA, the monitored left-truncated variance (LTV) with respect to the median of the epistemic uncertainty distribution obtained in each training period of the three experiments. The table also shows the corresponding risk parameters used in each case. CartPole has the smallest LTV value in average. Its LTV fluctuates over the whole training process, and the corresponding risk parameter remains at a high level. We

also observe that LTV is almost uni-modal over episodes in Nano Drone navigation and Knapsack. A high risk parameter (lower risk-awareness level), is favored at the beginning to acquire knowledge of the environment, whereas it subsequently declines in performance when large LTVs are detected. Knapsack has the largest LTV value which decreases slower than that in Nano Drone navigation after reaching the peak. This can explain why the superior performance of ORA lasts longer in the Knapsack problem than that in Nano Drone navigation (see Figure 2 and 4).

6 CONCLUSION

In this work, we propose an RL framework with an adaptive mechanism for learning risk levels to hedge against the epistemic uncertainties. The method only needs an extra multi-head ensemble structure so is eligible of employing different loss functions and distortion risk measures and being applied to IQN-type algorithms. Various applications validate the superiority of the proposed method. For future research, we plan to develop more accurate and efficient methods for epistemic uncertainty quantification, e.g., improving the scalability when a large number of ensemble networks are applied. We will also extend our approach to RL tasks under non-stationary environments. Our approach is expected to remain valid under non-stationary environment as it is adaptive according to the latest experience/episode (epistemic uncertainty estimation).

Acknowledgements

Wenjie Huang is supported by HKU Seed Fund for PI Research - Basic Research (Project No. 2402101377), NSFC Young Scientist Fund (Project No. 72201224), and Research Grants Council of Hong Kong (General Research Fund, Project No. 17211325).

Chin Pang Ho is supported, in part, by CityUHK Start-Up Grant (Project No. 9610481) and the Research Grants Council of Hong Kong (General Research Fund, Project No. 11510825).

References

Carlo Acerbi and Dirk Tasche. On the coherence of expected shortfall. *Journal of Banking & Finance*, 26(7):1487–1503, 2002.

Naman Agarwal, Alon Gonen, and Elad Hazan. Learning in non-convex games with an optimization oracle. In *Conference on Learning Theory*, pages 18–29. PMLR, 2019.

Rishabh Agarwal, Dale Schuurmans, and Mohammad Norouzi. An optimistic perspective on offline reinforcement learning. In *International Conference on Machine Learning*, pages 104–114. PMLR, 2020.

Bharathan Balaji, Jordan Bell-Masterson, Enes Bilgin, Andreas Damianou, Pablo Moreno Garcia, Arpit Jain, Runfei Luo, Alvaro Maggiar, Balakrishnan Narayanaswamy, and Chun Ye. Orl: Reinforcement learning benchmarks for online stochastic optimization problems. *arXiv preprint arXiv:1911.10641*, 2019.

Gabriel Barth-Maron, Matthew W Hoffman, David Budden, Will Dabney, Dan Horgan, Dhruva TB, Alistair Muldal, Nicolas Heess, and Timothy Lillicrap. Distributed distributional deterministic policy gradients. In *International Conference on Learning Representations*, 2018.

Marc G Bellemare, Will Dabney, and Rémi Munos. A distributional perspective on reinforcement learning. In *International Conference on Machine Learning*, pages 449–458. PMLR, 2017.

Julian Bernhard, Stefan Pollok, and Alois Knoll. Addressing inherent uncertainty: Risk-sensitive behavior generation for automated driving using distributional reinforcement learning. In *2019 IEEE Intelligent Vehicles Symposium (IV)*, pages 2148–2155. IEEE, 2019.

David B Brown and Melvyn Sim. Satisficing measures for analysis of risky positions. *Management Science*, 55(1): 71–84, 2009.

Sébastien Bubeck. Introduction to online optimization. *Lecture notes*, 2:1–86, 2011.

Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

Antonin Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20:89–97, 2004.

Bertrand Charpentier, Ransalu Senanayake, Mykel Kochenderfer, and Stephan Günnemann. Disentangling epistemic and aleatoric uncertainty in reinforcement learning. *arXiv preprint arXiv:2206.01558*, 2022.

Taehyun Cho, Seungyub Han, Heesoo Lee, Kyungjae Lee, and Jungwoo Lee. Pitfall of optimism: Distributional reinforcement learning by randomizing risk criterion. *Advances in Neural Information Processing Systems*, 36: 56802–56824, 2023.

Jinyoung Choi, Christopher Dance, Jung-Eun Kim, Seulbin Hwang, and Kyung-sik Park. Risk-conditioned distributional soft actor-critic for risk-sensitive navigation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8337–8344. IEEE, 2021.

William R Clements, Bastien Van Delft, Benoît-Marie Robaglia, Reda Bahi Slaoui, and Sébastien Toth. Estimating risk and uncertainty in deep reinforcement learning. *arXiv preprint arXiv:1905.09638*, 2019.

Will Dabney, Georg Ostrovski, David Silver, and Rémi Munos. Implicit quantile networks for distributional reinforcement learning. In *International Conference on Machine Learning*, pages 1096–1105. PMLR, 2018a.

Will Dabney, Mark Rowland, Marc Bellemare, and Rémi Munos. Distributional reinforcement learning with quantile regression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018b.

Jan Dhaene, Alexander Kukush, Daniël Linders, and Qihe Tang. Remarks on quantiles and distortion risk measures. *European Actuarial Journal*, 2:319–328, 2012.

Christos Dimitrakakis. Nearly optimal exploration-exploitation decision thresholds. In *Artificial Neural Networks–ICANN 2006: 16th International Conference, Athens, Greece, September 10-14, 2006. Proceedings, Part I 16*, pages 850–859. Springer, 2006.

Hannes Eriksson and Christos Dimitrakakis. Epistemic risk-sensitive reinforcement learning. In *28th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, ESANN 2020, Bruges, Belgium, October 2-4, 2020*, pages 339–344, 2020.

Hannes Eriksson, Debabrota Basu, Mina Alibeigi, and Christos Dimitrakakis. Sentinel: taming uncertainty with ensemble based distributional reinforcement learning. In *Uncertainty in Artificial Intelligence*, pages 631–640. PMLR, 2022.

Jia Lin Hai, Marek Petrik, Mohammad Ghavamzadeh, and Reazul Russel. Rasr: Risk-averse soft-robust mdps with evar and entropic risk. *arXiv preprint arXiv:2209.04067*, 2022.

Wenjie Huang. Data-driven satisficing measure and ranking. *Journal of the Operational Research Society*, 71(3):456–474, 2020.

Yumei Huang, Michael K Ng, and You-Wei Wen. A fast total variation minimization method for image restoration. *Multiscale Modeling & Simulation*, 7(2):774–795, 2008.

Christian D Hubbs, Hector D Perez, Owais Sarwar, Nikolaos V Sahinidis, Ignacio E Grossmann, and John M Wassick. Or-gym: A reinforcement learning library for operations research problems. *arXiv preprint arXiv:2008.06319*, 2020.

Peter J Huber. Robust estimation of a location parameter. In *Breakthroughs in statistics*, pages 492–518. Springer, 1992.

Ramtin Keramati, Christoph Dann, Alex Tamkin, and Emma Brunskill. Being optimistic to be conservative: Quickly learning a cvar policy. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 4436–4443, 2020.

Roger Koenker and Gilbert Bassett Jr. Regression quantiles. *Econometrica: Journal of the Econometric Society*, pages 33–50, 1978.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Chengbo Li, Wotao Yin, Hong Jiang, and Yin Zhang. An efficient augmented lagrangian method with applications to total variation minimization. *Computational Optimization and Applications*, 56:507–530, 2013.

Shiau Hong Lim and Ilyas Malik. Distributional reinforcement learning for risk-sensitive policies. In *Advances in Neural Information Processing Systems*, 2022.

Cheng Liu, Erik-Jan van Kampen, and Guido CHE De Croon. Adaptive risk-tendency: Nano drone navigation in cluttered environments with distributional reinforcement learning. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7198–7204. IEEE, 2023.

David JC MacKay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

A Rupam Mahmood, Dmytro Korenkevych, Gautham Vasan, William Ma, and James Bergstra. Benchmarking reinforcement learning algorithms on real-world robots. In *Conference on Robot Learning*, pages 561–591. PMLR, 2018.

H. B. Mann and D. R. Whitney. On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other. *The Annals of Mathematical Statistics*, 18(1): 50 – 60, 1947.

Borislav Mavrin, Hengshuai Yao, Linglong Kong, Kaiwen Wu, and Yaoliang Yu. Distributional reinforcement learning for efficient exploration. In *International Conference on Machine Learning*, pages 4424–4434. PMLR, 2019.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

Tetsuro Morimura, Masashi Sugiyama, Hisashi Kashima, Hirotaka Hachiya, and Toshiyuki Tanaka. Parametric return density estimation for reinforcement learning. In *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence*, pages 368–375, 2010.

Ted Moskovitz, Jack Parker-Holder, Aldo Pacchiano, Michael Arbel, and Michael Jordan. Tactical optimism and pessimism for deep reinforcement learning. *Advances in Neural Information Processing Systems*, 34:12849–12863, 2021.

Brendan O'Donoghue. Efficient exploration via epistemic-risk-seeking policy optimization. In *Proceedings of the 40th International Conference on Machine Learning*, pages 26382–26402, 2023.

Jihwan Oh, Joonkee Kim, and Se-Young Yun. Risk perspective exploration in distributional reinforcement learning. *ICML 2022 Workshop AI for Agent-Based Modelling*, 2022.

Ian Osband, Charles Blundell, Alexander Pritzel, and Benjamin Van Roy. Deep exploration via bootstrapped dqn. *Advances in Neural Information Processing Systems*, 29, 2016.

LA Prashanth, Cheng Jie, Michael Fu, Steve Marcus, and Csaba Szepesvári. Cumulative prospect theory meets reinforcement learning: Prediction and control. In *International Conference on Machine Learning*, pages 1406–1415. PMLR, 2016.

Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

Mark Rowland, Robert Dadashi, Saurabh Kumar, Rémi Munos, Marc G Bellemare, and Will Dabney. Statistics and samples in distributional reinforcement learning. In *International Conference on Machine Learning*, pages 5528–5536. PMLR, 2019.

Frederik Schubert, Theresa Eimer, Bodo Rosenhahn, and Marius Lindauer. Automatic risk adaptation in distributional reinforcement learning. *arXiv preprint arXiv:2106.06317*, 2021.

Shai Shalev-Shwartz et al. Online learning and online convex optimization. *Foundations and Trends® in Machine Learning*, 4(2):107–194, 2012.

Rahul Singh, Qinsheng Zhang, and Yongxin Chen. Improving robustness via risk averse distributional reinforcement learning. In *Learning for Dynamics and Control*, pages 958–968. PMLR, 2020.

Yunhao Tang and Shipra Agrawal. Exploration by distributional reinforcement learning. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pages 2710–2716, 2018.

Pravin M Vaidya. Speeding-up linear programming using fast matrix multiplication. In *30th Annual Symposium on Foundations of Computer Science*, pages 332–337. IEEE Computer Society, 1989.

Shaun Wang. Premium calculation by transforming the layer premium density. *ASTIN Bulletin: The Journal of the IAA*, 26(1):71–92, 1996.

Zhidong Wang, Chuan Hu, and Xi Zhang. Improving risk-averse distributional reinforcement learning with adaptive risk-seeking exploration. *IFAC-PapersOnLine*, 58(10): 237–242, 2024.

Zhidong Wang, Chongfeng Wei, Xiaolin Tang, Wanzhong Zhao, Chuan Hu, and Xi Zhang. Adaptive risk tendency in uncertainty-aware motion planning using risk-sensitive reinforcement learning. *Advanced Engineering Informatics*, 63:102942, 2025.

Derek Yang, Li Zhao, Zichuan Lin, Tao Qin, Jiang Bian, and Tie-Yan Liu. Fully parameterized quantile function for distributional reinforcement learning. *Advances in Neural Information Processing Systems*, 32, 2019.

Qisong Yang, Thiago D Simão, Simon H Tindemans, and Matthijs TJ Spaan. Safety-constrained reinforcement learning with a distributional safety critic. *Machine Learning*, pages 1–29, 2022.

Jesse Zhang, Brian Cheung, Chelsea Finn, Sergey Levine, and Dinesh Jayaraman. Cautious adaptation for reinforcement learning in safety-critical settings. In *International Conference on Machine Learning*, pages 11055–11065. PMLR, 2020.

Shangtong Zhang and Hengshuai Yao. Quota: The quantile option architecture for reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5797–5804, 2019.

Appendix
DRL-ORA: Distributional Reinforcement Learning with Online Epistemic Risk Adaptation

Yupeng Wu[1] **Wenyun Li**[2] **Wenjie Huang**[2] **Chin Pang Ho**[3]

[1]London Business School
[2]The University of Hong Kong
[3]City University of Hong Kong

A PRELIMINARIES ON THE IQN LOSS FUNCTION

For two samples $q,\ q'$ and the reward r_t at time step t, the TD error is $\delta_t^{q,\,q'} = r_t + \gamma Z_{q'}(s_{t+1}, \pi_\beta(s_{t+1})) - Z_q(s_t,\ a_t)$. The IQN loss function is given by:

$$\mathcal{L}\left(s_t, a_t, r_t, s_{t+1}\right) = \frac{1}{N'} \sum_{i=1}^{N} \sum_{j=1}^{N'} h_q^\kappa \left(\delta_t^{q_i,\, q'_j}\right),$$

where N and N' denote the respective number of i.i.d. samples $q_i,\ q_{j'} \sim U([0,\ 1])$ used to estimate the loss, and h_τ^κ is the quantile estimate generated by the Huber loss function (see Huber [1992]) with a hyper-parameter threshold κ to smooth the non-differentiable point at zero; that is,

$$h_\tau^\kappa \left(\delta_t^{q_i,\, q'_j}\right) = \left| q - \mathbb{I}\left\{\delta_t^{q_i,\, q'_j} < 0\right\}\right| \frac{L_\kappa\left(\delta_t^{q_i,\, q'_j}\right)}{\kappa};$$

$$L_\kappa \left(\delta_t^{q_i,\, q'_j}\right) = \begin{cases} \frac{1}{2}(\delta_t^{q_i,\, q'_j})^2, & \text{if } \left|\delta_t^{q_i,\, q'_j}\right| \le \kappa, \\ \kappa\left(\left|\delta_t^{q_i,\, q'_j}\right| - \frac{1}{2}\kappa\right), & \text{otherwise .} \end{cases}$$

where δ_t is the TD-error, $\mathbb{I}\{\cdot\}$ is the indicator function, and normally $\kappa = 1$ is chosen (see Dabney et al. [2018a]). A corresponding sample-based risk-aware policy is then obtained by approximating Q_β through the average of distorted return distribution samples. Based on the above settings, we are able to estimate the return distribution through a neural network structure, update the network parameters by by minimizing the IQN loss, and implement a risk-aware policy.

B PROOFS FOR TECHNICAL RESULTS

PROOF OF THEOREM 4: Based on Lemma 4.1 of Cesa-Bianchi and Lugosi [2006] and Agarwal et al. [2019], we can instead analyze a slightly different algorithm which draws only a single noise vector $\sigma \sim \exp(\eta)$, rather than drawing a fresh noise vector on every round. Proving regret bounds for this variant translates into asymptotically equivalent (expected) regret bounds for Algorithm 1. Set $l_0 = -\sigma \cdot \alpha$. We have the following several lemmas and proposition hold which are used to prove Theorem 4.

Lemma 7. *The loss function l_t is Lipschitz continuous in α.*

Proof. Use the fact that the sum of Lipschitz functions is also Lipschitz, and the reverse triangle inequality for L_2-norm. □

Denote the Lipschitz continuity modulus as $G > 0$ for $\{l_t\}_{t=0}^T$.

Lemma 8. *By the discretization set $\mathcal{A}' \subseteq \mathcal{A}$, we have from Lipschitz continuity of loss functions that,*

$$\min_{\alpha' \in \mathcal{A}'} \sum_{t=0}^{T} l_t(\alpha') - \min_{\alpha \in \mathcal{A}} \sum_{t=0}^{T} l_t(\alpha) \leq (T+1)G\epsilon.$$

Proof. Denote $\alpha^1 \in \arg\min_{\alpha \in \mathcal{A}} \sum_{t=0}^{T} l_t(\alpha)$ and $\alpha^2 \in \arg\min_{\alpha' \in \mathcal{A}'} \sum_{t=0}^{T} l_t(\alpha')$. We have by Lipschitz continuity of loss function that

$$\left| \sum_{t=0}^{T} l_t(\alpha^1) - \sum_{t=0}^{T} l_t(\alpha^2) \right| \leq (T+1)G|\alpha^1 - \alpha^2|.$$

And we further have

$$\left| \sum_{t=0}^{T} l_t(\alpha^1) - \sum_{t=0}^{T} l_t(\alpha^2) \right| \leq (T+1)G|\alpha^1 - \alpha^2| \leq (T+1)G\epsilon,$$

given that the Hausdorff distance is bounded by ϵ. The proof can be completed given that

$$\min_{\alpha' \in \mathcal{A}'} \sum_{t=0}^{T} l_t(\alpha') - \min_{\alpha \in \mathcal{A}} \sum_{t=0}^{T} l_t(\alpha) \geq 0.$$

□

Lemma 9. *The regret is upper bounded via,*

$$\sum_{t=0}^{T} l_t(\alpha'_t) - \min_{\alpha' \in \mathcal{A}'} \sum_{t=0}^{T} l_t(\alpha') \leq \sum_{t=0}^{T} l_t(\alpha'_t) - \sum_{t=0}^{T} l_t(\alpha'_{t+1}).$$

Proof. The proof is inspired by the proof of Lemma 2.1 in Shalev-Shwartz et al. [2012]. Subtracting $\sum_t l_t(\alpha'_t)$ from both sides of the inequality and rearranging, the desired inequality can be rewritten as

$$\sum_{t=0}^{T} l_t(\alpha'_{t+1}) \leq \min_{\alpha' \in \mathcal{A}'} \sum_{t=0}^{T} l_t(\alpha').$$

We prove this inequality by induction. The base case of $T = 0$ follows directly from the definition of α'_{t+1}. Assume the inequality holds for $T-1$, then for all $\alpha' \in \mathcal{A}'$, we have

$$\sum_{t=0}^{T-1} l_t(\alpha'_{t+1}) \leq \sum_{t=0}^{T-1} l_t(\alpha').$$

Adding $l_T(\alpha'_{T+1})$ to both sides we get

$$\sum_{t=0}^{T} l_t(\alpha'_{t+1}) \leq l_T(\alpha'_{T+1}) + \sum_{t=0}^{T-1} l_t(\alpha').$$

The above holds for all α' and in particular for $\alpha' = \alpha'_{T+1}$. Thus,

$$\sum_{t=0}^{T} l_t(\alpha'_{t+1}) \leq \sum_{t=0}^{T} l_t(\alpha'_{T+1}) = \min_{\alpha' \in \mathcal{A}'} \sum_{t=0}^{T} l_t(\alpha'),$$

where the last equation follows from the definition of α'_{T+1}. This concludes our inductive argument. □

By combining Lemma 8 and 9, we have

$$\sum_{t=0}^{T} l_t(\alpha'_{t+1}) \leq \min_{\alpha \in \mathcal{A}} \sum_{t=0}^{T} l_t(\alpha) + (T+1)G\epsilon.$$

We further have

$$\sum_{t=0}^{T} l_t(\alpha'_t) - \min_{\alpha \in \mathcal{A}} \sum_{t=0}^{T} l_t(\alpha) \leq \sum_{t=0}^{T} l_t(\alpha'_t) - \sum_{t=0}^{T} l_t(\alpha'_{t+1}) + (T+1)G\epsilon. \tag{5}$$

Lemma 10. *The expected regret of Algorithm 1, from inequality (5) has*

$$\begin{aligned}&\mathbb{E}\left[\sum_{t=1}^{T} l_t(\alpha'_t) - \min_{\alpha\in\mathcal{A}}\sum_{t=1}^{T} l_t(\alpha)\right]\\ \leq&\mathbb{E}\left[-\sigma\alpha^* + \sigma\alpha'_1\right] + \sum_{t=1}^{T}\mathbb{E}\left[l_t(\alpha'_t) - l_t(\alpha'_{t+1})\right] + (T+1)G\epsilon, \end{aligned} \tag{6}$$

where $\alpha^* \in \arg\min\{\sum_{t=1}^{T} l_t(\alpha),\ \alpha\in\mathcal{A}\}$.

Proof. Using the result of Lemma 8 and 9, we have

$$\sum_{t=0}^{T} l_t(\alpha'_t) - \min_{\alpha\in\mathcal{A}}\sum_{t=0}^{T} l_t(\alpha) - (T+1)G\epsilon \leq \sum_{t=0}^{T} l_t(\alpha'_t) - \min_{\alpha'\in\mathcal{A}'}\sum_{t=0}^{T} l_t(\alpha') \leq \sum_{t=0}^{T} l_t(\alpha'_t) - \sum_{t=0}^{T} l_t(\alpha'_{t+1}).$$

By rearranging the above terms (taking the l_0 out), and taking expectation over both sides of the inequality, we can get the desired result. □

Proposition 11. *The first two terms at the right hand side of the inequality (6) have a complexity bound* $O(T^{1/2})$.

Proof. The proof is inspired by the proof of Theorem 1 in Agarwal et al. [2019]. For any two functions $f_1,\ f_2 : \mathcal{A}' \to \mathbb{R}$ and scalar $\sigma_1,\ \sigma_2 \in \mathbb{R}$, let

$$\alpha_i(\sigma_i) \in \arg\min_{\alpha\in\mathcal{A}'}\left\{f_i(\alpha) - \sigma_i\alpha\right\},\ i = 1, 2.$$

Let $f = f_1 - f_2$ and $\sigma = \sigma_1 - \sigma_2$, we have that

$$f_1(\alpha_1(\sigma_1)) - \sigma_1\alpha_1(\sigma_1) \leq f_1(\alpha_2(\sigma_2)) - \sigma_1\alpha_2(\sigma_2),$$

and

$$f_2(\alpha_2(\sigma_2)) - \sigma_2\alpha_2(\sigma_2) \leq f_1(\alpha_1(\sigma_1)) - \sigma_2\alpha_1(\sigma_1).$$

By adding the above two inequalities and rearranging, we have

$$f_1(\alpha_1(\sigma_1)) - f_2(\alpha_2(\sigma_2)) \leq \sigma(\alpha_1(\sigma_1) - \alpha_2(\sigma_2)).$$

For any round t, consider substituting $f_1(\alpha) = \sum_{i<t} l_i(\alpha)$, $f_2(\alpha) = \sum_{i<t+1} l_i(\alpha)$, $\sigma_2 = \sigma$ and $\sigma_1 = \sigma' = \sigma + 2G$. We immediately get that

$$l_t(\alpha_t(\sigma')) - l_t(\alpha_{t+1}(\sigma')) \leq 2G(\alpha_t(\sigma') - \alpha_{t+1}(\sigma)).$$

Using the fact that l_t is G-Lipschitz, we get that

$$-G|\alpha_t(\sigma') - \alpha_{t+1}(\sigma)| \leq l_t(\alpha_t(\sigma')) - l_t(\alpha_{t+1}(\sigma')) \leq 2G(\alpha_t(\sigma') - \alpha_{t+1}(\sigma)),$$

which immediately implies that $\alpha_t(\sigma') \geq \alpha_{t+1}(\sigma)$. Similar calculations show that $\alpha_{t+1}(\sigma') \geq \alpha_t(\sigma)$ and $\alpha_t(\sigma') \geq \alpha_t(\sigma)$. In the following proof, we omit the dependence on t. We denote by $\alpha_{\min}(\sigma) = \min\{\alpha_t(\sigma),\ \alpha_{t+1}(\sigma)\}$, $\alpha_{\max}(\sigma) = \max\{\alpha_t(\sigma),\ \alpha_{t+1}(\sigma)\}$. First we observe that

$$\mathbb{E}[|\alpha_t(\sigma) - \alpha_{t+1}(\sigma)|] = \mathbb{E}[\alpha_{\max}(\sigma)] - \mathbb{E}[\alpha_{\min}(\sigma)].$$

Secondly the computation above implies that

$$\alpha_{\min}(\sigma') \geq \alpha_{\max}(\sigma). \tag{7}$$

Letting $\sigma' = \sigma + 2G$, we have

$$\begin{aligned}\mathbb{E}[\alpha_{\min}(\sigma)] =& \int_{\sigma=0}^{2G} \eta\exp(-\eta\sigma)\alpha_{\min}(\sigma)d\sigma + \int_{\sigma>2G} \eta\exp -\eta\sigma\alpha_{\min}(\sigma)d\sigma\\ \geq&(1-\exp(-2\eta G))(\mathbb{E}[\alpha_{\max}(\sigma)] - 1) + \int_{\sigma>0}\eta\exp(-\eta\sigma')\alpha_{\min}(\sigma')d\sigma\\ \geq&(1-\exp(-2\eta G))(\mathbb{E}[\alpha_{\max}(\sigma)] - 1) + \int_{\sigma>0}\eta\exp(-\eta\sigma')\alpha_{\max}(\sigma)d\sigma\\ =&(1-\exp(-2\eta G))(\mathbb{E}[\alpha_{\max}(\sigma)] - 1) + \exp(-2\eta G)\mathbb{E}[\alpha_{\max}(\sigma)]\\ =&\mathbb{E}[\alpha_{\max}(\sigma)] - (1-\exp(-2\eta G)) \geq \mathbb{E}[\alpha_{\max}(\sigma)] - 2\eta G,\end{aligned}$$

where the second inequality uses inequality (7) and and the last inequality uses the inequality $\exp(x) \geq 1 + x$.

Finally we have that

$$\begin{aligned}\mathbb{E}\left[-\sigma\alpha^* + \sigma\alpha'_1\right] + \sum_{t=1}^{T}\mathbb{E}\left[l_t(\alpha'_t) - l_t(\alpha'_{t+1})\right] \leq & \mathbb{E}[\sigma] + G\sum_{t=1}^{T}\mathbb{E}[\|\alpha'_t - \alpha'_{t+1}\|_1] \\ & \leq 1/\eta + 2\eta G^2 T.\end{aligned}$$

And by choose $\eta = T^{-1/2}$, we get the desired result. □

Finally, if we choose $\epsilon = O(T^{-1/2})$ in inequality (6), then the whole expected regret preserves the $O(T^{1/2})$ complexity. We thus complete the proof of Theorem 4.

PROOF OF THEOREM 5: From Problem (4) in main paper, we have

$$\begin{aligned}&\max\{1-\alpha:\ \text{CVaR}_\alpha(X_{t-1}-\tau_t)\leq 0\}\\ =&\max\left\{1-\alpha:\ \exists y\leq 0:\ y+\alpha^{-1}\mathbb{E}\left[(X_{t-1}-\tau_t-y)_+\right]\leq 0\right\}\\ =&\max\left\{1-\alpha:\ \exists y\leq 0:\ -1+\alpha^{-1}\mathbb{E}\left[(-(X_{t-1}-\tau_t)/y+1)_+\right]\leq 0\right\}\\ =&\max\left\{1-\alpha:\ \exists y\leq 0:\ 1-\alpha\leq 1-\mathbb{E}\left[(-(X_{t-1}-\tau_t)/y+1)_+\right]\right\}\\ =&\max_{b\geq 0}\left\{1-\mathbb{E}\left[(b(X_{t-1}-\tau_t)+1)_+\right]\right\}.\end{aligned}$$

Thus problem $\max\{\alpha:\ \text{CVaR}_\alpha(X_{t-1})\leq\tau_t\}$ is equivalent to

$$\min_{b\geq 0}\mathbb{E}\left[(b(X_{t-1}-\tau_t)+1)_+\right]. \tag{8}$$

PROOF OF THEOREM 6: Problem (8) can be approximated by a constrained linear program with $O(K)$ decision variables and $O(K)$ constraints, using sample average approximation. Define $\boldsymbol{c}\in\mathbb{R}^K$, and we have:

$$\begin{aligned}\min_{\boldsymbol{c},b\geq 0}\ & \frac{1}{K}\sum_{k=1}^{K}c_k & \\ \text{s.t.}\ & c_k\geq b(Q_{t-1}^{\theta_k}-\tau_t)+1, & \forall k=1,...,K,\\ & c_k\geq 0, & \forall k=1,...,K.\end{aligned}$$

where $Q_{t-1}^{\theta_k}$ is the output of the k-th ensemble network at period $t-1$ (we omit its dependence on state-action pair for simplicity). Using the structure of Problem (8), we can further reduce the computation. The approximation of Problem (8) is given by

$$\min_{b\geq 0}\frac{1}{K}\sum_{k=1}^{K}\left[b(Q_{t-1}^{\theta_k}-\tau_t)+1\right]_+, \tag{9}$$

which contains the sum of positive part functions where each positive part has slope $Q_{t-1}^{\theta_k}-\tau_t$ and each function pass the point $(0,\ 1)$. Problem (9) can be graphically illustrated by a simple case in Figure 5, where there are exactly two functions, both with positive and negative slopes, respectively. Given $b>0$, suppose $S_1+S_2\geq S_3+S_4\geq 0$, then we must have $\alpha_t=1$, otherwise, we will find the smallest b-intercept (i.e., b_1 and b_2), such that the sum of function gaps below 1 is lower than that above 1. Here we would check if at b_2, that $S_1+S_2>S_3$ holds.

By generalizing the above idea into the general case, we can design a search algorithm: Algorithm 3 to find α_t efficiently.

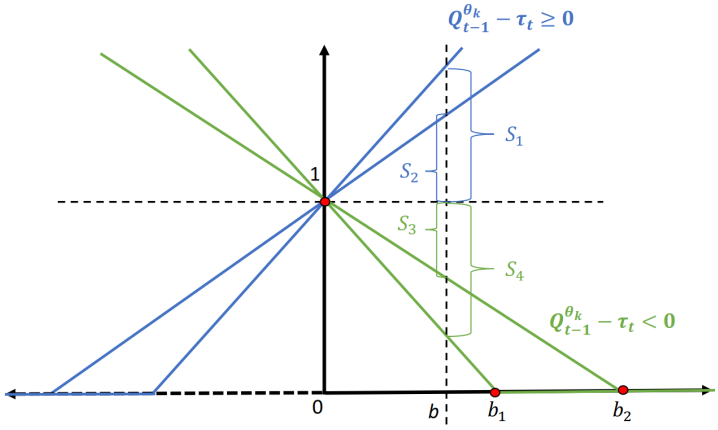


Figure 5: Graphic Illustration of Problem (9)

Algorithm 2 Search Algorithm for α_t

Denote the index set $[K']$ where for each $k \in [K']$ where $Q_{t-1}^{\theta_k} - \tau_t < 0$.
if $\sum_{k\in[K]/[K']}(Q_{t-1}^{\theta_k} - \tau_t) \geq \sum_{k\in[K']}(\tau_t - Q_{t-1}^{\theta_k})$, **then**
Set $\alpha_t = 1$,
else
Redefine and reorder the elements (the b-intercepts) in $[K']$ such that

$$\frac{1}{\tau_t - Q_{t-1}^{\theta_1}} \leq \cdots \leq \frac{1}{\tau_t - Q_{t-1}^{\theta_{|[K']|}}}.$$

Then, $b^* = 1/(\tau_t - Q_{t-1}^{\theta_{k^*}})$, where $k^* \in \{1, ..., |[K']|\}$ and

$$\sum_{i=1}^{k^*+1}(\tau_t - Q_{t-1}^{\theta_{k^*}}) \leq \sum_{k\in[K]/[K']}(Q_{t-1}^{\theta_k} - \tau_t) < \sum_{i=1}^{k^*}(\tau_t - Q_{t-1}^{\theta_{k^*}}).$$

end if

The most computational consuming step in Algorithm is the sorting b-intercepts in the set $[K']$ which has complexity $O(K\log(K))$, which much lower than that of the best known algorithm for linear program $O(K^{2.5})$ (see Vaidya [1989]).

C HANDLING THE QUANTILE DISTORTION CASE

The distortion function in Example 3, $g_\alpha(x) = \mathbb{I}\{x \leq \alpha\}$, is not Lipschitz continuous in α, but is right-continuous with at most finitely many discontinuity points on the grid $\mathcal{A}'$. Our general analysis (Lemma 7) assumes Lipschitz continuity as a sufficient condition; here we clarify that the quantile case is handled by a standard discretization argument without undermining the regret bounds.

Lemma 8 controls the error from discretizing the continuous domain $\mathcal{A}$ into a finite grid $\mathcal{A}'$. If the true optimal α^* falls exactly on a discontinuity point, selecting the nearest grid point incurs at most $(T+1)G\varepsilon$ additional error—the standard cost of discretization. By choosing the grid spacing ε sufficiently small, this error can be made arbitrarily small.

In practice, we set $\alpha_{\min} > 0$ and use a dense grid, approaching any discontinuity points from the right. This induces no significant regret penalty, and the overall regret bound retains its $O(T^{1/2})$ rate. We therefore report the Lipschitz condition as a sufficient rather than necessary assumption, with the quantile distortion covered by this discretization argument.

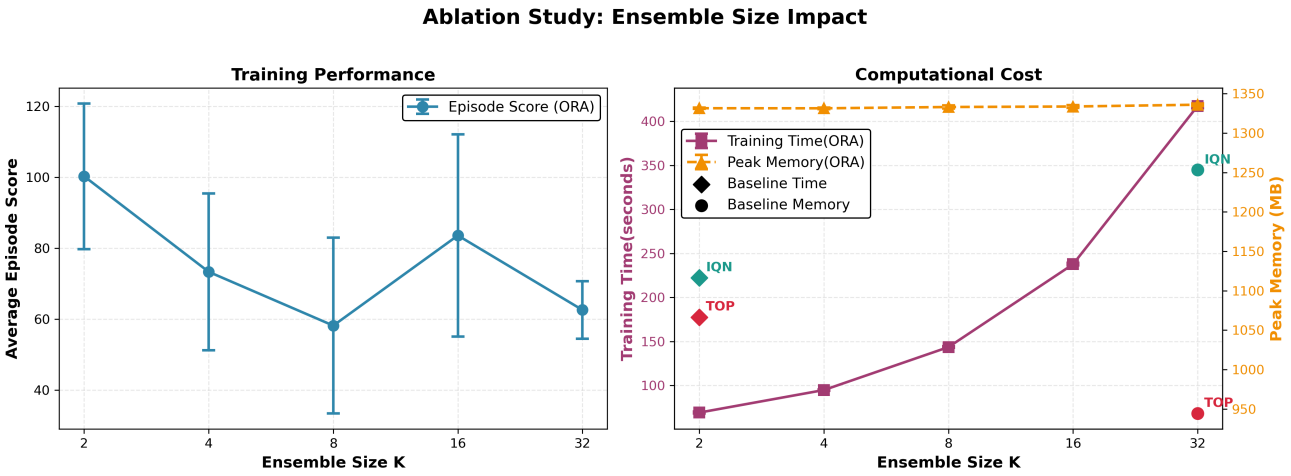


Figure 6: Impact of ensemble size on ORA's training performance and computational cost in CartPole environment. Results averaged over 3 runs per configuration (N = 2, 4, 8, 16, 32). Error bars show standard deviation. The time and memory overheads of IQN and TOP are indicated on the left and right sides of the second subplot, respectively.

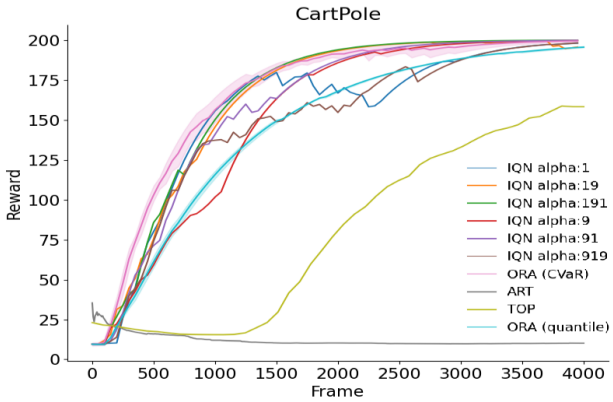
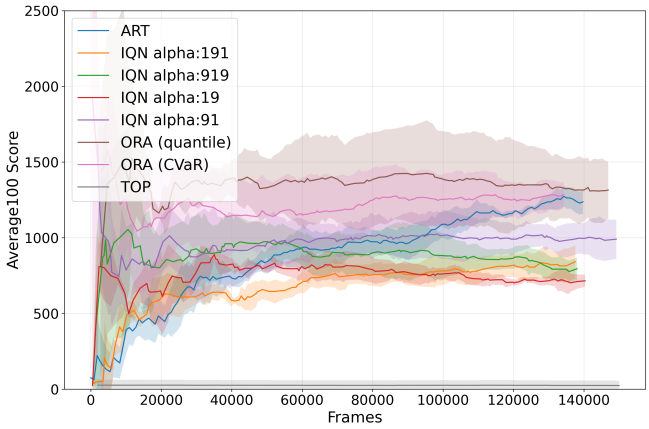


Figure 7: ORA with 90% confidence interval.

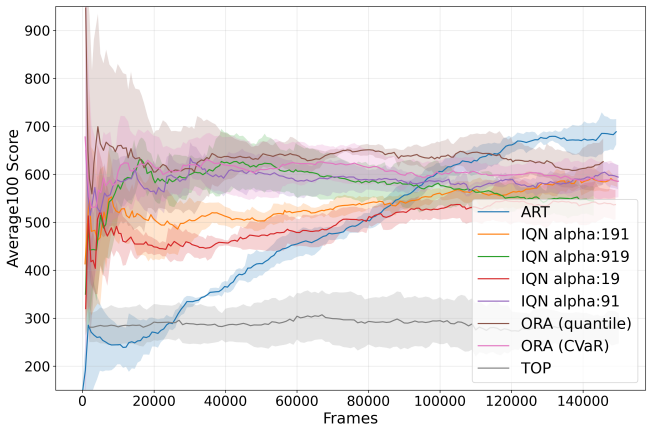
D SUPPLEMENTARY MATERIALS FOR EXPERIMENTS

Computational Test Although the ensemble network is inseparable from our method, making it impossible to investigate the ablation improvement of the ensemble network, the test on ensemble size K demonstrates that the ensemble network itself does not exhibit a significant correlation with performance improvement (Figure 6). Further computational tests reveal that the time overhead grows only slightly with the ensemble size K, while the associated memory overhead is negligible. This demonstrates that introducing the ensemble network does not incur a substantial computational burden. As a reference, we have also included the average time and space consumption of the TOP and IQN methods (the ensemble size is not applicable). The comparison further confirms that our method only requires slightly higher memory usage, and with the default K, the additional computational time is insignificant. Note that aforementioned experiments are conducted on the same GPU-accelerated device. Despite variations in model architectures of these approaches and consequently their hyperparameter configurations (which are specified in the following subsection), a uniform batch size of 64 was employed for all methods, with each using the parameter set determined to be optimal for it.

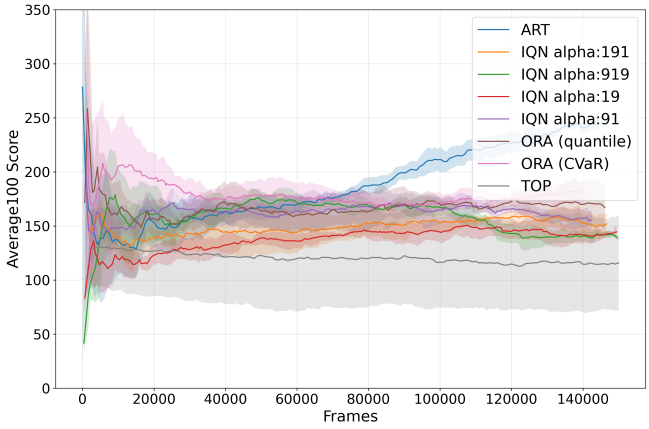
Table 3 records ORA's left-truncated variance (LTV), risk parameters, and median epistemic uncertainty per training period for all three experiments. The experimental results of ORA (with distortion function CVaR and quantile) and baselines are shown in Figure 8. Across three diverse Atari environments, our ORA method demonstrates statistically significant dominance ($p < 0.001$), achieving massive to large Rank-Biserial effect sizes against the TOP baseline (0.996 in Hero, 0.875 in MsPacman, 0.566 in SpaceInvaders) and substantial effects against ART (0.672, 0.511, 0.388, respectively). We note that no single algorithm universally dominates across all RL benchmarks. The ORA framework is designed to be adaptive—its



(a) Hero



(b) MsPacman



(c) SpaceInvaders

Figure 8: The average score of last 100 episodes curves (mean $\pm$ std) of ORA, ART, TOP, and baseline algorithms across three Atari environments over 3 seeds.

gains are most prominent in environments where epistemic uncertainty evolves substantially during training. For these three environments, the discretization of $\mathcal{A}'$ follows a grid size accuracy of 5, while in Nano Drone and knapsack, the grid size is bucketed to 1. The perturbation parameter η for the FTPL algorithm is set to 0.5 in all experiments. We conducted an ablation study to verify the sensitivity of ORA to this choice. Table 2 presents results for $\eta \in [0.01, 5.0]$ across CartPole, Knapsack, and Nano Drone environments. The key finding is that ORA performance is robust to η over a broad range: varying η in $[0.05, 2.00]$ changes the final best score by less than 5% across all tested environments, including the partially observable Nano Drone task with a starting score of -40.

The hyper-parameter settings for CartPole, Atari games, Knapsack and Nano Drone Navigation tasks are listed in Table 4, 5 and 6. Parameters not listed are as default in IQN. The implementation and hyper-parameter settings of ART and TOP in Atari games and Knapsack environments follows the open source [Moskovitz et al., 2021, Liu et al., 2023]. The experiment was conducted on a Windows platform with an Intel i5-13400F and an Nvidia RTX 5060. The dependencies and libraries required for each task can be found in the code appendix.

Table 2: Ablation study on perturbation noise scalar η. Results reported as mean $\pm$ std over 5 seeds.

Environment	$\eta = 0.01$	$\eta = 0.05$	$\eta = 0.10$	$\eta = 0.50$	$\eta = 1.00$	$\eta = 2.00$
CartPole	$188.27_{\pm 50.74}$	$186.55_{\pm 71.32}$	$187.83_{\pm 20.40}$	$\mathbf{201.30_{\pm 26.40}}$	$188.93_{\pm 40.79}$	$190.40_{\pm 38.69}$
Knapsack	$445.03_{\pm 107.77}$	$\mathbf{480.84_{\pm 31.13}}$	$462.38_{\pm 41.69}$	$470.08_{\pm 35.71}$	$463.21_{\pm 23.89}$	$452.65_{\pm 37.73}$
Nano Drone	$9.02_{\pm 18.83}$	$\mathbf{12.39_{\pm 17.66}}$	$7.09_{\pm 8.32}$	$8.40_{\pm 9.76}$	$10.76_{\pm 14.53}$	$7.93_{\pm 9.50}$

Table 5: Hyper-parameters in Knapsack

Parameter	Value
Optimizer	Adam
Learning rate	1e-3
Batch size	32
Replay buffer size	4096
Target network update speed	0.01
Discount factor	1.0
Epsilon greedy	0.1 (random action)
Ensemble networks	10
Layer size	64
Number of episodes	50000

Table 3: Relation between risk parameters and epistemic uncertainty

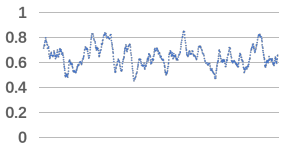
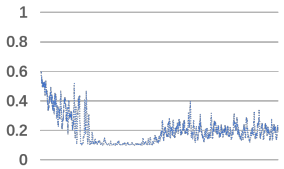
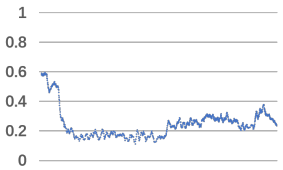
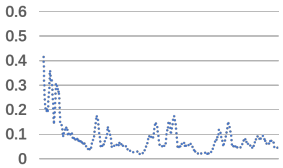
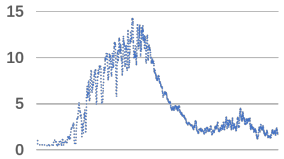
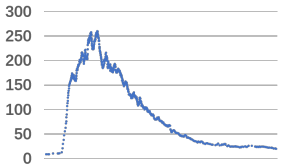
	CartPole	Nano Drone Navigation	Knapsack
Risk parameter α			
Epistemic uncertainty (LTV)			

Table 4: Hyper-parameters in CartPole and Atari game environments

Parameter	Value
Optimizer	Adam
Learning rate	0.03
Batch size	8
Replay buffer size	100000
Target network update speed	1
Discount factor	0.99
Epsilon greedy	0.1 (random action)
Ensemble networks	32
Layer size	256
Number of distortion samplings	64
Number of samplings for quantile loss approximation	8

Table 6: Hyper-parameters in Nano Drone Navigation

Parameter	Value
Optimizer	Adam
Learning rate	2×10^{-4}
Experience replay buffer with size	6×10^{4}
Discount factor	0.99
State update period	0.1 $[s]$
Maximum velocity	1 $[m/s]$
Radius of drone	0.05 $[m]$
Goal distance	6 $[m]$
Goal-reaching threshold	0.5 $[m]$
Safety margin	0.2 $[m]$
Number of episodes	1000
Number of distortion samplings	64
Batch size	32
Maximum time-steps	200
Number of samplings for quantile loss approximation	16
Number of velocity magnitudes	3
Lower bound on CVaR risk parameter	0.1
Learning rate in ART	0.5
Number of hidden layers	3
Number of units per layer	512
Ensemble networks	10