Bridging Distributional and Risk-Sensitive Reinforcement Learning: Balancing Statistical, Computational, and Risk Considerations

Hao Liang [1]

Abstract

High-stakes applications like finance and healthcare require risk-sensitive methods that maximize a risk measure of the return distribution. Existing risk-sensitive reinforcement learning faces computational and statistical challenges due to the non-linearity of risk measures. This paper proposes computationally efficient distributional reinforcement learning (DRL) algorithms with regret guarantees, addressing these challenges. In particular, we introduce two variants of the principled DRL algorithm, `RODI` (Liang & Luo, 2022), that use a novel distribution representation and projection method, maintaining regret bound while keeping computational efficiency. Our algorithms, `RODI-Rep`, demonstrate improved regret performance compared to traditional non-distributional RL methods through theoretical analysis and empirical validation.

1. Introduction

Standard reinforcement learning (RL) aims to develop optimal policies that maximize expected returns, often described as risk-neutral RL due to its focus on the average outcomes of return distributions (Sutton & Barto, 2018). However, in high-stakes environments such as finance (Davis & Lleo, 2008; Bielecki et al., 2000), medical treatment (Ernst et al., 2006), and operations research (Delage & Mannor, 2010), decision-makers frequently prioritize risk-sensitive measures that account for return distribution variability.

Originating from the foundational work of Howard & Matheson (1972), risk-sensitive reinforcement learning (RSRL) utilizing the exponential risk measure (ERM) has been extensively applied across various sectors (Shen et al., 2014; Nass et al., 2019; Hansen & Sargent, 2011). The ERM facilitates a balance between expected return and variance, with adjustable risk sensitivity through a risk parameter. However, the non-linear nature of ERM typically necessitates complex algorithmic solutions.

Distributional reinforcement learning (DRL) has outperformed traditional methods in several challenging risk-neutral scenarios (Bellemare et al., 2017; Dabney et al., 2018b;a), by learning the full return distribution rather than a scalar value function. This distributional insight offers a unique advantage in optimizing risk measures beyond mere expectations (Dabney et al., 2018a; Singh et al., 2020; Ma et al., 2020). Despite this, existing RSRL implementations using DRL lack comprehensive regret analysis (Dabney et al., 2018a; Ma et al., 2021; Achab & Neu, 2021), limiting their evaluative and enhancement capabilities in terms of sample efficiency.

Recently, Liang & Luo (2022) introduced a new DRL algorithm, `RODI`, with near-optimal regret bounds that bridge the gap between DRL and RSRL regarding sample efficiency. Yet, the algorithm faces computational challenges due to the infinite-dimensional nature of distributions, as highlighted in our study. This introduces a crucial question:

Is it feasible for DRL to achieve near-optimal regret in RSRL while maintaining computational efficiency?

Our work provides a positive answer by developing *computationally efficient* DRL algorithms with regret guarantees. We propose two variants of `RODI`, with computational efficiency and principled exploration strategies for tabular ERM-MDPs. These algorithms incorporate the principle of optimism in the face of uncertainty (OFU) at a distributional level, adeptly managing the exploration-exploitation trade-off. Thus, we effectively bridge the computational and sample complexity gap between DRL and RSRL. Our contributions advance the understanding and efficiency of RSRL through a distributional perspective.

1.1. Related Work

The field of DRL has seen significant growth since the pioneering work by (Bellemare et al., 2017). Numerous studies

[1]School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen. Correspondence to: Hao Liang <haoliang1@link.cuhk.edu.cn>.

Workshop on Foundations of Reinforcement Learning and Control at the 41st International Conference on Machine Learning, Vienna, Austria.

have focused on enhancing performance in risk-neutral environments (see Rowland et al., 2018; Dabney et al., 2018b;a; Barth-Maron et al., 2018; Yang et al., 2019; Lyle et al., 2019; Zhang et al., 2021). However, efforts to incorporate risk-sensitive behaviors are relatively scarce, with notable contributions including Dabney et al. (2018a); Ma et al. (2021); Achab & Neu (2021).

A substantial volume of research on RSRL has explored the use of the ERM across various contexts (Borkar, 2001; 2002; Borkar & Meyn, 2002; Borkar, 2010; Bäuerle & Rieder, 2014; Di Masi et al., 2000; Di Masi & Stettner, 2007; Cavazos-Cadena & Hernández-Hernández, 2011; Jaśkiewicz, 2007; Ma et al., 2020; Mihatsch & Neuneier, 2002; Osogami, 2012; Patek, 2001; Shen et al., 2013; 2014). These investigations typically deal with known transitions and rewards or are situated within infinite-horizon settings, often overlooking sample complexity considerations.

Recent works by Fei et al. (2020) and Fei et al. (2021) delve into RSRL employing ERM within the same framework. Fei et al. (2020) introduced the first regret-guaranteed algorithms for risk-sensitive episodic Markov decision processes (MDPs), but their regret upper bounds contain an additional factor of $\exp(|\beta|H^2)$ and their lower bound proof contains errors, leading to a weaker bound. Fei et al. (2021) refined their algorithm by introducing a doubly decaying bonus that effectively removes the $\exp(|\beta|H^2)$ factor, but the issue with the lower bound was not resolved. Liang & Luo (2022) further advances the field by proposing the `RODI` algorithm, a principled DRL framework that achieves near-optimal regret bounds and establishes a tight minimax lower bound. However, the practical application of `RODI` is hampered by computational inefficiencies.

1.2. Contributions

This paper makes the following primary contributions:

- We introduce an novel method for distribution representation and projection specifically designed to mitigate the computational inefficiencies encountered in the existing DRL algorithm, `RODI`.
- We propose the `RODI-Rep` algorithm, an enhancement of `RODI` that integrates the distribution representation and projection techniques. This algorithm maintains the same regret bound as `RODI` and enjoys high computational efficiency (see Figure 1).
- We provide both theoretical and empirical validations of the advantages posed by `RODI-Rep` compared to traditional non-distributional RL algorithms.

2. Preliminaries

Notations We use $\mathbb{I}\{\cdot\}$ to denote the indicator function. For any $x \in \mathbb{R}$, we define $[x]^+ \triangleq \max\{x, 0\}$. We denote by δ_c the Dirac measure at c. We denote by $\mathcal{D}(a, b)$, $\mathcal{D}_M$ and $\mathcal{D}$ the set of distributions supported on $[a, b]$, $[0, M]$ and the set of all distributions respectively. For a discrete set $x = \{x_1, \cdots, x_n\}$ and a probability vector $p = (p_1, \cdots, p_n)$, the notation (x, p) represents the discrete distribution with $\mathbb{P}(X = x_i) = p_i$. We use $(x_1, x_2; p)$ to denote a binary r.v. taking values x_1 and x_2 with probability $1 - p$ and p. For a discrete distribution $\eta = (x, p)$, we use $|\eta| = |x|$ to denote the number of atoms of the distribution η. We use $\tilde{\mathcal{O}}(\cdot)$ to denote $\mathcal{O}(\cdot)$ omitting logarithmic factors.

Episodic MDP An episodic MDP is identified by $\mathcal{M} \triangleq (\mathcal{S}, \mathcal{A}, (P_h)_{h\in[H]}, (r_h)_{h\in[H]}, H)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ the action space, $P_h : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ the probability transition kernel at step h, $r_h : \mathcal{S} \times \mathcal{A} \to [0, 1]$ the collection of reward functions at step h and H the length of one episode. The agent interacts with the environment for K episodes. At the beginning of episode k, Nature selects an initial state s_1^k arbitrarily. In step h, the agent takes action a_h^k and observes reward $r_h(s_h^k, a_h^k)$ and reaches the next state $s_{h+1}^k \sim P_h(\cdot|s_h^k, a_h^k)$. The episode terminates at $H + 1$ with $r_{H+1} = 0$, then the agent proceeds to next episode.

Entropic risk measure and exponential utility We direct our attention to ERM, a prominent risk measure in risk-sensitive decision-making, such as mathematical finance (Föllmer & Schied, 2016), Markovian decision processes (Howard & Matheson, 1972; Bäuerle & Rieder, 2014). For a random variable $X \sim F$ and a non-zero coefficient β, the ERM is defined as:

$$U_\beta(X) \triangleq \frac{1}{\beta} \log\left(\mathbb{E}_{X\sim F}\left[e^{\beta X}\right]\right) = \frac{1}{\beta} \log\left(\int_{\mathbb{R}} e^{\beta X} dF(x)\right).$$

We denote $U_\beta(F)$ as $U_\beta(X)$ for $X \sim F$. When β possesses a small absolute value, employing Taylor's expansion yields

$$U_\beta(X) = \mathbb{E}[X] + \frac{\beta}{2}\mathbb{V}[X] + \mathcal{O}(\beta^2). \tag{1}$$

Therefore, a decision-maker aiming to maximize the ERM value demonstrates risk-seeking behavior (preferring higher uncertainty in X) when $\beta > 0$, and risk-averse behavior (preferring lower uncertainty in X) when $\beta < 0$. The absolute value of β dictates the risk sensitivity, with the measure converging to the mean functional as β approaches zero.

ERM is closed related to the Exponential Utility (EU):

$$E_\beta(F) \triangleq e^{\beta U_\beta(F)} = \int_{\mathbb{R}} e^{\beta x} dF(x).$$

The equivalence between ERM and EU in terms of optimal policies is a critical aspect leveraged by Fei et al. (2021) and Liang & Luo (2022) to facilitate their regret analysis.

Algorithm	Regret bound	Time	Space
RSVI (Fei et al., 2020)	$\tilde{\mathcal{O}}\left(\exp(\lvert\beta\rvert H^2)\frac{\exp(\lvert\beta\rvert H)-1}{\lvert\beta\rvert}\sqrt{HS^2AT}\right)$	$\mathcal{O}\left(TS^2A\right)$	$\mathcal{O}\left(HSA+T\right)$
RSVI2 (Fei et al., 2021)	$\tilde{\mathcal{O}}\left(\frac{\exp(\lvert\beta\rvert H)-1}{\lvert\beta\rvert}\sqrt{HS^2AT}\right)$		
RODI-Rep (ours)			
RODI (Liang & Luo, 2022)		$\mathcal{O}(KS^H)$	$\mathcal{O}(S^H)$
lower bound (Liang & Luo, 2022)	$\Omega\left(\frac{\exp(\beta H/6)-1}{\beta}\sqrt{SAT}\right)$	-	-

Table 1. Regret bounds and computational complexity comparisons.

2.1. Risk-sensitive Distributional Dynamic Programming Revisited

We revisits the Distributional Dynamic Programming (DDP) framework for risk-sensitive control proposed in Liang & Luo (2022). To start with, We define the return for a policy π starting from state-action pair (s,a) at step h

$$Z_h^\pi(s,a) \triangleq \sum_{h'=h}^{H} r_{h'}(s_{h'},a_{h'}) \mid s_h=s, a_{h'}=\pi_{h'}(s_{h'}).$$

Define $Y_h^\pi(s) \triangleq Z_h^\pi(s,\pi_h(s))$, then it is immediate that

$$Z_h^\pi(s,a) = r_h(s,a) + Y_{h+1}^\pi(S'), S' \sim P_h(\cdot|s,a).$$

There are two sources of randomness in $Z_h^\pi(s,a)$: the transition P_h^π and the next-state return Y_{h+1}^π. Denote by $\nu_h^\pi(s)$ and $\eta_h^\pi(s,a)$ the cumulative distribution function (CDF) corresponding to $Y_h^\pi(s)$ and $Z_h^\pi(s,a)$ respectively. Rewriting the random variable in the form of CDF, we have the distributional Bellman equation

$$\eta_h^\pi(s,a) = \sum_{s'} P_h(s'|s,a)\nu_{h+1}^\pi(s')(\cdot - r_h(s,a)),$$
$$\nu_h^\pi(s) = \eta_h^\pi(s,\pi_h(s)).$$

The risk-sensitive action-value functions of a policy π at step h are defined as

$$Q_h^\pi(s,a) \triangleq U_\beta(Z_h^\pi(s,a)), V_h^\pi(s) \triangleq Q_h^\pi(s,\pi_h(s)).$$

We focus on the *risk-sensitive control* setting, in which the goal is to find an optimal policy to maximize the risk-sensitive value function

$$\pi^*(s) \triangleq \arg\max_{(\pi_1,\ldots,\pi_H)\in\Pi} V_1^{\pi_1\cdots\pi_H}(s).$$

In the risk-sensitive setting, however, the principle of optimality does not always hold for general risk measures. For example, the optimal policy for CVaR may be non-Markovian or history-dependent (Shapiro et al., 2021). By identifying certain properties of ERM, Liang & Luo (2022) establishes the *distributional Bellman optimality equation* in the risk-sensitive setting. In particular, the optimal policy π^* is given by the following backward recursions:

$$\nu_{H+1}^*(s) = \psi_0,\ \eta_h^*(s,a) = [P_h\nu_{h+1}^*](s,a)(\cdot - r_h(s,a)),$$
$$\pi_h^*(s) = \arg\max_{a\in\mathcal{A}} U_\beta(\eta_h^*(s,a)), \nu_h^*(s) = \eta_h^*(s,\pi_h^*(s)), \tag{2}$$

where $F(\cdot - c)$ denotes the CDF obtained by shifting F to the right by c. The sequence $(\eta_h^*)_{h\in[H]}$ and $(\nu_h^*)_{h\in[H]}$ represent the sequence of distributions corresponding to the optimal returns in each step.

For simplicity, we define the *distributional Bellman operator* $\mathcal{T}(P,r) : \mathcal{D}^{\mathcal{S}} \to \mathcal{D}^{\mathcal{S}\times\mathcal{A}}$ with associated model $(P,r) = (P(s,a), r(s,a))_{(s,a)\in\mathcal{S}\times\mathcal{A}}$ as

$$[\mathcal{T}(P,r)\nu](s,a) \triangleq [P\nu](s,a)(\cdot - r(s,a)),\ \forall (s,a)\in\mathcal{S}\times\mathcal{A}.$$

Denote by $\mathcal{T}_h \triangleq \mathcal{T}(P_h,r_h)$, then we can rewrite the Bellman recursion in Equation 2 in a compact form:

$$\eta_h^*(s,a) = [\mathcal{T}_h\nu_{h+1}^*](s,a). \tag{3}$$

3. Computational Inefficiency of RODI

Based on the distributional dynamic programming framework, Liang & Luo (2022) introduces the algorithm **R**isk-sensitive **O**ptimistic **D**istribution **I**teration (RODI), as detailed in Algorithm 1. In each episode, Algorithm 1 comprises two distinct phases: the planning phase and the interaction phase. During the planning phase, the algorithm executes an optimistic variant of the approximate Risk-Sensitive Distributional Dynamic Programming (RS-DDP), progressing backward from step $H+1$ to step 1 within each episode. This process results in a policy to be employed during the subsequent interaction phase.

RODI deviates from the DDP in two crucial updates:

$$\hat{\eta}_h \leftarrow \hat{\mathcal{T}}_h\nu_{h+1}$$
$$\tilde{\eta}_h \leftarrow \mathrm{O}_c\hat{\eta}_h.$$

In RODI, the *approximate distributional Bellman operator* $\hat{\mathcal{T}}$ is applied first, which relies on the empirical transition $\hat{P}$ rather than the true transition P. Then, the *distributional optimism operator* O_c is used to generate an optimistic return distribution.

Algorithm 1 RODI (Liang & Luo, 2022)

Input: T and δ
Initialize $N_h(\cdot,\cdot) \leftarrow 0$; $\eta_h(\cdot,\cdot), \nu_h(\cdot) \leftarrow \delta_{H+1-h}$
for $k = 1 : K$ **do**
  for $h = H : 1$ **do**
    $\eta_h(\cdot,\cdot) \leftarrow [\mathcal{T}(\hat{P}_h, r_h)\nu_{h+1}](\cdot,\cdot)$
    $c_h(\cdot,\cdot) \leftarrow \sqrt{\frac{2S}{N_h(\cdot,\cdot)\vee 1}\iota}$
    $\eta_h(\cdot,\cdot) \leftarrow O_{c_h(\cdot,\cdot)}\eta_h(\cdot,\cdot)$
    $\pi_h(\cdot) \leftarrow \arg\max_a U_\beta(\eta_h(\cdot,a))$
    $\nu_h(\cdot) \leftarrow \eta_h(\cdot, \pi_h(\cdot))$
  end for
  Receive s_1^k
  for $h = 1 : H$ **do**
    $a_h^k \leftarrow \pi_h(s_h^k)$ and transit to s_{h+1}^k
    Update $\hat{P}$
  end for
end for

3.1. Computational inefficiency

While RODI enjoys near-optimal regret guarantee, it suffers from computational inefficiency, especially in contexts with a large number of states or a long horizon. For better illustration, let's consider a *Markov Reward Process* with S states at each step. In particular, assume the transition kernel is uniform ($P_h(s'|s) = 1/S$) for any $(h, s') \in [H-1] \times \mathcal{S}$, and the reward function is bounded ($r_h(s) \in [0,1]$). Starting from the final step H, the return distribution $\eta_H(s) = \delta_{r_H(s)}$ is a Dirac function centered at $r_H(s)$. Applying the distributional Bellman equation at step $H-1$, we get

$$\eta_{H-1}(s) = \sum_{s'} p_{H-1}(s'|s)\delta_{r_H(s')+r_{H-1}(s)}.$$

$|\eta|$ represents the number of atoms (distinct elements) in a discrete distribution η, indicating the memory required to store this distribution. Since $|\eta_H(s)| = |\delta_{r_H(s)}| = 1$ for each $s \in \mathcal{S}$, and $\eta_{H-1}(s)$ is a uniform mixture of all $\eta_H(s)$ shifted by $r_{H-1}(s)$, we find

$$|\eta_{H-1}(s)| = \left| \left(r_{H-1}(s) + r_H(s'), \frac{1}{S} \right)_{s' \in \mathcal{S}} \right| = \mathcal{O}(S).$$

Continuing this process backwards through the time steps:

$$|\eta_{H-2}(s)| = \mathcal{O}(S^2)$$
$$\dots$$
$$|\eta_1(s)| = \mathcal{O}(S^{H-1}).$$

This analysis shows that the number of atoms in the return distribution *exponentially* increases with the horizon H, scaled by the number of states S at each application of the distributional Bellman operator. As a result, the memory and computational requirements to implement an *exact* distributional RL algorithm like RODI become prohibitive, particularly for problems with many states or a long horizon. This exponential growth in complexity highlights the computational challenges associated with RODI and underscores the need for *approximations* for practical implementations.

4. Distribution Representation and Projection

To address the computational challenges, we introduce two variants of RODI that use *distribution representation*. A widely used method of distribution representation is the *categorical representation* (Bellemare et al., 2023). This approach parameterizes the probability distribution at fixed locations. Specifically, we consider the simplest form of categorical representation that uses only two atoms. We refer to this as the *Bernoulli representation*. It represents the set of all discrete distributions with two distinct atoms, denoted as $\theta = (\theta_1, \theta_2)$. It is formally defined as:

$$\mathscr{F}_{\mathrm{B}}(\theta) = \{(1-p)\delta_{\theta_1} + p\delta_{\theta_2} : p \in [0,1]\}.$$

We introduce the Bernoulli representation for $\mathcal{T}_h$. Let

$$\begin{aligned}\bar{\nu}_{h+1}(s) &= (L_{h+1}(s), R_{h+1}(s); q_{h+1}(s)) \\ &\in \mathscr{F}_{\mathrm{B}}(L_{h+1}(s), R_{h+1}(s))\end{aligned}$$

be a Bernoulli representation of the true return distribution $\nu_{h+1}(s)$, where $L_{h+1}(s)$ and $R_{h+1}(s)$ are the left and right atoms, and $q_{h+1}(s)$ is the probability at $R_{h+1}(s)$. Applying $\mathcal{T}_h$ to $\bar{\nu}_{h+1}$, we obtain

$$\begin{aligned}[\mathcal{T}_h\bar{\nu}_{h+1}](s,a) = (&r_h(s,a) + L_{h+1}(s'), r_h(s,a) + R_{h+1}(s'); \\ &p_h(s'|s,a)q_{h+1}(s'))_{s'\in\mathcal{S}} \notin \mathscr{F}_{\mathrm{B}}.\end{aligned}$$

The result is no longer a Bernoulli distribution but a categorical distribution with (at most) $2S$ atoms. This demonstrates that the Bernoulli representation is not *closed* under $\mathcal{T}_h$

$$\nu \in \mathscr{F}_{\mathrm{B}} \not\Longrightarrow \mathcal{T}_h\nu \in \mathscr{F}_{\mathrm{B}}.$$

To overcome this issue, we introduce the *Bernoulli projection operator*. This operator serves as a mapping from the space of all probability distributions to $\mathscr{F}_{\mathrm{B}}$, and we denote it as $\Pi : \mathcal{D} \mapsto \mathscr{F}_{\mathrm{B}}$. Algorithmically, we add a projection step immediately after the application of $\mathcal{T}$, resulting in a *projected distributional Bellman operator* $\Pi\mathcal{T}$. This projection ensures that each iteration of $\eta_h = \Pi\mathcal{T}_h\nu_{h+1}$ is representable using a limited amount of memory.

The projection operator is not unique. Previous work (Bellemare et al., 2023) have developed projection operators aiming to find the best approximation to a given probability distribution, as measured by a specific probability metric. We introduce a novel type of Bernoulli projection that *preserves the ERM value*, an essential aspect in risk-sensitive

settings. Starting from a Dirac measure δ_c, we define the *value-equivalent Bernoulli projection operator* as:

$$\Pi\delta_c \triangleq (1-q(c;\theta))\delta_{\theta_1} + q(c;\theta)\delta_{\theta_2} = (\theta_1, \theta_2; q(c;\theta)),$$

where the probability is defined as

$$q(c;\theta) = \frac{e^{\beta c} - e^{\beta\theta_1}}{e^{\beta\theta_2} - e^{\beta\theta_1}} \in [0,1]. \tag{4}$$

It is easy to verify that $U_\beta(\Pi\delta_c) = U_\beta(\delta_c) = c, \forall c \in [\theta_1, \theta_2]$. Now we extend the definition to a categorical distributions $(c_i, p_i)_{i\in[n]}$ as:

$$\begin{aligned}\Pi\left(c_i, p_i\right)_{i\in[n]} &= \Pi\left(\sum_{i\in[n]} p_i\delta_{c_i}\right) \triangleq \sum_i p_i \Pi\delta_{c_i} \\ &= \left(\theta_1, \theta_2; \sum_i p_i q(c_i;\theta)\right).\end{aligned}$$

Given that $\mathrm{EU}\left(\delta_c\right) = \mathrm{EU}\left(\Pi\delta_c\right)$, the linearity of EU implies

$$\mathrm{EU}\left(\sum_i p_i\delta_{c_i}\right) = \mathrm{EU}\left(\Pi\sum_i p_i\delta_{c_i}\right).$$

This verifies the value equivalence of Π. To ensure the preservation of the value, the only requirement is that the interval $[\theta_1, \theta_2]$ covers the support of the input distribution, i.e., $\theta_1 \le \min c_i \le \max c_i \le \theta_2$.

The projection preserves the risk value of the original distribution, enabling efficient and accurate representation in DRL for RSRL. Drawing from these observations, we propose two DRL algorithms with Bernoulli representation, differing in the order of projection and optimism operator. We term the two algorithms as `RODI-Rep`.

5. DRL with Bernoulli Representation

Given that $\eta_h \in \mathcal{D}_{H+1-h}$, we set the *uniform* location parameters, which are independent of (s,a), as

$$L_h \triangleq 0,\ R_h \triangleq H+1-h.$$

We represent each iterate by a Bernoulli distribution

$$\begin{aligned}\eta_h^k(s,a) &= (1-q_h^k(s,a))\delta_{L_h} + q_h^k(s,a)\delta_{R_h}, \\ \nu_h^k(s) &= (1-q_h^k(s))\delta_{L_h} + q_h^k(s)\delta_{R_h},\end{aligned}$$

where we overload the notation for $q_h^k(s,a)$ and $q_h^k(s)$. Applying $\hat{\mathcal{T}}_h$ to the Bernoulli represented $\nu_{h+1}^k \in \mathscr{F}_{\mathrm{B}}$ yields

$$\begin{aligned}&\eta_h^k(s,a) = [\hat{\mathcal{T}}_h\nu_{h+1}^k](s,a) \\ &= \left(r_h(s,a) + L_{h+1}, r_h(s,a) + R_{h+1}; [\hat{P}_h^k q_{h+1}^k](s,a)\right).\end{aligned}$$

With slight abuse of notation, we let

$$L_h(s,a) \triangleq r_h(s,a) + L_{h+1},\ R_h(s,a) \triangleq r_h(s,a) + R_{h+1}.$$

$\eta_h^k(s,a)$ is a Bernoulli distribution with support not coinciding with L_h and R_h. We propose two different algorithms differing in the order of projection and optimism operator.

5.1. Optimism-Then-Projection

`RODI-OTP` applies the optimism operator first, followed by the projection operator:

$$\eta_h^k \leftarrow \Pi \mathrm{O}_c \hat{\mathcal{T}}_h \nu_{h+1}^k.$$

Note that $\eta_h^k \leftarrow \hat{\mathcal{T}}_h\nu_{h+1}^k \in \mathscr{F}_{\mathrm{B}}(r_h(s,a) + L_{h+1}, r_h(s,a) + R_{h+1})$. For Bernoulli distribution, the optimism operator admits a simple form

$$\mathrm{O}_c\left(a, b; p\right) = \left(a, b; \min(p+c, 1)\right).$$

Applying optimism operator to η_h^k yields

$$\begin{aligned}&\mathrm{O}_{c_h^k(s,a)}\left(\eta_h^k(s,a)\right) \\ &= \left(L_h(s,a), R_h(s,a); \min\left([\hat{P}_h^k q_{h+1}^k](s,a) + c_h^k(s,a), 1\right)\right).\end{aligned}$$

We can simplify the update in a parametric form

$$\begin{aligned}q_h^k(s,a) &\leftarrow [\hat{P}_h^k q_{h+1}^k](s,a), \\ q_h^k(s,a) &\leftarrow \min(q_h^k(s,a) + c_h^k(s,a), 1).\end{aligned}$$

We apply the projection rule (cf. Equation 4) to obtain

$$q_h^k(s,a) \leftarrow (1-q_h^k(s,a))q_h^L(s,a) + q_h^k(s,a)q_h^R(s,a),$$

where

$$\begin{aligned}q_h^R(s,a) &\triangleq q(L_h(s,a); L_h, R_h) = \frac{e^{\beta(r_h(s,a)+H-h)} - 1}{e^{\beta(H+1-h)} - 1}, \\ q_h^L(s,a) &\triangleq q(R_h(s,a); L_h, R_h) = \frac{e^{\beta r_h(s,a)} - 1}{e^{\beta(H+1-h)} - 1}.\end{aligned}$$

Remark 5.1. $q_h^R(s,a)$ and $q_h^L(s,a)$ are *fixed* (independent of k) and *known*. Thus we can compute their values for all (h,s,a) in advance.

5.2. Projection-Then-Optimism

`RODI-PTO` applies the projection operator first, followed by the optimism operator:

$$\eta_h^k \leftarrow \mathrm{O}_c \Pi \hat{\mathcal{T}}_h \nu_{h+1}^k.$$

The update can also be represented in a parametric form:

$$\begin{aligned}q_h^k(s,a) &\leftarrow [\hat{P}_h^k q_{h+1}^k](s,a), \\ q_h^k(s,a) &\leftarrow (1-q_h^k(s,a))q_h^L(s,a) + q_h^k(s,a)q_h^R(s,a), \\ q_h^k(s,a) &\leftarrow \min(q_h^k(s,a) + c_h^k(s,a), 1).\end{aligned}$$

After applying optimism operator and projection operator, both RODI-OTP and RODI-PTO update the value functions and policies accordingly

$$Q_h^k(s,a) \leftarrow \frac{1}{\beta}\log\left(1 - q_h^k(s,a) + q_h^k(s,a)e^{\beta(H+1-h)}\right)$$
$$\pi_h^k(s) \leftarrow \arg\max_a Q_h^k(s,a), V_h^k(s) \leftarrow Q_h^k(s,\pi_h^k(s))$$
$$q_h^k(s) \leftarrow q_h^k(s,\pi_h^k(s)).$$

Computational complexity. The *time complexity* of RODI-OTP and RODI-PTO is given as follows: i) computation of q^L and q^R: $\mathcal{O}(HSA)$; ii) parametric Bellman update: $KHSA \cdot \mathcal{O}(S)$; iii) projection: $KHSA \cdot \mathcal{O}(1)$; iv) optimism operator: $KHSA \cdot \mathcal{O}(1)$; v) computation of Q-function: $KHSA \cdot \mathcal{O}(1)$; vi) greedy policy: $KHS \cdot \mathcal{O}(A \log A)$. Therefore, the total time complexity is

$$\mathcal{O}(KHSA(S + \log A),$$

which is the same as that of RSVI2. The *space complexity* of both algorithm is given as follows: i) q^L and q^R: $\mathcal{O}(HSA)$; ii) $N_h(s,a)$: $\mathcal{O}(HSA)$; iii) trajectory $(s_h^k, a_h^k)_{k,h}$: $\mathcal{O}(T)$; iv) probabilities $q_h(s,a)$: $\mathcal{O}(HSA)$; v) action-value function: $\mathcal{O}(HSA)$. Therefore, their total space complexity is $\mathcal{O}(HSA + T)$.

While RODI-OTP and RODI-PTO adapts RODI by Bernoulli representation, they maintain the optimism mainly due to the value-equivalence property of the projection operator. Therefore, they enjoys near-optimal regret bound as RODI while maintaining computational efficiency.

5.3. Optimism of RODI-OTP

For analysis, we write the update of RODI-OTP in distributional form as:

$$\begin{aligned}
\hat{\eta}_h(s,a) &= [\hat{\mathcal{T}}_h \nu_{h+1}](s,a) \\
\tilde{\eta}_h(s,a) &= \mathrm{O}_{c_h(s,a)}\hat{\eta}_h(s,a) \\
\eta_h(s,a) &= \Pi\tilde{\eta}_h(s,a) \\
Q_h(s,a) &= U_\beta\left(\eta_h(s,a)\right), \\
\pi_h(s) &= \arg\max_a Q_h(s,a) \\
\nu_h(s) &= \eta_h(s,\pi_h(s)).
\end{aligned} \tag{5}$$

Proposition 5.2 (Optimism of RODI-OTP)**.** *Let Q_h^k and V_h^k be the value functions generated by RODI-OTP as Equation 5. It holds that $Q_h^k(s,a) \geq Q_h^*(s,a)$ and $V_h^k(s) \geq V_h^*(s)$ for any (k,h,s,a) with high probability.*

The proof is deferred to Section A.

5.4. Optimism of RODI-PTO

We rewrite the update of $q_h(s,a)$ in RODI-PTO as:

$$\begin{aligned}
&\hat{q}_h(s,a) \leftarrow [\hat{P}_h q_{h+1}](s,a), \\
&\hat{\eta}_h(s,a) = (L_h(s,a), R_h(s,a); \hat{q}_h(s,a)) \\
&\bar{q}_h(s,a) \leftarrow (1 - \hat{q}_h(s,a))q_h^L(s,a) + \hat{q}_h(s,a)q_h^R(s,a), \\
&\bar{\eta}_h(s,a) = (L_h, R_h; \bar{q}_h(s,a)) \\
&q_h(s,a) \leftarrow \min(\bar{q}_h(s,a) + c_h(s,a), 1), \\
&\eta_h(s,a) = (L_h, R_h; q_h(s,a))\,.
\end{aligned} \tag{6}$$

Proposition 5.3 (Optimism of RODI-PTO)**.** *Let Q_h^k and V_h^k be the value functions generated by RODI-PTO as Equation 6. It holds that $Q_h^k(s,a) \geq Q_h^*(s,a)$ and $V_h^k(s) \geq V_h^*(s)$ for any (k,h,s,a) with high probability.*

The proof is deferred to Section A.

6. Theoretical Comparisons

6.1. RODI vs. RSVI2

We first provide theoretical justifications regarding the regret ranking of RSVI (Fei et al., 2020), RSVI2 (Fei et al., 2021), and RODI (Liang & Luo, 2022), which demonstrates the advantage of distributional optimism over bonus-based optimism used in RSVI and RSVI2. A key observation regarding the ranking of their value functions V^k is that:

$$\text{value functions} : \texttt{RSVI} > \texttt{RSVI2} > \texttt{RODI} \geq V^*.$$

This ordering will be formally presented in Equation 7. The last part of this inequality sequence indicates that all these value functions are indeed optimistic. Given that the level of optimism is mirrored in the value functions, we can deduce:

$$\text{optimism level} : \texttt{RSVI} > \texttt{RSVI2} > \texttt{RODI}.$$

Considering the relationship between regret and the optimistic value function V^k

$$\text{Regret} = \sum_{k\in[K]} V_1^* - V_1^{\pi^k} \leq \sum_{k\in[K]} V_1^k - V_1^{\pi^k},$$

it is intuitive that a smaller V^k or less optimism induces reduced regret. Consequently, their regret can be ranked as:

$$\text{regret} : \texttt{RSVI} > \texttt{RSVI2} > \texttt{RODI},$$

which explains Figure 1. The regret bounds of RODI should at least match those of RSVI2, explaining the ranking of their regret bounds reported in Table 1:

$$\text{regret bound} : \texttt{RSVI} > \texttt{RSVI2} = \texttt{RODI}.$$

Despite sharing same regret bounds with RSVI2, RODI outperforms RSVI2 both theoretically and empirically. Formally speaking, let V', V'', V denote the value functions

generated by RSVI, RSVI2, and RODI respectively. Let $\tilde{\eta}$ denote the distribution generated by RODI. We omit k for simplicity.

Proposition 6.1. *Fix (s, a, k, h). The comparison of their values is as follows:*

$$\begin{aligned} \textit{RSVI}\ & \frac{1}{\beta}\log\left(\left[\hat{P}_h e^{\beta V'_{h+1}}\right] + b'_h\right) \\ & \overset{(a)}{>} \frac{1}{\beta}\log\left(\left[\hat{P}_h e^{\beta V''_{h+1}}\right] + b'_h\right) \\ & \overset{(b)}{>} \frac{1}{\beta}\log\left(\left[\hat{P}_h e^{\beta V''_{h+1}}\right] + b''_h\right) \quad \textit{RSVI2} \\ & \overset{(c)}{>} U_\beta\left(\tilde{\eta}_h\right) \quad \textit{RODI} \\ & \overset{(d)}{>} \frac{1}{\beta}\log\left(\left[P_h e^{\beta V^*_{h+1}}\right]\right). \end{aligned} \tag{7}$$

Both RSVI and RSVI2 use exploration bonuses, defined as $b'_h = |e^{\beta H} - 1|c_h$ and $b''_h = |e^{\beta(H+1-h)} - 1|c_h$ respectively, where $c_h(s, a)$ represents the model estimation error

$$\left\|\hat{P}_h(s,a) - P_h(s,a)\right\|_1 \le c_h(s,a) = \sqrt{\frac{S\iota}{N_h(s,a)}}.$$

Both b''_h and b'_h are formulated as a multiplier times c_h. Notably, b''_h, referred to as the the *doubly decaying bonus* (Fei et al., 2021), decreases its multiplier exponentially across stages h, contrasting with b'_h in RSVI. In comparison, RODI directly incorporates optimism into the return distribution using an optimism constant c_h. A connection between c_h and the bonus via the Lipschitz constant of EU can be estblished

$$b''_h = L(E_\beta, H-h)c_h < L(E_\beta, H)c_h = b'_h,$$

where $L(E_\beta, M)$ denotes the Lipschitz constant of EU over the distributions supported in $[0, M]$. This distributional perspective posits that RSVI and RSVI2 design bonuses to offset the error in value estimates, which is bounded by the product of the Lipschitz constant of EU and the error in the return distribution:

$$\begin{aligned} V^k_h - V_h &\le L(E_\beta, H-h)\left\|\eta^k_h - \eta_h\right\| \\ &\le L(E_\beta, H-h)\left\|P^k_h - P_h\right\| \\ &\le L(E_\beta, H-h)c^k_h. \end{aligned}$$

Under the distributional perspective, the multiplier in the bonus b''_h is interpreted as the Lipschitz constant that links the return estimation error c_h to the value estimation error b''_h. The Lipschitz constant decreases exponentially in h as the range $[0, H-h]$ of the return distribution narrows.

In conclusion, bonus-based optimism requires an exponentially decaying multiplier or Lipschitz constant, whereas distributional optimism functions directly at the distributional level, obviating the need for a multiplier. Next, we theoretically justify the regret ranking of RODI-OTP and RODI-PTO, which interpolates between RODI and RSVI2.

6.2. RODI-Rep vs. RSVI2

We delve into the analysis by first explaining why RODI-PTO achieves marginally lower regret compared to RSVI2, and subsequently, we justify the advantage of RODI-OTP over RODI-PTO.

Near-equivalence between RSVI2 and RODI-PTO We can show the near-equivalence between RSVI2 and RODI-PTO using induction.

Proposition 6.2 (Near-equivalence)**.** *Let V and V' denote the value functions generated by RODI-PTO and RSVI2 respectively. Then we have $V_h \le V'_h$. Moreover, $V_h = V'_h$ for every $h \in [H]$ if $\bar{q}_h(s,a) + c_h(s,a) \le 1$ for every (h, s, a).*

The proof is deferred to Section A. The condition is likely to be met for large values of k, considering that

$$k \uparrow \Longrightarrow N^k_h \downarrow \Longrightarrow c^k_h \propto 1/\sqrt{N^k_h} \downarrow.$$

Benefits of RODI-OTP The recursion of $q_h(s, a)$ in RODI-OTP writes

$$\begin{aligned} &\hat{q}_h(s,a) \leftarrow [\hat{P}_h q_{h+1}](s,a) \\ &\tilde{q}_h(s,a) \leftarrow \min(\hat{q}_h(s,a) + c_h(s,a), 1) \\ &q_h(s,a) \leftarrow (1 - \tilde{q}_h(s,a))q^L_h(s,a) + \tilde{q}_h(s,a)q^R_h(s,a). \end{aligned}$$

Proposition 6.3. *Let V and V' denote the value functions generated by RODI-OTP and RSVI2 respectively. We have*

$$\begin{aligned} Q_h &\le \frac{1}{\beta}\log\left(e^{\beta r_h}[\hat{P}_h e^{\beta V_{h+1}}] + c_h(s,a)e^{\beta r_h}(e^{\beta(H-h)} - 1)\right) \\ &< \frac{1}{\beta}\log\left(e^{\beta r_h}[\hat{P}_h e^{\beta V'_{h+1}}] + c_h e^{\beta r_h}(e^{\beta(H-h)} - 1)\right) \\ &< \frac{1}{\beta}\log\left(e^{\beta r_h}[\hat{P}_h e^{\beta V'_{h+1}}] + c_h(e^{\beta(H+1-h)} - 1)\right) \\ &= Q'_h. \end{aligned}$$

The proof is deferred to Section A.

Remark 6.4. This explains why RODI-OTP achieves an order of magnitude improvement in regret compared with RSVI2 as well as RODI-PTO, as the "optimism level ratio" of RODI-OTP to RSVI2 at step h is quantifiable by

$$\frac{e^{\beta(r_h(s,a)+H-h)} - e^{\beta r_h(s,a)}}{e^{\beta(H+1-h)} - 1} < 1.$$

Why OTP is better than PTO. The superiority of OTP over PTO can be substantiated through an insightful observation about the optimization problem:

$$\begin{aligned} \min_q \quad & U_\beta\left(L, R; q\right) \\ \text{s.t.} \quad & U_\beta\left(L, R; q\right) \ge U_\beta\left(\eta\right) \\ & \left\|\eta - \hat{\eta}\right\|_\infty \le c \\ & \eta = D(\text{Supp}(\hat{\eta})) \end{aligned} \tag{8}$$

Let $(L, R; \tilde{q})$ be the optimal solution to this problem. It turns out that the optimal solution is given by $(L, R; \tilde{q}) = \Pi \mathrm{O}_c \hat{\eta}$, aligning with the OTP principle. Fixing (h, s, a), we interpret $\hat{\eta} \triangleq [\hat{\mathcal{T}}_h \nu_{h+1}](s, a)$ as the empirical Bellman operator applied to ν_{h+1}. Suppose ν_{h+1} is optimistic relative to the true distribution ν^*_{h+1}, i.e., $U_\beta(\nu_{h+1}) \geq \nu^*_{h+1}$. Define $\check{\eta} \triangleq [\mathcal{T}_h \nu_{h+1}](s, a)$, which is the exact Bellman operator applied to ν_{h+1}. Given that

$$\|\hat{\eta} - \check{\eta}\|_\infty = \left\| [(\hat{\mathcal{T}}_h - \mathcal{T}_h)\nu_{h+1}](s, a) \right\|_\infty \leq c_h(s, a),$$

the optimal solution satisfies

$$\begin{aligned} U_\beta(L, R; \tilde{q}) &\geq U_\beta(\check{\eta}) = U_\beta\left([\hat{\mathcal{T}}_h \nu_{h+1}](s, a)\right) \\ &\geq U_\beta\left([\hat{\mathcal{T}}_h \nu^*_{h+1}](s, a)\right) \\ &= U_\beta(\eta^*_h(s, a)) = Q^*_h(s, a). \end{aligned}$$

Hence, the optimal solution $(L, R; \tilde{q})$ is optimistic over $\eta^*_h(s, a)$. The nature of the optimization problem compels $(L, R; \tilde{q})$ to be the Bernoulli distribution with support (L_h, R_h) that necessitates minimal optimism over $\eta^*_h(s, a)$. Notably, the PTO solution $\mathrm{O}_c \Pi \hat{\eta}$ is also a feasible solution. Consequently, OTP induces less optimism than PTO:

$$U_\beta(\Pi \mathrm{O}_c \hat{\eta}) < U_\beta(\mathrm{O}_c \Pi \hat{\eta}).$$

This analysis elucidates the inherent advantage of the OTP approach over PTO. By inverting the order of the projection and optimism operators, OTP not only ensures an optimism over the true distribution but also guarantees that the induced optimism is minimal and necessary.

7. Numerical Experiments

To validate the empirical performance of our algorithms, we conducted numerical experiments comparing the proposed RODI-Rep, with the risk-neutral algorithm UCBVI (Azar et al., 2017), RSVI in (Fei et al., 2020), RSVI2 in (Fei et al., 2021), and RODI in (Liang & Luo, 2022).

The experimental setup involved an MDP with $S = 5$ states, $A = 5$ actions, and a horizon $H = 5$, mirroring the setup in (Du et al., 2022). The MDP consists of a fixed initial state denoted as state 0, and S additional states. The agent started in state 0 and could take actions from the set $[A]$, transitioning to one of the states in $[S]$ in the next step.

This MDP was designed to be highly risky, with the risk-neutral optimal policy leading to a mean reward of 0.5 but with a chance of receiving no reward. A risk-aware policy might prefer the last action A, which offers slightly less mean reward but a more consistent return, indicating lower risk. We set $\delta = 0.005$ and $\beta = -1.1$. The results, as illustrated in Figure 1, demonstrates the regret ranking of these algorithms:

$$\texttt{RODI} < \underbrace{\texttt{RODI-OTP} < \texttt{RODI-PTO}}_{\texttt{RODI-Rep}} \lesssim \texttt{RSVI2} < \texttt{RSVI}.$$

Figure 1 includes the following key observations:
(i) Advantage of distributional over non-Distributional algorithms: DRL algorithms (RODI and RODI-Rep) outperforms non-distributional algorithms, demonstrating the effectiveness of distributional optimism over bonus-based optimism.
(ii) Performance of RODI vs. RODI-Rep: While RODI shows better performance than RODI-Rep, the latter offers a balance between statistical and computational efficiency.
(iii) Comparison of RODI-Rep with RSVI2: RODI-Rep demonstrates advantages over RSVI2 in terms of sample efficiency, while also maintaining computational efficiency.

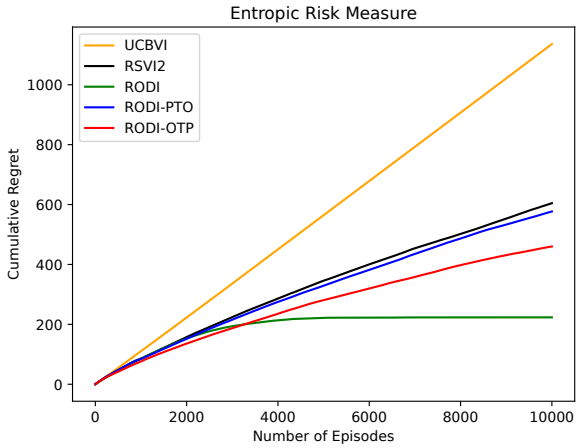

Figure 1. Regret comparisons for different algorithms.

8. Conclusion

We introduces significant advancements in the integration of DRL and RSRL through the development of the RODI algorithm. Our innovations address critical challenges in computational efficiency and provide robust regret guarantees. The proposed RODI-Rep variant, in particular, demonstrates improved regret performance compared to traditional non-distributional methods while maintaining high computational efficiency. Promising future directions include extending the DRL algorithm with distribution representation to accommodate large state-action spaces.

References

Achab, M. and Neu, G. Robustness and risk management via distributional dynamic programming. *arXiv preprint arXiv:2112.15430*, 2021.

Azar, M. G., Osband, I., and Munos, R. Minimax regret bounds for reinforcement learning. In *International Conference on Machine Learning*, pp. 263–272. PMLR, 2017.

Barth-Maron, G., Hoffman, M. W., Budden, D., Dabney, W., Horgan, D., Tb, D., Muldal, A., Heess, N., and Lillicrap, T. Distributed distributional deterministic policy gradients. *arXiv preprint arXiv:1804.08617*, 2018.

Bäuerle, N. and Rieder, U. More risk-sensitive markov decision processes. *Mathematics of Operations Research*, 39(1):105–120, 2014.

Bellemare, M. G., Dabney, W., and Munos, R. A distributional perspective on reinforcement learning. In *International Conference on Machine Learning*, pp. 449–458. PMLR, 2017.

Bellemare, M. G., Dabney, W., and Rowland, M. *Distributional Reinforcement Learning*. MIT Press, 2023. http://www.distributional-rl.org.

Bielecki, T. R., Pliska, S. R., and Sherris, M. Risk sensitive asset allocation. *Journal of Economic Dynamics and Control*, 24(8):1145–1177, 2000.

Borkar, V. S. A sensitivity formula for risk-sensitive cost and the actor–critic algorithm. *Systems & Control Letters*, 44(5):339–346, 2001.

Borkar, V. S. Q-learning for risk-sensitive control. *Mathematics of operations research*, 27(2):294–311, 2002.

Borkar, V. S. Learning algorithms for risk-sensitive control. In *Proceedings of the 19th International Symposium on Mathematical Theory of Networks and Systems–MTNS*, volume 5, 2010.

Borkar, V. S. and Meyn, S. P. Risk-sensitive optimal control for markov decision processes with monotone cost. *Mathematics of Operations Research*, 27(1):192–209, 2002.

Cavazos-Cadena, R. and Hernández-Hernández, D. Discounted approximations for risk-sensitive average criteria in markov decision chains with finite state space. *Mathematics of Operations Research*, 36(1):133–146, 2011.

Dabney, W., Ostrovski, G., Silver, D., and Munos, R. Implicit quantile networks for distributional reinforcement learning. In *International conference on machine learning*, pp. 1096–1105. PMLR, 2018a.

Dabney, W., Rowland, M., Bellemare, M. G., and Munos, R. Distributional reinforcement learning with quantile regression. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018b.

Davis, M. and Lleo, S. Risk-sensitive benchmarked asset management. *Quantitative Finance*, 8(4):415–426, 2008.

Delage, E. and Mannor, S. Percentile optimization for markov decision processes with parameter uncertainty. *Operations research*, 58(1):203–213, 2010.

Di Masi, G. B. and Stettner, Ł. Infinite horizon risk sensitive control of discrete time markov processes under minorization property. *SIAM Journal on Control and Optimization*, 46(1):231–252, 2007.

Di Masi, G. B. et al. Infinite horizon risk sensitive control of discrete time markov processes with small risk. *Systems & control letters*, 40(1):15–20, 2000.

Du, Y., Wang, S., and Huang, L. Provably efficient risk-sensitive reinforcement learning: Iterated cvar and worst path. In *The Eleventh International Conference on Learning Representations*, 2022.

Ernst, D., Stan, G.-B., Goncalves, J., and Wehenkel, L. Clinical data based optimal sti strategies for hiv: a reinforcement learning approach. In *Proceedings of the 45th IEEE Conference on Decision and Control*, pp. 667–672. IEEE, 2006.

Fei, Y., Yang, Z., Chen, Y., Wang, Z., and Xie, Q. Risk-sensitive reinforcement learning: Near-optimal risk-sample tradeoff in regret. *arXiv preprint arXiv:2006.13827*, 2020.

Fei, Y., Yang, Z., Chen, Y., and Wang, Z. Exponential bellman equation and improved regret bounds for risk-sensitive reinforcement learning. *Advances in Neural Information Processing Systems*, 34, 2021.

Föllmer, H. and Schied, A. Stochastic finance. In *Stochastic Finance*. de Gruyter, 2016.

Hansen, L. P. and Sargent, T. J. Robustness. In *Robustness*. Princeton university press, 2011.

Howard, R. A. and Matheson, J. E. Risk-sensitive markov decision processes. *Management science*, 18(7):356–369, 1972.

Jaśkiewicz, A. Average optimality for risk-sensitive control with general state space. *The annals of applied probability*, 17(2):654–675, 2007.

Langley, P. Crafting papers on machine learning. In Langley, P. (ed.), *Proceedings of the 17th International Conference on Machine Learning (ICML 2000)*, pp. 1207–1216, Stanford, CA, 2000. Morgan Kaufmann.

Liang, H. and Luo, Z.-Q. Bridging distributional and risk-sensitive reinforcement learning with provable regret bounds. *arXiv preprint arXiv:2210.14051*, 2022.

Lyle, C., Bellemare, M. G., and Castro, P. S. A comparative analysis of expected and distributional reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 4504–4511, 2019.

Ma, X., Xia, L., Zhou, Z., Yang, J., and Zhao, Q. Dsac: Distributional soft actor critic for risk-sensitive reinforcement learning. *arXiv preprint arXiv:2004.14547*, 2020.

Ma, Y., Jayaraman, D., and Bastani, O. Conservative offline distributional reinforcement learning. *Advances in Neural Information Processing Systems*, 34, 2021.

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

Nass, D., Belousov, B., and Peters, J. Entropic risk measure in policy search. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1101–1106. IEEE, 2019.

Osogami, T. Robustness and risk-sensitivity in markov decision processes. *Advances in Neural Information Processing Systems*, 25:233–241, 2012.

Patek, S. D. On terminating markov decision processes with a risk-averse objective function. *Automatica*, 37(9): 1379–1386, 2001.

Rowland, M., Bellemare, M., Dabney, W., Munos, R., and Teh, Y. W. An analysis of categorical distributional reinforcement learning. In *International Conference on Artificial Intelligence and Statistics*, pp. 29–37. PMLR, 2018.

Shapiro, A., Dentcheva, D., and Ruszczynski, A. *Lectures on stochastic programming: modeling and theory*. SIAM, 2021.

Shen, Y., Stannat, W., and Obermayer, K. Risk-sensitive markov control processes. *SIAM Journal on Control and Optimization*, 51(5):3652–3672, 2013.

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

Singh, R., Zhang, Q., and Chen, Y. Improving robustness via risk averse distributional reinforcement learning. In *Learning for Dynamics and Control*, pp. 958–968. PMLR, 2020.

Sutton, R. S. and Barto, A. G. *Reinforcement learning: An introduction*. MIT press, 2018.

Yang, D., Zhao, L., Lin, Z., Qin, T., Bian, J., and Liu, T.-Y. Fully parameterized quantile function for distributional reinforcement learning. *Advances in neural information processing systems*, 32, 2019.

Zhang, P., Chen, X., Zhao, L., Xiong, W., Qin, T., and Liu, T.-Y. Distributional reinforcement learning for multi-dimensional reward functions. *Advances in Neural Information Processing Systems*, 34:1519–1529, 2021.

A. Missing Proofs

A.1. Proof of Proposition 5.2

Proof. Define

$$\check{\eta}_h(s,a) \triangleq [\mathcal{T}_h \nu_{h+1}](s,a) = [P_h \nu_{h+1}]\,[s,a](\cdot - r_h(s,a)),$$

which is the Bellman target that replaces $\hat{P}_h$ by the true model P_h. Note that $\nu_{h+1} \in \mathscr{F}_{\mathrm{B}}$ is the distribution generated by the algorithm, which is Bernoulli represented, rather than the optimal distribution ν^*_{h+1}. Since

$$\begin{aligned}
\|\check{\eta}_h(s,a) - \hat{\eta}_h(s,a)\|_\infty &= \left\| \left[\hat{P}_h \nu_{h+1}\right][s,a](\cdot - r_h(s,a)) - [P_h \nu_{h+1}]\,[s,a](\cdot - r_h(s,a)) \right\|_\infty \\
&= \left\| \left[\hat{P}_h \nu_{h+1}\right][s,a] - [P_h \nu_{h+1}]\,[s,a] \right\|_\infty \\
&\le \left\| \hat{P}_h(s,a) - P_h(s,a) \right\|_1 \le c_h(s,a),
\end{aligned}$$

we have

$$\tilde{\eta}_h(s,a) = \mathrm{O}_{c_h(s,a)} \hat{\eta}_h(s,a) \succeq \check{\eta}_h(s,a).$$

We can prove the argument by induction. Fix $h+1 \in [2:H+1]$. Suppose $V_{h+1} = U_\beta(\eta_{h+1}) \ge U_\beta\left(\eta^*_{h+1}\right) = V^*_{h+1}$ for any s. It follows that

$$\begin{aligned}
Q_h(s,a) &= U_\beta(\eta_h(s,a)) = U_\beta(\Pi\tilde{\eta}_h(s,a)) = U_\beta(\tilde{\eta}_h(s,a)) = U_\beta\left(\mathrm{O}_{c_h(s,a)}\hat{\eta}_h(s,a)\right) \\
&\ge U_\beta(\check{\eta}_h(s,a)) = U_\beta(\mathcal{T}_h \nu_{h+1}) \\
&\ge U_\beta\left(\mathcal{T}_h \nu^*_{h+1}\right) = Q^*_h(s,a),
\end{aligned}$$

which implies $V_h(s) \ge V^*_h(s)$ for any s. The induction is completed. □

A.2. Proof of Proposition 5.3

Proof. Define

$$\check{q}_h(s,a) \triangleq [P_h q_{h+1}]\,[s,a], \quad \check{\eta}_h(s,a) \triangleq \left(L_h(s,a), R_h(s,a); \check{q}_h(s,a)\right),$$

then we have

$$\Pi\check{\eta}_h(s,a) = \left(L_h, R_h; (1-\check{q}_h(s,a))q^L_h(s,a) + \check{q}_h(s,a)q^R_h(s,a)\right).$$

$\check{\eta}_h(s,a)$ and $\hat{\eta}_h(s,a)$ are both Bernoulli distributions with the same support, thus

$$\|\check{\eta}_h(s,a) - \hat{\eta}_h(s,a)\|_\infty = |\check{q}_h(s,a) - \hat{q}_h(s,a)| = \left|\left[(\hat{P}_h - P_h)q_{h+1}\right](s,a)\right| \le \left\|(\hat{P}_h - P_h)(s,a)\right\|_1.$$

We have

$$\begin{aligned}
\|\Pi\check{\eta}_h(s,a) - \Pi\hat{\eta}_h(s,a)\|_\infty &= \left|(1-\check{q}_h(s,a))q^L_h(s,a) + \check{q}_h(s,a)q^R_h(s,a) - (1-\hat{q}_h(s,a))q^L_h(s,a) - \hat{q}_h(s,a)q^R_h(s,a)\right| \\
&= \left|(\check{q}_h(s,a) - \hat{q}_h(s,a))(q^R_h(s,a) - q^L_h(s,a))\right| \\
&= \left|[(\hat{P}_h - P_h)q_{h+1}](s,a)(q^R_h(s,a) - q^L_h(s,a))\right| \\
&= (q^R_h(s,a) - q^L_h(s,a))\,\|\check{\eta}_h(s,a) - \hat{\eta}_h(s,a)\|_\infty \\
&\le (q^R_h(s,a) - q^L_h(s,a))\left\|\hat{P}_h(s,a) - P_h(s,a)\right\|_1 \\
&\le (q^R_h(s,a) - q^L_h(s,a))c_h(s,a) < c_h(s,a).
\end{aligned}$$

Suppose $V_{h+1} = U_\beta(\eta_{h+1}) \ge U_\beta\left(\eta^*_{h+1}\right) = V^*_{h+1}$ for any s. Since

$$\eta_h(s,a) = \mathrm{O}_{c_h(s,a)}\Pi\hat{\eta}_h(s,a) \succeq \Pi\check{\eta}_h(s,a),$$

we have

$$\begin{aligned}Q_h(s,a) &= U_\beta\left(\eta_h(s,a)\right) = U_\beta\left(\mathrm{O}_{c_h(s,a)}\bar{\eta}_h(s,a)\right) = U_\beta\left(\mathrm{O}_{c_h(s,a)}\Pi_h\hat{\eta}_h(s,a)\right)\\ &\geq U_\beta\left(\Pi_h\check{\eta}_h(s,a)\right)\\ &= U_\beta\left(\check{\eta}_h(s,a)\right) = U_\beta\left(\left[\mathcal{T}_h\nu_{h+1}\right](s,a)\right)\right) \geq U_\beta\left(\left[\mathcal{T}_h\nu^*_{h+1}\right](s,a)\right)\right) = Q^*_h(s,a).\end{aligned}$$

which implies $V_h(s) \geq V^*_h(s)$ for any s. The induction is completed. □

A.3. Proof of Proposition 6.2

Proof. Let V and V' denote the value functions generated by `RODI-PTO` and `RSVI2` respectively. We start with the base case that $h = H$. By the construction of `RODI-PTO`, we have

$$\begin{aligned}&Q_H(s,a) = r_H(s,a) = Q'_H(s,a) \Longrightarrow\\ &V_H(s) = \max_a Q_H(s,a) = \max_a Q'_H(s,a) = V'_H(s),\end{aligned}$$

verifying the equivalence at step H. Now fix $h \in [H-1]$. Suppose the following holds

$$\begin{aligned}V_{h+1}(s) &= \frac{1}{\beta}\log\left(1 - q_{h+1}(s) + q_{h+1}(s)e^{\beta(H-h)}\right)\\ &\leq V'_{h+1}(s), \forall s \in \mathcal{S} \Longrightarrow\\ 1 - q_{h+1}(s) + q_{h+1}(s)e^{\beta(H-h)} &\leq e^{\beta V'_{h+1}(s)}, \forall s \in \mathcal{S}.\end{aligned}$$

It follows that

$$\begin{aligned}Q_h(s,a) &= \frac{1}{\beta}\log\left((1-q_h(s,a))e^0 + q_h(s,a)e^{\beta(H+1-h)}\right)\\ &= \frac{1}{\beta}\log\left(1 + q_h(s,a)(e^{\beta(H+1-h)} - 1)\right)\\ &\leq \frac{1}{\beta}\log\left(1 + \bar{q}_h(s,a)(e^{\beta(H+1-h)} - 1) + c_h(s,a)(e^{\beta(H+1-h)} - 1)\right),\end{aligned}$$

where the last inequality becomes equality if $\bar{q}_h(s,a) + c_h(s,a) \leq 1$. By the definition of projection, we obtain

$$\begin{aligned}1 + \bar{q}_h(s,a)(e^{\beta(H+1-h)} - 1) &= 1 - \bar{q}_h(s,a) + \bar{q}_h(s,a)e^{\beta(H+1-h)}\\ &= (1 - \hat{q}_h(s,a))e^{\beta r_h(s,a)} + \hat{q}_h(s,a)e^{\beta(r_h(s,a)+H-h)}\\ &= [\hat{P}_h(1-q_{h+1})](s,a)e^{\beta r_h(s,a)} + [\hat{P}_h q_{h+1}](s,a)e^{\beta(r_h(s,a)+H-h)}\\ &= \sum_{s'}\hat{P}_h(s'|s,a)\left((1-q_{h+1}(s'))e^{\beta r_h(s,a)} + q_{h+1}(s')e^{\beta(r_h(s,a)+H-h)}\right)\\ &= e^{\beta r_h(s,a)}\sum_{s'}\hat{P}_h(s'|s,a)\left((1-q_{h+1}(s')) + q_{h+1}(s')e^{\beta(H-h)}\right)\\ &= e^{\beta r_h(s,a)}\sum_{s'}\hat{P}_h(s'|s,a)e^{\beta V_{h+1}(s')}\\ &\leq e^{\beta r_h(s,a)}\sum_{s'}\hat{P}_h(s'|s,a)e^{\beta V'_{h+1}(s')},\end{aligned}$$

which implies

$$Q_h(s,a) \leq \frac{1}{\beta}\log\left(e^{\beta r_h(s,a)}\sum_{s'}\hat{P}_h(s'|s,a)e^{\beta V'_{h+1}(s')} + c_h(s,a)(e^{\beta(H+1-h)} - 1)\right) = Q'_h(s,a).$$

Then we have $V_h(s) = \max_a Q_h(s,a) \leq \max_a Q'_h(s,a) = V'_h(s)$. The induction is completed. Moreover, it holds that $V_h = V'_h$ for every $h \in [H]$ if $\bar{q}_h(s,a) + c_h(s,a) \leq 1$ for every (h,s,a). This condition is likely to be met for large values of k, considering that

$$k \uparrow \Longrightarrow N^k_h \downarrow \Longrightarrow c^k_h \propto 1/\sqrt{N^k_h} \downarrow.$$

□

A.4. Proof of Proposition 6.3

Proof. The recursion of $q_h(s,a)$ in `RODI-OTP` writes

$$\begin{aligned}
&\hat{q}_h(s,a) \leftarrow [\hat{P}_h q_{h+1}](s,a)\\
&\tilde{q}_h(s,a) \leftarrow \min(\hat{q}_h(s,a) + c_h(s,a), 1)\\
&q_h(s,a) \leftarrow (1-\tilde{q}_h(s,a))q_h^L(s,a) + \tilde{q}_h(s,a)q_h^R(s,a).
\end{aligned}$$

Fix $(h,s,a) \in [H-1] \times \mathcal{S} \times \mathcal{A}$. Note that

$$\begin{aligned}
V_{h+1}(s) = \frac{1}{\beta}\log\left(1 - q_{h+1}(s) + q_{h+1}(s)e^{\beta(H-h)}\right), \forall s \in \mathcal{S},\\
\implies [\hat{P}_h e^{\beta V_{h+1}}](s,a) = (1-\hat{q}_h(s,a)) + \hat{q}_h(s,a)e^{\beta(H-h)}, \forall (s,a),
\end{aligned}$$

then we have

$$\begin{aligned}
Q_h(s,a) &= \frac{1}{\beta}\log\left(1 - q_h(s,a) + q_h(s,a)e^{\beta(H+1-h)}\right)\\
&= \frac{1}{\beta}\log\left((1-\tilde{q}_h(s,a))e^{\beta r_h(s,a)} + \tilde{q}_h(s,a)e^{\beta(r_h(s,a)+H-h)}\right)\\
&\le \frac{1}{\beta}\log\left((1-\hat{q}_h(s,a))e^{\beta r_h(s,a)} + \hat{q}_h(s,a)e^{\beta(r_h(s,a)+H-h)} + c_h(s,a)(e^{\beta(r_h(s,a)+H-h)} - e^{\beta r_h(s,a)})\right)\\
&= \frac{1}{\beta}\log\left(e^{\beta r_h(s,a)}[\hat{P}_h e^{\beta V_{h+1}}](s,a) + c_h(s,a)e^{\beta r_h(s,a)}(e^{\beta(H-h)} - 1)\right)\\
&< \frac{1}{\beta}\log\left(e^{\beta r_h(s,a)}[\hat{P}_h e^{\beta V'_{h+1}}](s,a) + c_h(s,a)e^{\beta r_h(s,a)}(e^{\beta(H-h)} - 1)\right)\\
&< \frac{1}{\beta}\log\left(e^{\beta r_h(s,a)}[\hat{P}_h e^{\beta V'_{h+1}}](s,a) + c_h(s,a)(e^{\beta(H+1-h)} - 1)\right) = Q'_h(s,a).
\end{aligned}$$

□