Risk-Sensitive Reinforcement Learning with Coherent Risk Measures

By

XINYI NI

DISSERTATION

Submitted in partial satisfaction of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in

Electrical and Computer Engineering

in the

OFFICE OF GRADUATE STUDIES

of the

UNIVERSITY OF CALIFORNIA

DAVIS

Approved:

_____

Lifeng Lai, Chair

_____

Bernard C. Levy

_____

Xin Liu

Committee in Charge

2025

Abstract

Reinforcement Learning (RL) is a branch of machine learning that focuses on training agents to make sequential decisions. By interacting with the environment, an RL agent learns optimal policies that guide its actions. While traditional RL algorithms focus primarily on maximizing expected rewards, they often overlook the risks associated with uncertain or adverse outcomes. This limitation is particularly problematic in high-stakes applications—such as autonomous driving, healthcare, and finance—where the consequences of poor decision-making can be significant. To address this gap, the field of risk-sensitive reinforcement learning has emerged, enhancing the safety and robustness of RL agents in uncertain environments.

This thesis explores advancements in risk-sensitive RL by developing novel algorithms, frameworks, and analysis techniques to address uncertainty and robustness in sequential decision-making.

One of the primary focuses is the application of Entropic Value-at-Risk (EVaR), a recently introduced risk measure, to RL. Unlike the conventional Conditional Value-at-Risk (CVaR), EVaR characterizes distributional uncertainty using Kullback-Leibler (KL) divergence, which better aligns with common practices in machine learning. This alignment enables a broader application in risk-sensitive RL problems where robustness to uncertainty is crucial. To achieve this, we propose value iteration and policy gradient algorithms that incorporate EVaR optimization within the Markov Decision Process (MDP) framework. The proposed algorithms are shown to converge and perform effectively through numerical experiments, demonstrating the practicality and relevance of EVaR for robust decision-making in RL.

Building upon this exploration of risk measures, we introduce the ϕ-Divergence-Risk (PhiD-R), a general class of coherent risk measures that includes existing risk measures such as CVaR and EVaR as special cases and extends the potential for RL applications by covering a broader range of risk preferences. The PhiD-R class allows the study of risk-sensitive RL using various ϕ-divergences, thus creating a flexible framework adaptable to multiple types

of uncertainty. For this class of risk measure, we develop a trajectory-based policy gradient method tailored specifically for PhiD-R, providing both theoretical convergence guarantees and practical validations through extensive simulation experiments. This work not only enhances our understanding of risk-sensitive learning but also contributes algorithms that are robust and versatile across a range of RL environments.

In addition to exploring risk measures, this dissertation examines the robustness of risk-sensitive RL under Robust MDPs (RMDPs). RMDPs provide a framework for decision-making under worst-case scenarios by optimizing over ambiguity sets, which define possible variations in the transition dynamics. While previous research on RMDPs has largely focused on risk-neutral approaches, we extend this work to risk-sensitive contexts. Leveraging the coherence properties of CVaR, we establish a connection between robustness and risk sensitivity, thereby enabling risk-sensitive RL techniques to solve robust decision-making problems. We further introduce a novel risk measure, NCVaR, specifically designed to handle state-action-dependent uncertainties, a common feature in real-world applications. Through value iteration algorithms and simulations, we validate that NCVaR optimization improves robustness in complex and uncertain RL environments.

The thesis also addresses a critical challenge in RL: exploration. In traditional reward-free RL, exploration is guided without a specific reward function, enabling adaptability across various reward settings. However, efficient exploration strategies in risk-sensitive RL are still underdeveloped. To fill this gap, we propose a risk-sensitive reward-free RL framework based on CVaR, aiming to balance efficient exploration with risk constraints. We develop the CVaR-RF-UCRL algorithm, designed to perform effective CVaR-based exploration under risk-sensitive criteria, and establish its performance guarantees by proving it is PAC with sample complexity upper bound. We further introduce two planning algorithms, CVaR-VI and CVaR-VI-DISC, and validate the approach with empirical experiments, demonstrating its utility in safe and efficient exploration. We also establish a lower bound on the sample complexity for any CVaR-RF algorithm.

Acknowledgement

First and foremost, I express my deepest gratitude to my advisor, Professor Lifeng Lai. Under his guidance, I was introduced to the fascinating world of reinforcement learning, which has shaped my research trajectory and academic growth. His unwavering support and encouragement have been invaluable. Professor Lai's warm kindness and steadfast belief in my potential gave me the strength to persevere and continue pushing forward. His passion for research has been an inspiring force, motivating me to explore deeply and aim higher. I am incredibly fortunate to have had the opportunity to learn and grow under his mentorship.

Apart from my advisor, I would like to extend my appreciation to my committee members, Professor Bernard C. Levy and Professor Xin Liu, for their invaluable insights and feedback. Their thoughtful comments and encouragement have been instrumental in refining my research and improving my results.

I would also like to acknowledge my research group members—Fuwei Li, Minhui Huang, Xinyang Cao, Puning Zhao, Xiaochuan Ma, Yulu Jin, Guanlin Liu, Parisa Oftadeh, Chenye Yang, Haodong Liang, Jipeng Li, Renxiang Huang, and Mo Lyu—for their support, collaboration, and technical advice throughout this journey.

To my friends, thank you for your constant support and understanding. Your presence and encouragement have been a source of strength. I am truly grateful for the laughter, advice, and companionship you have shared with me throughout this journey.

Finally, and most importantly, my heart overflows with gratitude for my parents and my sister. I am profoundly thankful for the countless sacrifices they made to support me through this journey. Their love, patience, and encouragement have been my foundation, and this accomplishment is as much theirs as it is mine. I am deeply appreciative of their belief in me and the strength they provided, allowing me to reach this milestone. I am blessed to have them by my side, and I dedicate this work to them with all my love and gratitude.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Decision making is a process of making choices by identifying a decision, gathering information, and assessing alternative resolutions and its goal is to identify an optimal strategy (a mapping from current system states to available actions), where the performance is incurred by a cost function [62]. The evaluation criteria that are deemed relevant to the decision makers is captured by this cost function. Reinforcement learning (RL) [94] is an area of machine learning where agents learn from interacting with the environment to determine the actions. The environment is typically stated as a Markov decision process (MDP) which will be described in Section 1.1. A common goal in solving these sequential decision making tasks is to determine an optimal policy that minimizes the expected total discounted cost, which is also named risk-neutral approach [15]. Although the risk-neutral approach is quite popular, it doesn't properly account for events that are rare but have serious consequences.

To address this potential issue, many applications[23, 24, 39, 40, 43, 44, 52, 55, 57, 61, 64, 73, 91, 96, 99, 105] focus on minimizing a risk-sensitive criterion rather than the risk-neutral criterion, which provides a promising approach to scenarios where it is important to control risks.

Various risk measures have been studied and applied in risk-sensitive decision-making, including Value-at-Risk (VaR) and Conditional Value-at-Risk (CVaR). In this work, we begin by presenting an overview of risk measures in Section 1.2. Following this, we review risk-sensitive

RL, discussing current solutions and their limitations in Section 1.3. To address the limitations of these existing works, this thesis introduces a novel application of a risk measure called Entropic Value-at-Risk (EVaR) to risk-sensitive RL and proposes a broader class of risk measures for RL applications. Additionally, we examine the robustness of risk-sensitive RL under model uncertainty and design more efficient exploration algorithms within a reward-free framework for risk-sensitive RL. Section 1.5 outlines the main contributions and organization of this thesis.

1.1 Markov Decision Processes

In RL, the underlying mathematical mode is the MDP, which represents a probabilistic sequential decision-making framework such that the set of transition probabilities to next states depend only on the current state and action of the system.

An MDP is represented by the tuple $(\mathcal{X}, \mathcal{A}, C, P, P_0, x_0, \gamma)$, where $\mathcal{X}$ denotes the state space, $\mathcal{A}$ is the action space, $C(x,a) \in [-C_{\max}, C_{\max}]$ represents a bounded deterministic cost, $P(\cdot|x,a)$ is the transition probability distribution, P_0 is the state distribution of the initial state x_0, and $\gamma \in [0,1]$ is the discount factor. For each state $x \in \mathcal{X}$, $\mathcal{A}(x)$ denotes the set of available actions. Note that the specific bound for the cost may vary according to different assumptions in this thesis, and each chapter will clearly specify these classifications. For convenience, we omit γ when $\gamma = 1$.

Intuitively, solving an MDP aims to determine some policies π (mappings from states to actions) under given cost functions. In order to propose the optimization problem formulations, here we define some feasible set of policies π. For $t \geq 1$, let $H_t = H_{t-1} \times \mathcal{A} \times \mathcal{X}$ with $H_0 = \mathcal{X}$ denote the space of possible histories up to time t and $h_t = (x_0, a_0, \ldots, x_{t-1}, a_{t-1}, x_t)$ is an element in H_t. For each time t, the policy π_t is a mapping from h_t to the probability distribution over the action space $\mathcal{A}$. Let $\Pi_{H,t}$ be the set of all t-step history-dependent policies, i.e., $\Pi_{H,t} := \{\pi_0 : H_0 \to \mathcal{A}, \pi_1 : H_1 \to \mathcal{A}, \ldots, \pi_t : H_t \to \mathcal{A} | \pi_j(\cdot|h_j) \in \mathcal{A}$ for all $h_j \in H_j, 1 \leq j \leq t\}$. Let $\Pi_H = \lim_{t\to\infty} \Pi_{H,t}$ be the set of all history-dependent policies. Similarly, we can define the Markovian policies as $\Pi_M = \lim_{t\to\infty} \Pi_{M,t}$ where $\Pi_{M,t} := \{\pi_0 :$

$\mathcal{X} \to \mathcal{A}, \pi_1 : \mathcal{X} \to \mathcal{A}, \ldots, \pi_t : \mathcal{X} \to \mathcal{A} | \pi_j(\cdot|x_j) \in \mathcal{A}$ for all $x_j \in \mathcal{X}, 1 \leq j \leq t\}$. One special case is the stationary Markovian policy denoted by $\Pi_{M,S}$, where the policies are time-homogeneous, i.e., $\pi_j = \pi$ for all $j \geq 0$. Compared with history-dependent policies, the stationary Markovian policies are more structured, which means the actions only depend on current state and its state-action mapping is time-independent. This makes the procedure of determining an optimal policy under stationary Markovian policies more computationally tractable. Commonly, the corresponding solution techniques involve dynamic programming algorithms [15] and policy gradient methods [23, 70]. In policy gradient methods, the stationary Markovian policy $\pi(\cdot|x)$ is parameterized by a κ-dimensional vector θ. Thus, the space of all such policies can be expressed as $\{\pi(\cdot|x, \theta) : x \in \mathcal{X}, \theta \in \Theta \subseteq \mathbb{R}^\kappa\}$, where Θ is a convex, compact set.

Let T denote the length of time horizons. The cost function under policy π for a given state x is defined as the total discounted cost accumulated by the agent when it starts at state x and follows policy π, i.e.,

$$J^\pi(x) = \sum_{t=0}^{T} \gamma^t C(x_t, a_t) \mid x_0 = x, \pi.$$

Similarly, for a state-action pair (x, a), the cost function is defined as

$$J^\pi(x, a) = \sum_{t=0}^{T} \gamma^t C(x_t, a_t) \mid x_0 = x, a_0 = a, \pi.$$

The expected values of the random variables $J^\pi(x)$ and $J^\pi(x, a)$, known as the value and action-value functions of policy π, are defined as

$$V^\pi(x) = \mathbb{E}[J^\pi(x)], \quad Q^\pi(x, a) = \mathbb{E}[J^\pi(x, a)].$$

In policy gradient methods, the policy π is parameterized by a vector θ, allowing the cost

functions to be expressed as:

$$J^{\theta}(x) = \sum_{t=0}^{T} \gamma^t C(x_t, a_t) \mid x_0 = x, \pi(\cdot|\cdot, \theta),$$

$$J^{\theta}(x, a) = \sum_{t=0}^{T} \gamma^t C(x_t, a_t) \mid x_0 = x, a_0 = a, \pi(\cdot|\cdot, \theta).$$

The corresponding value functions are defined as

$$V^{\theta}(x) = \mathbb{E}[J^{\theta}(x)], \quad Q^{\theta}(x, a) = \mathbb{E}[J^{\theta}(x, a)].$$

For convenience, we will use π-indexed functions throughout, except when discussing policy gradient methods.

In MDPs, a widely adopted optimization formulation is the risk-neutral criterion, where the objective function is the expected total discounted cost. The optimization problem is formulated as

$$\min_{\pi \in \Pi_H} \mathbb{E}[J^{\pi}(x_0)],$$

and for an infinite-horizon setting, the problem becomes:

$$\min_{\pi \in \Pi_H} \mathbb{E}\left[\lim_{T \to \infty} J^{\pi}(x_0)\right].$$

In [15], it is shown that the optimal policy of the above optimization problem is a stationary Markovian policy.

1.2 Overview of Risk Measures

1.2.1 Uncertainty in MDPs

Here we describe two sources of uncertainty while using the MDP model. The first one is inherent-uncertainty, which describes the cost variability due to the stochasticity of an MDP. The second one is the model-uncertainty, which comes from the inaccuracy of transition probability and cost of the MDP, more generally, it accounts for the errors in the representations of MDP. Both uncertainty are incurred by the total discounted cost random variable [24].

As mentioned in Section1.1, the risk-neutral criterion is widely used. However, despite its popularity, it doesn't take either the uncertainties of cost nor its sensitivity to modeling errors into account, which may significantly degrade the performance of the optimal policy [65] when there are uncertainties or modeling errors. The uncertainty of the cost can be addressed in risk-sensitive MDPs [47] by utilizing a risk measure rather than the risk-neutral expectation. A risk measure is a mapping from a random variable to a real value. Typically, the risk object is derived from the total discounted cost. The sensitive issue could be solved in robust MDPs by choosing some uncertainty sets to model the uncertainty and considering the worst case [75] over these uncertainty sets. [4] proposes an important concept named coherent risk measures, which satisfy four basic axioms: translation invariance, subadditivity, monotonicity and positive homogeneity. A useful property is that each coherent risk measure has a dual representation. [38] extends the concept of coherent risk measures by introducing the notion of convex measure or risk. They also provide the corresponding extension of the dual representation. [79] further shows that risk-sensitive MDP with certain coherent risk measures is equivalent to robust MDP of minimizing the worst-case expectation over the uncertainty set determined by the dual representation of the risk measure. Therefore, suitably choosing risk measure can decrease the influence of both issues at the same time.

Here, we present a real-life example to illustrate the importance of considering risk measures instead of risk-neutral approaches (definitions of VaR and CVaR are provided in Section 1.2).

Consider a financial portfolio optimization problem where an agent allocates capital among three assets: a risk-free bond with a fixed 2% annual return, a low-risk stock with an average

annual return of 5% (standard deviation: 2%), and a high-risk stock with an average annual return of 10% (standard deviation: 15%). Let the portfolio allocation under a risk-neutral criterion assign 10%, 20%, and 70% of the capital to the bond, low-risk stock, and high-risk stock, respectively, while the risk-sensitive criterion, optimizing the CVaR at the 95% confidence level, assigns 40%, 40%, and 20% to the respective assets. The annual portfolio return R is a weighted sum of the individual asset returns:

$$R = w_1 r_1 + w_2 r_2 + w_3 r_3,$$

where w_i are the portfolio weights and r_i are the annual returns of the assets. Assume the returns follow normal distributions: $r_1 = 0.02$, $r_2 \sim \mathcal{N}(0.05, 0.02^2)$, $r_3 \sim \mathcal{N}(0.10, 0.15^2)$. For convenience, we denote std as standard deviation.

Case 1: Risk-Neutral Criterion. With portfolio weights: $w = [0.1, 0.2, 0.7]$, the expected portfolio return is:

$$\mathbb{E}[R] = 0.1 \cdot 0.02 + 0.2 \cdot 0.05 + 0.7 \cdot 0.10 = 0.081\,(8.1\%).$$

For portfolio standard deviation, we have

$$\text{Var}[R] = 0.1^2 \cdot 0 + 0.2^2 \cdot 0.02^2 + 0.7^2 \cdot 0.15^2 = 0.01129,$$

$$\text{Std}[R] = \sqrt{0.01129} = 0.1063\,(10.63\%).$$

Using Monte Carlo simulation, the CVaR at the 95% confidence level is approximately -12%.

Case 2: Risk-Sensitive Criterion (CVaR Optimization at 95%). With portfolio weights: $w = [0.4, 0.4, 0.2]$, the expected portfolio return is

$$\mathbb{E}[R] = 0.4 \cdot 0.02 + 0.4 \cdot 0.05 + 0.2 \cdot 0.10 = 0.048\,(4.8\%).$$

For, portfolio standard deviation, we have

$$\text{Var}[R] = 0.4^2 \cdot 0 + 0.4^2 \cdot 0.02^2 + 0.2^2 \cdot 0.15^2 = 0.00292,$$

$$\text{Std}[R] = \sqrt{0.00292} = 0.054\,(5.4\%).$$

Using Monte Carlo simulation, the CVaR at the 95% confidence level is approximately -3%.

Under the risk-neutral criterion, the portfolio achieves a higher expected return of 8.1%, but with significantly higher volatility and a CVaR of -12%, indicating greater exposure to extreme losses. Conversely, the risk-sensitive criterion reduces the expected return to 4.8%, but significantly lowers the portfolio volatility and improves the CVaR to -3%, providing better protection against adverse outcomes.

1.2.2 Coherent Risk Measures

Consider a probability space $(\Omega, \mathcal{F}, P)$, where Ω is the set of all possible outcomes, $\mathcal{F}$ is a σ-algebra over Ω and P is a probability measure over $\mathcal{F}$. Let $\mathcal{Z}$ denote the space of random variables $Z : \Omega \to (-\infty, \infty)$ over the probability space $(\Omega, \mathcal{F}, P)$. A risk measure ρ is a mapping from a random variable $Z \in \mathcal{Z}$ to a real value. In risk-sensitive RL, Z usually presents the reward or cost and the goal is to determine the optimal strategies that minimize $\rho(Z)$. In the last few decades, many different risk measures have been proposed and investigated in the risk-sensitive decision making context. All these risk measures can be classified into two categories: coherent measures and non-coherent measures. A risk measure ρ is coherent if it satisfies the following properties mentioned in [4].

(P1) Translation Invariance: $\rho(Z + c) = \rho(Z) + c$ for any $Z \in \mathcal{Z}$ and $c \in \mathbb{R}$;

(P2) Subadditivity: $\rho(Z_1 + Z_2) \leq \rho(Z_1) + \rho(Z_2)$ for all $Z_1, Z_2 \in \mathcal{Z}$;

(P3) Monotonicity: If $Z_1, Z_2 \in \mathcal{Z}$ and $Z_1(w) \leq Z_2(w)$ for all $w \in \Omega$, then $\rho(Z_1) \leq \rho(Z_2)$;

(P4) Positive homogeneity: $\rho(\lambda Z) = \lambda\rho(Z)$ for all $Z \in \mathcal{Z}$ and $\lambda \geq 0$.

Another very useful property of coherent risk measures is the dual representation theorem [88],

which connects the risk-sensitiveness to robustness. The theorem can be expressed as: a risk measure ρ is coherent if and only if there exists a convex bounded and closed set $\mathcal{U}$ such that

$$\rho(Z) = \max_{\xi:\xi P \in \mathcal{U}(P)} \mathbb{E}_{\xi}[Z].$$

The result essentially states that any coherent risk measure is an expectation with respect to a worst-case function ξP, chosen adversarially from the risk envelope $\mathcal{U}(P)$ [24]. Examples of coherent measures include the CVaR and EVaR [2] etc. Examples of non coherent measures include variance, mean-standard-deviation and VaR etc [4].

In risk-sensitive decision making, more applications begin to consider optimization problems in which the objective function involves a coherent risk measure of the total discounted cost. The reason is that properties (P1)-(P4) ensure the "rationality" of single-period risk assessments. Take financial investment as an example: (P1) means that the deterministic part of an investment portfolio does not contribute to its risk; (P2) ensures that diversifying an investment will reduce its risk; (P3) guarantees that an asset with a higher cost for every possible scenario is indeed riskier; (P4) means that doubling a position in an asset doubles its risk. Moreover, as mentioned in Section 1.2.1, by using this representation of coherent risk, we can show that robust MDPs are equivalent to risk-sensitive MDPs while using coherent risk measure. Thus, both uncertainties can be solved by applying coherent risk measure to decision making.

1.2.3 Value-at-Risk and Conditional Value-at-Risk

In the following, we review risk measures that are directly related to our work. Let Z be a bounded random variable (i.e., $\mathbb{E}[|Z|] < \infty$) on the probability space $(\Omega, \mathcal{F}, P)$ with the cumulative distribution function (CDF) $F(z) = P(Z \leq z)$. The VaR [4] at confidence level $\alpha \in [0, 1]$ is the $1 - \alpha$ quantile of Z. Since we interpret Z as a cost in this thesis, VaR is defined as:

$$\text{VaR}_{\alpha}(Z) = \inf\{z | F(z) \geq \alpha\}.$$

The visual illustration of VaR can be founded in Figure 1.1. VaR is a well-known risk measure of risk-sensitive MDPs. However, VaR is not coherent due to the lack of subadditivity and convexity [4]. Furthermore, VaR is unstable and difficult to optimize when the costs are not normally distributed [85, 86].

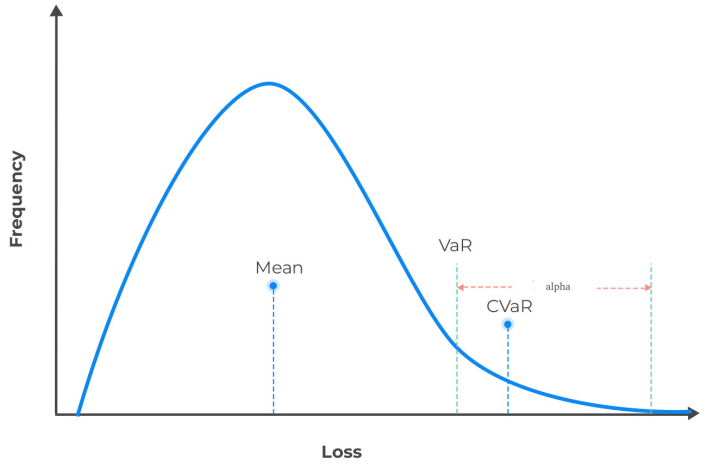


Figure 1.1: Illustration of VaR and CVaR.

To address these shortcomings, a new risk measure called CVaR has been developed [85, 86]. CVaR is defined as the mean of the worst $\alpha\%$ of values of Z, i.e.,

$$\text{CVaR}_\alpha(Z) = \inf_{b\in\mathbb{R}} \left\{ b + \frac{1}{\alpha}\mathbb{E}_P[(Z-b)^+] \right\},$$

where $(z)^+ = \max(z, 0)$. From its definition, we can see that CVaR_α is decreasing in α, i.e, CVaR_α tends to $\max(Z)$ as α decreasing to 0 and $\text{CVaR}_1(Z)$ equals $\mathbb{E}(Z)$. CVaR is able to quantify risk beyond VaR and is a coherent risk measure. Due to these advantages, CVaR has been extensively applied to RL problems [23, 24, 40, 52, 91, 96, 97, 105].

As mentioned above, for each coherent risk measure, there is a useful alternative dual representation [2]. Before introducing the dual representation of CVaR, we introduce some notation. Let Q be another probability measure on $(\Omega, \mathcal{F})$, Q is said to be absolutely continuous with respect to P (denoted by $Q \ll P$) if $P(A) = 0$ implies $Q(A) = 0$ for any measurable set $A \in \mathcal{F}$. If $Q \ll P$, then by probability theory there is a well-defined Radon-Nikodym derivative

$\frac{dQ}{dP}$ and the alternative dual representation for CVaR can be written as [3]:

$$\text{CVaR}_\alpha(Z) = \sup_{Q \in \mathcal{U}_{\text{CVaR}}} \mathbb{E}_Q(Z), \tag{1.1}$$

where

$$\mathcal{U}_{\text{CVaR}} = \left\{ Q \ll P : \frac{dQ}{dP} \in \left[0, \frac{1}{\alpha}\right] \right\}. \tag{1.2}$$

From this dual representation, $\text{CVaR}_\alpha(Z)$ can be interpreted as the largest mean of Z computed using distribution Q that is in the neighborhood of P defined in (1.2). In the dual representation of CVaR, the uncertainty set of the CVaR optimization is defined by distributions whose Radon-Nikodym derivative is constrained to a certain range. While the uncertainty set corresponding to CVaR is certainly relevant for some RL applications [24], it is a less common way to define distribution neighborhood in machine learning applications and hence its interpretation for machine learning applications is less natural. This leads to the question of whether we can apply risk measures, whose uncertainty sets in their dual presentations are defined using widely used metrics and have more natural interpretations in machine learning applications, to design risk-sensitive RL algorithms.

1.2.4 Entropic Value-at-Risk

[2] propose a risk measure named EVaR from the Chernoff inequality of the VaR. Let L_{M^+} be the set of all Borel measurable functions $Z : \Omega \to \mathbb{R}$ whose moment generating function $M_Z(t) = \mathbb{E}_P\left[e^{\nu Z}\right]$ exists for $\nu \geq 0$. The EVaR of a random variable $Z \in L_{M^+}$ with confidence level α is defined as

$$\text{EVaR}_\alpha(Z) = \inf_{\nu > 0} \left\{ \nu^{-1} \ln(M_Z(\nu)) - \nu^{-1} \ln \alpha \right\}. \tag{1.3}$$

EVaR is shown to be the tightest upper bound for both VaR and CVaR. Similar to CVaR, EVaR is coherent risk measure and EVaR_α is decreasing in α, i.e, EVaR_α tends to $\max(Z)$ as α decreasing to 0 and $\text{EVaR}_1(Z)$ equals $\mathbb{E}(Z)$.

Since EVaR is a coherent risk measure, one appealing feature of EVaR is its dual representation [2]:

$$\text{EVaR}_\alpha(Z) = \sup_{Q \in \mathcal{U}_{\text{EVaR}}} \mathbb{E}_Q(Z), \tag{1.4}$$

where

$$\mathcal{U}_{\text{EVaR}} = \{Q \ll P : D_{KL}(Q \parallel P) \leq -\ln \alpha\}.$$

Here D_{KL} refers to Kullback-Leibler (KL) divergence between probability measures Q and P. Since KL divergence is also called relative entropy, this measure is then called entropic value-at-risk. From (1.4), we can see that $\text{EVaR}_\alpha(Z)$ has a very nice interpretation: it is the largest mean of Z computed using distribution Q, who is in the $-\ln \alpha$-neighborhood (defined using KL divergence) of P. Compared with the dual representation of CVaR, it's more common and natural to use KL divergence rather than the Radon-Nikodym derivative to define the distance between distributions in machine learning applications. Therefore, EVaR might be a natural risk measure for RL.

We now would like to comment on another risk measure named entropic risk measure. For $Z \in L_M$ with $\nu \neq 0$, the entropic risk [37] is defined as:

$$\rho_{\text{entropic}} = \nu^{-1} \log(M_Z(-\nu)).$$

Although having entropic in both names, entropic risk measure and EVaR are quite different not only in the mathematical interpretation but also in their properties. Different from the t in EVaR acting as a positive variable for getting the infimum, the parameter t in entropic risk measure is already given by the user depending on the user's tolerance towards risk. Although entropic risk measure is convex, it is not coherent [47]. Moreover, despite the popularity of entropic risk measure in literature, its practical applications have proven to be problematic in [16, 36, 42] since it's very sensitive to errors in the underlying distribution.

1.3 Risk-Sensitive Reinforcement Learning

Due to these uncertainties mentioned in Section 1.2.1 and the increased awareness of events that have small probability but high consequences, many applications focus on minimize a risk-sensitive criterion rather than the risk-neutral criterion in decision making under the MDP framework. The optimization over a risk-sensitive criterion is called risk-sensitive decision making.

In this literature, researchers are aiming to find a 'good' risk criterion such that is both conceptually meaningful and computationally tractable. The earliest risk measure used in risk-sensitive MDPs is the exponential risk measure $\frac{1}{\beta}\mathbb{E}[e^{\beta Z}]$, where Z is the total cost and the parameter β is determined by the user to control its tolerance towards risk [47]. However, the choice of suitable β is often challenging. This issue motivated several other approaches, such as considering the maximization of a strictly concave function of the distribution of the terminal state [25] and variance-related risk measures in [92]. Numerous alternative risk measures have recently been investigated under this framework. VaR and CVaR are two promising such alternatives. Recall the definition of VaR and CVaR, we know that they both aim at quantifying costs that might be encountered in the tail of the distribution of cost, despite in different ways. Both VaR and CVaR have been studied in risk-sensitive MDPs [8, 18, 22–24, 80].

RL techniques are also implemented to solve risk-sensitive decision making, which is called risk-sensitive RL. Widely used RL techniques include dynamic programming [89] and policy gradient [9, 66, 103]. The term dynamic programming (DP) refers to a collection of algorithms that can be used to compute optimal policies. The classical DP includes policy evaluation, policy improvement, policy iteration and value iteration and it can be applied to problems with a few millions of states. Asynchronous DP can be applied to larger problems but overall, DP is not practical enough to very large problems [94]. In policy gradient, the policies are parameterized by a parameter vector and policy search is performance via gradient methods. Risk-sensitive RL algorithms based on these RL techniques and risk measures have also been research in [29, 58, 69, 81, 96].

1.4 Research Motivation and Problems Addressed

Despite the growth of risk-sensitive RL approaches, the field still faces substantial challenges in ensuring robustness, efficiency, and adaptability across diverse scenarios, particularly in settings where model parameters are uncertain or where efficient exploration is crucial.

This thesis tackles four primary problems:

1) **Interpretability and Computational Tractability of Risk Measures:** Many conventional risk measures, such as VaR and Conditional CVaR, have limitations in interpretability or computational efficiency, especially when applied to RL. VaR is not coherent, making it difficult to optimize reliably, while CVaR's dual representation lacks a natural interpretation in many machine learning contexts, complicating its application to RL. This raises the need for alternative risk measures that can balance coherence, interpretability, and computational feasibility.

2) **Lack of a Unified Risk-Sensitive Framework for Flexibility:** Current RL methods often rely on fixed risk measures, limiting flexibility to adapt to diverse risk preferences or nuanced decision-making requirements. These approaches may not generalize well to a range of coherent risk measures, highlighting the need for a framework that supports diverse measures while maintaining efficiency and robustness.

3) **Ensuring Robustness Under Model Uncertainty:** Traditional RL often assumes fixed model parameters, but real-world applications commonly face uncertainties in transition probabilities and costs, leading to model inaccuracies and degraded performance. While Robust MDPs (RMDPs) offer a solution by optimizing policies against worst-case scenarios, most RMDP research assumes risk-neutral objectives. Extending RMDPs to handle risk-sensitive objectives introduces complexity, particularly when ambiguity sets are decision-dependent, where the uncertainty itself varies with actions taken.

4) **Efficient Exploration Without a Predefined Reward Function:** Exploration is fundamental to effective learning, especially in RL applications where agents must explore unfamiliar environments. In risk-sensitive RL, exploration poses unique challenges, as the standard risk-neutral strategies may not adequately address the need to mitigate low-probability,

high-impact events. Moreover, the reward-free setting—where a predefined reward function is absent—requires the agent to gather information that is broadly applicable across potential rewards, creating an urgent need for sample-efficient algorithms that can balance exploration and risk sensitivity.

Addressing these challenges requires innovative methods that integrate robust theoretical foundations with practical applicability, advancing risk-sensitive RL toward broader, more reliable real-world deployment.

1.5 Contributions and Outline

The main contributions of this thesis are organized across four chapters, each addressing a specific problem in risk-sensitive RL.

In Chapter 2, we solve the interpretability issue by introducing the application of a novel risk measure, EVaR, to risk-sensitive RL. EVaR provides a more natural and interpretable measure of uncertainty than conventional measures, such as CVaR, by defining uncertainty in terms of KL divergence. We develop two approaches: EVaR optimization with value iteration (EVaR-VI) and EVaR optimization with policy gradient (EVaR-PG). We validate these approaches with theoretical proofs and numerical experiments, demonstrating the practicality of EVaR for RL applications. These results have been published in [70, 71].

Chapter 3 proposes a generalized risk-sensitive RL framework using ϕ-Divergence-Risk (PhiD-R), a flexible class of coherent risk measures that includes widely used measures such as CVaR and EVaR. To solve RL problems under this new framework, we develop a trajectory-based policy gradient algorithm that efficiently estimates gradients for PhiD-R and converges to locally optimal policies. This framework ensures flexibility and computational efficiency, enhancing decision-making robustness across diverse risk measures while maintaining computational tractability. This work has been submitted to IEEE Transactions on Information Theory [72].

Chapter 4 investigates the robustness of risk-sensitive RL within the RMDP framework, focusing on CVaR as a risk measure. We develop methods to determine optimal policies that minimize the worst-case CVaR within an ambiguity set of transition probabilities. Furthermore, we extend this analysis to handle decision-dependent uncertainty, introducing a new risk measure, NCVaR, that enhances robustness under dynamic uncertainty settings. We validate our approach with theoretical analysis and simulations, highlighting the potential of NCVaR to improve robustness in risk-sensitive RL. This study has been published in [73].

In Chapter 5, we address the challenge of efficient exploration in risk-sensitive RL within a reward-free framework. We propose a CVaR-based risk-sensitive reward-free RL framework (CVaR-RF RL) to collect sample-efficient exploration trajectories that are applicable to any reward function. Our exploration algorithm, CVaR-RF-UCRL, achieves near-optimal sample complexity, and we introduce a planning algorithm, CVaR-RF-planning, equipped with CVaR-VI (CVaR optimization with value iteration) and CVaR-VI-DISC (CVaR-VI with discretization) to ensure practical applicability. This framework advances exploration efficiency in risk-sensitive RL, a crucial factor in deploying RL in diverse real-world settings. These results have been published in [74].

Chapter 2

Risk-Sensitive Reinforcement Learning with EVaR

2.1 Introduction

In risk-sensitive MDPs, one well-known risk measure is VaR. However, VaR is not coherent due to the lack of subadditivity and convexity [4]. Furthermore, VaR is unstable and difficult to optimize when the costs are not normally distributed [85, 86]. To address these shortcomings, Rockafellar and Uryasev developed a new risk measure called CVaR in [85] and [86]. CVaR is able to quantify risk beyond VaR and is a coherent risk measure. Due to these advantages, CVaR has been extensively applied to RL problems [23, 24, 40, 52, 91, 96, 97, 105].

However, as detailed in Section 1.2.3, in the dual representation of CVaR, the uncertainty set of the CVaR optimization is defined by distributions whose Radon-Nikodym derivative is constrained to a certain range. While the uncertainty set corresponding to CVaR is certainly relevant for some RL applications [24], it is a less common way to define distribution neighborhood in machine learning applications and hence its interpretation for machine learning applications is less natural. This leads to the question of whether we can apply risk measures, whose uncertainty sets in their dual presentations are defined using widely used metrics and have more natural interpretations in

machine learning applications, to design risk-sensitive RL algorithms. One promising coherent risk measure is EVaR developed recently by Ahmadi et al. [2]. EVaR is a coherent risk measure that is derived from the Chernoff inequality for the VaR. In particular, EVaR is the tightest upper bound for both VaR and CVaR [2].

One appealing feature of EVaR is the uncertainty set in its dual representation. In particular, [2] shows that the uncertainty set in the dual representation of EVaR is defined by distributions whose KL distance to the nominal distribution is less or equal to a certain level. As a result, minimizing EVaR is equivalent to minimizing the worst-case expectation over distributions whose KL distance to the nominal distribution is less or equal to a certain level. As KL distance is widely used to define distances between distributions in machine learning applications, EVaR appears to be a natural risk measure to use for RL problems.

Considering all these advantages of EVaR, we introduce a new approach to determine the optimal policies for risk-sensitive decision making problem based on the optimization of EVaR. To the best of our knowledge, this is the first time that EVaR is applied in risk-sensitive MDPs. In our approach, the goal is to determine the optimal policies that minimize the EVaR value of the total discounted cost.

Chapter Contribution: In this chapter, we develop two approaches to solve the EVaR optimization problem: a value iteration-based method, termed EVaR-VI, and a policy gradient-based method, termed EVaR-PG.

In EVaR-VI, due to the coherent property of EVaR, we can apply the alternative dual representation for EVaR in [2] and then the problem becomes an optimization problem over an uncertainty set. However, in the uncertainty set, we need to know the probability distribution of the total discounted cost under different policies, which is quite hard to obtain. To address this issue, we utilize the conditional decomposition theorem of version independent risk functions in [82] to develop the conditional EVaR decomposition theorem that reveals the connection of EVaR computation between the current state and the next state. After utilizing conditional EVaR decomposition theorem, the EVaR problem becomes an optimization problem over the

uncertainty set defined on the one-step transition kernel of the underlying MDP using KL distance. Following the idea of dynamic programming, we define value function and Bellman operator for EVaR. Similar with the Bellman operator, we show that the EVaR Bellman operator also has the monotonicity, transition invariance and contraction properties, which guarantees the existence of the unique fixed-point solution. Combining with these useful properties, we develop an EVaR value iteration algorithm, which recursively update the EVaR value at each time step and gradually converge to the optimize value. According to the optimal value function, we can then construct a method to extract the optimal policy as a stationary Markovian policy, which is more structured and easier for implementation. However, using the conditional EVaR decomposition theorem will bring in an augmented continuous space representing the confidence level, which makes our algorithm not practical enough. To improve the practicality, we follow the idea of linear interpolation in [24] to develop an approximate value iteration algorithm, in which we choose some points of the confidence level rather than using its whole continuous space. Similar with the EVaR value iteration algorithm, we also define the interpolated EVaR Bellman operator and show that it also has these useful properties as mentioned in EVaR Bellman operator. Therefore, we can follow the same procedure to develop the approximate version of the value iteration algorithm and analyze the error bounds between these two algorithms. Furthermore, for the scenarios where we do not know the transition kernel of the underlying MDP model, we adapt the sample average approximation (SAA) approach introduced in [88] and [98] to estimate the transition probability and design the sample based EVaR algorithm following the same procedure. Moreover, we validate the proposed algorithms using numerical examples.

In EVaR-PG, we follow the idea of policy gradient method to minimize the EVaR value of the total discounted cost. In policy gradient method, policies are parameterized by a vector and we use gradient descent in the parameter space to search for optimal policies. In this work, we propose a trajectory-based EVaR policy gradient algorithm. We first reformulate the EVaR optimization problem by plugging in the definition of EVaR, which enables us to compute the gradient more easily. The general idea is to descent in the policy parameter as well as the parameter that comes

from EVaR's definition w.r.t gradients to find a local minimum of the EVaR optimization problem. In order to ensure the usability of this approach, we generate sample trajectories to estimate the gradients. Then we develop updates rules for these parameters, in which projections are adopted and the stepsizes are chosen to satisfy certain conditions to ensure the convergence of this approach. For the convergence analysis, we first regard these updates as a multi-time scale discrete stochastic approximation and show the sequences converge to the solution of the corresponding continuous time system with different speed. By applying Lyapunov analysis, we further show that these sequences converge to the local asymptotically stable points, which guarantees that the solution is a local minimum. Numerical examples for this algorithm are also provided by applying this approach to the optimal stopping problem.

Chapter Organization: The remainder of this chapter is organized as follows. Section 2.2 details the EVaR-VI approach. Specifically, Section 2.2.1 introduces the problem formulation, followed by Section 2.2.2, which presents the value iteration algorithms and a practical approximation using linear interpolation. Section 2.2.3 addresses cases where the underlying MDP model is unknown, providing a sample-based algorithm. Section 2.3 covers the EVaR-PG approach. In Section 2.3.1, we provide mathematical preliminaries for EVaR-PG, followed by the problem formulation in Section 2.3.2. Section 2.3.3 introduces the trajectory-based EVaR policy gradient method. In Section 2.4, we present numerical simulation results for both EVaR-VI and EVaR-PG. Finally, concluding remarks are provided in Section 2.5.

2.2 EVaR Optimization with Value Iteration: EVaR-VI

In RL, a common approach to solving sequential decision-making problems is value iteration. This method estimates the value of each state-action pair, enabling the agent to derive an optimal policy. Value iteration is effective in discrete action spaces, providing a straightforward framework to converge on an optimal strategy. It is especially useful in environments with well-defined transitions and rewards, where accurate value estimates can be efficiently computed. Here, we

apply EVaR to risk-sensitive RL and solve the optimization problem using value iteration.

2.2.1 Problem Formulation

The problem formulation of EVaR optimization in risk-sensitive RL can be written as

$$\min_{\pi \in \Pi_H} \text{EVaR}_\alpha \left(\lim_{T \to \infty} J^\pi(x_0) \right), \tag{2.1}$$

where $\pi = \{\pi_0, \pi_1, \dots\}$ is the policy sequence with action $a_t = \pi_t(h_t)$ for $t = \{0, 1, \dots\}$.

Now, let P_S be the true probability measure of the total discounted cost under policy π and Q_S denote another probability measure over this space. Using the dual representation (1.4) of EVaR, we can write the optimization problem (2.1) as

$$\min_{\pi \in \Pi_H} \sup_{Q_S \in \mathcal{U}_{\text{EVaR}}(\alpha, P_S)} \mathbb{E}_{Q_S} \left(\lim_{T \to \infty} J^\pi(x_0) \right),$$

where

$$\mathcal{U}_{\text{EVaR}}(\alpha, P_S) = \{Q_S \ll P_S : D_{KL}(Q_S \parallel P_S) \leq -\ln \alpha\}.$$

However, it is challenging to optimize over this uncertainty set on the probability distribution of the total discounted cost. As will be discussed in the sequel, we will solve this problem by using the EVaR decomposition theorem proposed in Section 2.2.2, which reveals the connection between the current state and next state in EVaR computation and allows us to optimize over the uncertainty set defined on the transition kernel $P(\cdot|x, a)$ using KL distance. Note that in standard RL, we only aim to minimize the total discounted cost under the transition kernel $P(\cdot|x, a)$. Now with EVaR and its dual representation, the objective is to minimize the worst cost for all kernels in the neighborhood of $P(\cdot|x, a)$ as defined in KL distance, so as to achieve robustness.

2.2.2 Value Iteration for EVaR

In order to solve the primary optimization problem (2.1), we follow the idea of dynamic programming and apply the decomposition theorem of version independent risk measures used in [24] [82]. One important function in RL is the Bellman operator, which describes a recursively update for value function. Our approach follows the similar idea to derive the EVaR Bellman operator and then uses the value iteration process to obtain the optimal solution of (2.1).

To begin with, we introduce the decomposition theorem for conditional EVaR. Firstly, equipped with the dual representation for EVaR in [2] and the definition of conditional risk measures in [82], the conditional EVaR at random confidence level can be defined as following.

Definition 1. Let $\mathcal{F}_t$ be a sub-σ-algebra over the space (Ω, P), i.e., $\mathcal{F}_t \subset \mathcal{F}$ and ξ_t be a measurable random variable w.r.t. $\mathcal{F}_t$, then the conditional EVaR with confidence level $\alpha \in [0, 1]$ is defined as

$$\text{EVaR}_\alpha(Z|\mathcal{F}_t) = \operatorname{esssup} \mathbb{E}_P(\xi_t Z|\mathcal{F}_t),$$

where the $'esssup'$ is taken over the set $\{\xi_t : \mathbb{E}[\xi_t|\mathcal{F}_t] = 1, D_{KL}(\xi_t P||P) \leq -\ln \alpha\}$.

Then, we introduce *version independent* risk measures mentioned in [82]. Let Z_1 and Z_2 be two random variables in $\mathcal{Z}$, then a risk measure ρ is *version independent* if $\rho(Z_1) = \rho(Z_2)$ whenever Z_1 and Z_2 shares the same law, i.e., $P(Z_1 \leq z) = P(Z_2 \leq z)$ for all $z \in \mathbb{R}$. By Corollary 3.1 in [2], we know EVaR is a version independent risk functional. Now, we can apply Theorem 21 in [82] to propose the EVaR decomposition theorem.

Theorem 1. For any $\tau > t \geq 0$, let $\mathcal{F}_t \subset \mathcal{F}_\tau$ be two sub-σ-algebra of $\mathcal{F}$. The conditional EVaR at random confidence level α ($\alpha \in [0, 1]$ a.s.) obeys the nested decomposition

$$\text{EVaR}_\alpha(Z|\mathcal{F}_t) = \operatorname{esssup} \mathbb{E}_P[\xi_\tau \cdot \text{EVaR}_{\alpha;\xi_\tau}(Z|\mathcal{F}_\tau)|\mathcal{F}_t]$$

where the essential supremum is taken among all feasible dual random variables ξ_τ measurable

with respect to $\mathcal{F}_\tau$.

Remark 1. In this chapter, Q and P are two probability mass functions (PMFs) and P is the true transition probability of the underlying MDP model. Since we are more interested in the EVaR decomposition between the current state x_t and the next state x_{t+1} under policy π, here we choose $\mathcal{F}_\tau$ to be H_{t+1} and $\mathcal{F}_t$ to be H_t. Therefore, ξ_τ can be represented as

$$\xi(x_{t+1}) = \frac{Q(x_{t+1}|x_t, a_t)}{P(x_{t+1}|x_t, a_t)} \geq 0$$

for any $t \geq 0$, where a_t is the action induced by π at x_t. Recall the uncertainty set in EVaR dual representation,

$$\mathcal{U}_{\text{EVaR}} = \{Q \ll P : D_{KL}(Q \parallel P) \leq -\ln \alpha\}.$$

Note that in discrete case, the KL distance is

$$D_{KL}(Q \parallel P) = \sum_{x_{t+1} \in \mathcal{X}} Q(x_{t+1}|x_t, a_t) \log \frac{Q(x_{t+1}|x_t, a_t)}{P(x_{t+1}|x_t, a_t)}.$$

Inserting $Q(x_{t+1}|x_t, a_t) = \xi(x_{t+1}) \cdot P(x_{t+1}|x_t, a_t)$ to the above equation and using the fact that Q is a PMF, then we know $\xi(x_{t+1})$ should be in the set

$$\mathcal{U}_{\text{EVaR}}(\alpha, P(\cdot|x_t, a_t)) = \Big\{\xi : \sum_{x_{t+1} \in \mathcal{X}} \xi(x_{t+1}) P(x_{t+1}|x_t, a_t) \log \xi(x_{t+1}) \leq -\ln \alpha, \sum_{x_{t+1} \in \mathcal{X}} \xi(x_{t+1}) P(x_{t+1}|x_t, a_t) = 1\Big\}.$$

Then the decomposition in Theorem 1 can be rewritten as

$$\text{EVaR}_\alpha(Z|H_t, \pi) = \operatorname{esssup} \mathbb{E}_P[\xi(x_{t+1}) \cdot \text{EVaR}_{\alpha\xi(x_{t+1})}(Z|H_{t+1}, \pi)|H_t, \pi], \tag{2.2}$$

where the $'$esssup$'$ is taken over $\xi \in \mathcal{U}_{\text{EVaR}}(\alpha, P(\cdot|x_t, a_t))$.

Note that the $'$esssup$'$ can be replaced by $'$max$'$ since the set $\mathcal{U}_{\text{EVaR}}$ is convex and compact.

Theorem 1 establishes a connection between the current state and the next state for EVaR computation. Comparing with directly computing EVaR value based on its definition, which involves the sum of infinitely many random variables and an uncertainty set depending on the policy, it provides a recursive method to compute EVaR that involves optimization over uncertainty set of the one-step transition kernel $P(\cdot|x,a)$. Due to the difference of confidence level on both side in equation (8), following the idea in [24], we augment the state space $\mathcal{X}$ with an additional continuous space $\mathcal{Y} = (0,1]$, which represents the space of confidence level. Following the idea of standard dynamic programming, we define the value function for EVaR as follows.

Definition 2. For any $x \in \mathcal{X}, y \in \mathcal{Y}$, the value-function $V(x,y)$ is defined as:

$$V(x,y) = \min_{\pi \in \Pi_H} \text{EVaR}_y \left(\lim_{T \to \infty} J^\pi(x) \right). \tag{2.3}$$

Equipped with Theorem 1 and Definition 2, we can define the EVaR Bellman operator.

Definition 3. The EVaR Bellman operator $\mathbf{T} : \mathcal{X} \times \mathcal{Y} \to \mathcal{X} \times \mathcal{Y}$ is defined as:

$$\mathbf{T}[V](x,y) = \min_{a \in \mathcal{A}} \left[C(x,a) + \gamma \max_{\xi \in \mathcal{U}_{\text{EVaR}}(y, P(\cdot|x,a))} \sum_{x' \in \mathcal{X}} \xi(x') V(x', y\xi(x')) P(x'|x,a) \right]. \tag{2.4}$$

Here we introduce some useful properties of the EVaR Bellman operator.

Lemma 1. The Bellman operator $\mathbf{T} : \mathcal{X} \times \mathcal{Y} \to \mathcal{X} \times \mathcal{Y}$ has the following properties:

(1) *Monotonicity*: If $V_1 \leq V_2$ component-wisely, then $\mathbf{T}[V_1] \leq \mathbf{T}[V_2]$.

(2) *Transition Invariance*: For a constant c, $\mathbf{T}[V + c] = \mathbf{T}[V] + \gamma c$.

(3) *Contraction*: $\| \mathbf{T}[V_1] - \mathbf{T}[V_2] \|_\infty \leq \gamma \| V_1 - V_2 \|_\infty$, where $\| f \|_\infty = \sup_{x \in \mathcal{X}, y \in \mathcal{Y}} |f(x,y)|$.

(4) *Concavity preserving in y*: For any $x \in \mathcal{X}$, suppose $yV(x,y)$ is concave in $y \in \mathcal{Y}$. Then the maximization problem in (2.4) is concave. Furthermore, $y\mathbf{T}[V](x,y)$ is concave in y.

Proof. Please refer to Appendix A.1 for details. □

Similar with standard dynamic programming, Property 3 shows that the EVaR Bellman operator is contraction, which is important and useful for the design of convergent value iteration

algorithms based on EVaR. Property 4 indicates that the optimization problem in our value iteration update process is concave and therefore computationally tractable.

After defining the Bellman operator for EVaR, we need to determine the optimal condition and the optimal policy. In the following theorem, we show that for any $x \in \mathcal{X}$ and $y \in \mathcal{Y}$, the fixed point solution of $\mathbf{T}[V](x, y) = V(x, y)$ exists and it is unique. Moreover, the solution for the original optimization problem (2.1) is equal to the fixed point solution with $x_0 = x$ and $\alpha = y$.

Theorem 2. For any $(x, y) \in \mathcal{X} \times \mathcal{Y}$, $\mathbf{T}[V](x, y) = V(x, y)$ has a unique solution $V^*(x, y)$. Furthermore, this unique solution is equal to the optimal value of (2.1), i.e.,

$$V^*(x, y) = \min_{\pi \in \Pi_H} \text{EVaR}_y \left(\lim_{T \to \infty} J^\pi(x) \right). \tag{2.5}$$

Proof. Please refer to Appendix A.2 for details. □

We now discuss how to determine the optimal policy from V^*. Although the original optimization problem (2.1) is based on history-dependent policies, we can show that the optimal condition in Theorem 2 can be obtained by following a stationary Markovian policy, which can be constructed as a greedy policy with respect to the optimal condition V^*. Compared to historic-dependent policies, stationary Markovian policies are more structured, i.e., actions only depend on current states and the mappings from states to actions are time-independent, and hence are easier for implementation.

Theorem 3. Given initial conditions x_0, $y_0 = \alpha$ and the unique fixed-point solution $V^*(x, y)$ for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$, let u^* be a stationary Markovian policy defined as:

$$u^*(x_k, y_k) = a_k^*, \forall k \geq 0, \tag{2.6}$$

and for $k \geq 1$, the state transitions are

$$x_k \sim P\big(\cdot\,|x_{k-1}, a^*_{k-1}\big), y_k = y_{k-1}\xi_{x_{k-1},y_{k-1},a^*_{k-1}}(x_k), \tag{2.7}$$

where a^* and $\xi_{x,y,a^*}(\cdot)$ are solutions of the min-max optimization problem in $\mathbf{T}[V^*](x,y)$. Then u^* is an optimal policy for problem (2.1) with initial state x_0 and confidence level α.

Proof. Please refer to Appendix A.3 for details. □

Equipped with Theorem 2 and Theorem 3, we can now design a value iteration process to solve the EVaR optimization problem in (2.1).

Algorithm 2.1 EVaR Value Iteration

1: **Initialization**: for any $x \in \mathcal{X}$ and $y \in \mathcal{Y}$, arbitrarily choose $V_0(x,y)$.
2: **for** $t = 0, 1, 2, \ldots$ **do**
3: **for** all $(x,y) \in \mathcal{X} \times \mathcal{Y}$ **do**
4: recursively applying the EVaR Bellman operator as

$$V_{t+1}(x,y) = \mathbf{T}[V_t](x,y),$$

5: **end for**
6: **end for**
7: Get the optimal value function by $V^*(x,y) = \lim_{t\to\infty} V_t(x,y)$.
8: Selecting the specific initial state x_0 and confidence level α, the solution of EVaR optimization problem can be immediately obtained as $V^*(x_0,\alpha)$.
9: Following Theorem 3, one can derive an optimal Markovian policy w.r.t $V^*(x,y)$.

However, Algorithm 2.1 is not practical enough due to the augmented continuous space $\mathcal{Y}$. To address this issue, we follow the idea of applying linear interpolation from the paper of CVaR in [24]. Moreover, in order to ensure the computational tractability of our approach, the initial value function should satisfy the following assumption to preserve the concavity of the EVaR Bellman operator $\mathbf{T}$.

Assumption 1. The initial value function $V_0(x,y)$ satisfies the following properties:

(1) $yV_0(x,y)$ is concave in $y \in \mathcal{Y}$;

(2) $V_0(x,y)$ is continuous and bounded in $y \in \mathcal{Y}$ for any $x \in \mathcal{X}$.

In the linear interpolation, for the confidence level, we choose a finite set from the continuous space $\mathcal{Y}$. For each $x \in \mathcal{X}$, let $N(x)$ be the number of interpolation points of confidence level and the corresponding set is $Y(x) = \{y_1, y_2, \ldots, y_{N(x)}\} \in [0,1]^{N(x)}$ with $y_1 = 0$ and $y_{N(x)} = 1$. Then the linear interpolation of the concave function $yV(x,y)$ can be written as

$$\mathcal{I}_x[V](y) = y_i V(x, y_i) + \frac{y_{i+1}V(x, y_{i+1}) - y_i V(x, y_i)}{y_{i+1} - y_i}(y - y_i)$$

where $y_i = \max\{y' \in Y(x) : y' \leq y\}$ and y_{i+1} is the closet point such that $y \in [y_i, y_{i+1}]$.

Now we can define the interpolated Bellman operator as follows:

$$\mathbf{T}_{\mathcal{I}}[V](x,y) = \min_{a \in \mathcal{A}} \left[C(x,a) + \gamma \max_{\xi \in \mathcal{U}_{\text{EVaR}}(y, P(\cdot|x,a))} \sum_{x' \in \mathcal{X}} \frac{\mathcal{I}_{x'}[V](y\xi(x'))}{y} P(x'|x,a) \right]. \tag{2.8}$$

Notice that when the confidence level y tends to 0, by L' Hospital's rule, one has $\lim_{y \to 0} \frac{\mathcal{I}_{x'}[V](y\xi(x'))}{y} = V(x,0)\xi(x)$, which means at $y = 0$ the interpolated Bellman operator $\mathbf{T}_I$ is equivalent to the original Bellman operator, i.e, $\mathbf{T}_I[V](x,0) = \min_{a \in \mathcal{A}} \big[C(x,a) + \gamma \max_{x' \in \mathcal{X}: P(x'|x,a) > 0} V(x',0) \big]$.

Similar with the EVaR Bellman operator, we can show that the interpolated EVaR Bellman operator has the following useful properties: (1) monotonicity; (2) transition invariance; (3) contraction; and (4) concavity preserving in y. Property 3 helps us to construct the value iteration process with linear interpolation and ensures the existence of the unique fixed-point solution. Property 4 indicates the computational tractability of the inner maximization problem in (4.10). Moreover, property 4 will be used in bounding the error of our approximate algorithm. Details of the proofs of these properties are omitted as they are very similar to the corresponding proofs for the EVaR Bellman operator. Combining with Theorem 1 and these properties, we can design an approximate version of Algorithm 1.

Since the EVaR bellman operator has the concavity preserving property, Theorem 7 in [24] can be used to bound the error between EVaR value iteration and approximate EVaR value iteration. In particular, suppose that Assumption 7 is satisfied and $\epsilon > 0$ is an error tolerance parameter. For

Algorithm 2.2 EVaR Value Iteration with Linear Interpolation

1: **Initialization:** choose the set of interpolation points $Y(x)$ and the initial value function $V_0(x, y)$ satisfying Assumption 7.
2: **for** $t = 1, 2, \ldots$ **do**
3: **for** each $x \in \mathcal{X}$ and each $y_i \in Y(x)$ **do**
4: update the estimate of value function by

$$V_t(x, y_i) = \mathbf{T}_{\mathcal{I}}[V_{t-1}](x, y_i),$$

5: **end for**
6: **end for**
7: Get the near-optimal value function by $\hat{V}^*(x, y_i) = \lim_{t\to\infty} V_t(x, y_i)$.
8: Selecting the specific initial state x_0 and confidence level α, the solution of EVaR optimization problem with linear interpolation can be immediately obtained as $\hat{V}^*(x_0, \alpha)$.
9: Following Theorem 3, one can derived an optimal policy w.r.t $\hat{V}^*(x, y)$.

any state $x \in \mathcal{X}$ and step $t \geq 0$, choose $y_2 > 0$ such that $V_t(x, y_2) - V_t(x, 0) \geq -\epsilon$ and update the interpolation points according to: $y_{i+1} = \theta y_i, \forall i \geq 2$ with $\theta \geq 1$. Then following same steps as in Theorem 7 in [24], one can show that Algorithm 2 has the following error bound:

$$\frac{-\gamma}{1-\gamma} O\left((\theta - 1) + \epsilon\right) \leq \hat{V}^*(x_0, a) - \min_{\pi\in\Pi_H} \text{EVaR}_\alpha \left(\lim_{T\to\infty} J^\pi(x_0)\right) \leq 0$$

and the following finite time convergence error bound:

$$\left|\mathbf{T}_I^n[V_0](x_0, \alpha) - \min_{\pi\in\Pi_H} \text{EVaR} \left(\lim_{T\to\infty} J^\pi(x_0)\right)\right| \leq \frac{O((\theta - 1) + \epsilon)}{1-\gamma} + O(\gamma^n).$$

From these bounds, we know that when the number of interpolated points becomes large enough, i.e., $\theta \to 1$ and the tolerance parameter $\epsilon \to 0$, the error tends to 0.

2.2.3 Linear Interpolated EVaR with Sample Average Approximation

In Section 2.2.2, we assume that the transition probability of the underlying MDP model are known, which is often not the case in practice. Therefore, in this section, we propose a sample-based counterpart for Algorithm 2, which also approximates the solution of the primary

EVaR optimization problem in (2.1). In previous sections, we only define the value function. Now, without the model information, to obtain the policy, we need to define the state-action value function, state-action Bellman operator as well as the state-action interpolated Bellman operator for EVaR. Notice that we use the set of interpolation points $Y(x)$ rather than the whole continuous space $\mathcal{Y}$.

Definition 4. For any $x \in \mathcal{X}$, $y \in Y(x)$ and $a \in \mathcal{A}$, the state-action value function for EVaR MDP is defined as

$$Q^*(x, y, a) = \min_{\pi \in \Pi_H} \text{EVaR}_y\left(\lim_{T \to \infty} J^\pi(x, a) \right).$$

Definition 5. For any $x \in \mathcal{X}$, $y \in Y(x)$ and $a \in \mathcal{A}$, the state-action Bellman operator $\mathbf{F}$ is defined as

$$\mathbf{F}[Q](x, y, a) = C(x, a) + \gamma \max_{\xi \in \mathcal{U}_{\text{EVaR}}(y, P(\cdot|x,a))} \sum_{x' \in \mathcal{X}} \xi(x') V(y\xi(x')) P(x'|x, a),$$

where

$$V(x, y) = \min_{a \in \mathcal{A}} Q(x, y, a).$$

Definition 6. For any $x \in \mathcal{X}, y \in Y(x)$, the state-action interpolated Bellman operator is defined as

$$\mathbf{F}_{\mathcal{I}}[Q](x, y, a) = C(x, a) + \gamma \max_{\xi \in \mathcal{U}_{\text{EVaR}}(y, P(\cdot|x,a))} \sum_{x' \in \mathcal{X}} \frac{\mathcal{I}_{x'}[V](y\xi(x'))}{y} P(x'|x, a),$$

and the corresponding interpolated value iteration update:

$$Q(x, y, a) := C(x, a) + \gamma \max_{\xi \in \mathcal{U}_{\text{EVaR}}(y, P(\cdot|x,a))} \sum_{x' \in \mathcal{X}} \frac{\mathcal{I}_{x'}[V](y\xi(x'))}{y} P(x'|x, a). \tag{2.9}$$

Similar with the estimate of optimal value function $\hat{V}^*$, $\hat{Q}^*(x, y, a)$ denotes the unique solution

of $\mathbf{F}_{\mathcal{I}}[Q](x,y,a) = Q(x,y,a), \forall x \in \mathcal{X}, y \in Y(x), a \in \mathcal{A}$. According to the similar contraction argument, we can show the existence as well as the uniqueness of the fixed-point solution of $\mathbf{F}_I$. Without loss of generality, we assume that the set of EVaR-level interpolation points $\mathbf{Y}(x)$ is uniform at any state $x \in \mathcal{X}$. We consider synchronous setting where all the state-action value functions are updated at each time step.

When the transition probability P is unknown, we utilize the SAA approach introduced in [88] and [98] to estimate it. Let N_k denote the number of episodes and for each $(x,a) \in \mathcal{X} \times \mathcal{A}$, we run N_k episodes and then get the sampled transitions $\{x'^{,1}, \ldots, x'^{,N_k}\} \sim P(x'|x,a)$. Based on these samples, we can calculate the empirical transition probability $P_{N_k}(x'|x,a)$ by

$$P_{N_k}(x'|x,a) = \frac{1}{N_k} \sum_{i=1}^{N_k} \mathbf{1}\{x'^{,i} = x'|x,a\}, \forall x, x' \in \mathcal{X}, a \in \mathcal{A}, \tag{2.10}$$

and replace the inner maximization problem in (2.9) with the following one:

$$\max_{\xi \in \mathcal{U}_{\text{EVaR}}(y, P_{N_k}(\cdot|x,a))} \frac{1}{N_k} \sum_{i=1}^{N_k} \frac{\mathcal{I}_{x'^{,i}}[V_k](y\xi(x'^{,i}))}{y}.$$

As shown in [13], SAA is consistent, which means the solution of maximization problem equipped with SAA converges to the original solution as $N_k \to \infty$. The details of the consistency can be found in [88]. Now we can derive a sample-based EVaR algorithm as described in Algorithm 2.3.

In Algorithm 2.3, we first choose the set of interpolation points $Y(x)$ according to $y_{i+1} = \theta y_i, \forall i \geq 2$ with $\theta \geq 1$ and randomly assign values to the initial state-action value function $Q_0(x,y,a)$ for any $x \in \mathcal{X}, y \in \mathcal{Y}$ and $a \in \mathcal{A}(x)$, e.g., $Q_0(x,y,a) = 0$. Since the exact transition probability of the underlying model is unknown, we use Monte Carlo method to sample N_k trajectories for states $(x'^{,1}, \ldots, x'^{,N_k})$ and calculate the empirical transition probability $P_{N_k}(x'|x,a)$ by (2.10). In the iteration process, we update the state-action value function by equation (2.11) with step size satisfying (2.12) until the state-action value function converges. Lastly, a near-optimal policy can be constructed as a greedy policy with respect to the near-optimal value.In the following

Algorithm 2.3 Sample-based EVaR Algorithm

1: **Initialization:** choose the set of interpolation points $Y(x)$ and the initial state-action value function $Q_0(x, y, a) = 0$ for any $x \in \mathcal{X}$, $y \in Y(x)$ and $a \in \mathcal{A}(x)$.
2: Sample $N_k \geq 1$ for states $(x'^{,1}, \ldots, x'^{,N_k})$ and calculate the empirical transition probability $P_{N_k}(x'|x, a)$ by (2.10).
3: **for** $k = 1, 2, \ldots$ **do**
4: **for** each state x and action a **do**
5: update the state-action value function as follows:

$$Q_{k+1}(x, y, a) = Q_k(x, y, a) + \beta_k(x, y, a) \cdot \Bigg(- Q_k(x, y, a) + C(x, a) + \gamma \max_{\xi \in \mathcal{U}_{\text{EVaR}}(y, P_{N_k}(\cdot|x,a))} \frac{1}{N_k} \sum_{i=1}^{N_k} \frac{\mathcal{I}_{x'^{,i}}[V_k](y\xi(x'^{,i}))}{y} \Bigg). \tag{2.11}$$

where the value function is $V_k(x, y) = \min_{a \in \mathcal{A}} Q_k(x, y, a)$, and the step size $\beta_k(x, y, a)$ satisfies

$$\sum_k \beta_k(x, y, a) = \infty, \quad \sum_k \beta_k^2(x, y, a) < \infty. \tag{2.12}$$

6: **end for**
7: **end for**
8: A near-optimal policy can be constructed as

$$\tilde{\pi}^*(x, y) \in arg \min_{a \in \mathcal{A}} Q_{\bar{k}(x,y,a)}, \quad \forall x \in \mathcal{X}, \forall y \in \mathbf{Y} \tag{2.13}$$

where $\bar{k}$ is the iteration index when the learning is stopped.

theorem, we provide the convergence of Algorithm 3.

Theorem 4. Suppose the step size $\beta_k(x, y, a)$ follows the update rule in (2.12) and the sample size $N_k \to \infty$ as $k \to \infty$. Then recursively applying (2.11) makes $\{Q_k(x, y, a)\}_{k \in \mathbb{N}}$ converges to the fixed-point solution $\hat{Q}^*(x, y, a)$ component-wise with probability 1.

Proof. Please refer to Appendix A.4. □

2.3 EVaR Optimization with Policy Gradient: EVaR-PG

While value iteration offers a solid foundation for determining optimal strategies in many RL problems, it has limitations, particularly in complex or continuous action spaces. To overcome

these challenges, we turn to policy gradient methods, which directly optimize the policy rather than estimating values for each state-action pair. Policy gradient methods are well-suited for environments with stochastic or continuous actions, providing a more flexible and effective approach. In the following section, we introduce the policy gradient approach to solve the EVaR optimization problem, starting with preliminaries on new notations and a reformulated problem setup.

2.3.1 Preliminaries

In this section, the cost function $C(x, a)$ is assumed to be bounded within $[0, C_{\max}]$. For simplicity, we set the initial state distribution as $P_0 = \mathbf{1}\{x = x_0\}$.

In order to compare the performance of our approach with CVaR, we denote EVaR as

$$\text{EVaR}_\alpha(Z) = \inf_{\nu>0} \left\{ \nu^{-1} \ln \frac{M_Z(\nu)}{1-\alpha} \right\}. \tag{2.14}$$

Note that this form is still consistent with the one introduced in Section 1.2, as we are considering losses here.

Recall that EVaR is a coherent risk measure, optimizing over EVaR under the MDP model solve these uncertainties mention in Section 1.2.1.

Here we discuss more about boundedness of the optimal value of the parameter ν indexed in the definition of EVaR.

Remark 2. For generalization, we here assume that Z is bounded, i.e., $Z \in [Z_{\min}, Z_{\max}]$. From [2], we know that $\mathbb{E}[Z] \leq \text{EVaR}_\alpha(Z) \leq \text{esssup}(Z)$ [2], which means $\mathbb{E}[Z] \leq \inf_{\nu>0}\{\nu^{-1} \ln \frac{\mathbb{E}[e^{Z\nu}]}{1-\alpha}\} \leq \text{esssup}(Z)$. Let ν^* be the corresponding optimal value to get infimum of EVaR, then $\mathbb{E}[Z] \leq \nu^{*-1} \ln \frac{\mathbb{E}[e^{Z\nu^*}]}{1-\alpha} \leq \text{esssup}(Z)$. From this, we know ν^* is in the range $[-\frac{1-\alpha}{Z_{\max}-Z_{\min}}, +\infty]$. Let $V_{\min} = -\frac{(1-\gamma)\ln(1-\alpha)}{C_{\max}-C_{\min}}$, then ν^* is lower bounded by $V_{\min}$.

From Remark 2, ν is lower bounded by $V_{\min}$. In order to ensure that ν is always bounded in the gradient descent process, we make an assumption about its upper boundedness as follows.

Assumption 2. ν is upper bounded, i.e., $\nu \leq V_{\max}$.

2.3.2 Problem Statement

Our goal is to find an optimal policy which is parameterized by θ that solves the following optimization problem for a given confidence level $\alpha \in (0, 1)$:

$$\min_{\theta} \text{EVaR}_{\alpha}(J^{\theta}(x_0)), \tag{2.15}$$

which can be reformulated as:

$$\min_{\theta,\nu} L(\nu, \theta) := \nu^{-1} \ln \frac{\mathbb{E}[e^{J^{\theta}(x_0)\nu}]}{1-\alpha}. \tag{2.16}$$

In the following, we make assumption about the stationary policy π.

Assumption 3. For any (x, a), $\pi(a|x, \theta)$ is continuously differentiable in Θ and $\nabla_{\theta}\pi(a|x, \theta)$ is a Lipschitz function in θ for every $(x, a) \in \mathcal{X} \times \mathcal{A}$. Moreover, the ratio $\nabla_{\theta}\pi(a|x, \theta)/\pi(a|x, \theta)$ is bounded for all $\theta \in \mathbb{R}^{\kappa}$ and every $(x, a) \in \mathcal{X} \times \mathcal{A}$ (this is also assumed in [9]).

One example satisfying the assumption is

$$\pi(a = j|x = i, \theta) = \frac{e^{\theta_{ij}}}{\sum_{j=1}^{m} e^{\theta_{ij}}}$$

where $\theta = [\theta_{11}, \cdots, \theta_{1m}, \cdots, \theta_{nm}] \in \mathbb{R}^{nm}$. Let $\pi_{ij}(\theta)$ denote $\pi(a = j|x = i, \theta)$, then

$$\frac{\partial \pi_{ij}(\theta)/\partial \theta_{ij}}{\pi_{ij}(\theta)} = 1 - \pi_{ij}(\theta), \quad \frac{\partial \pi_{ij}(\theta)/\partial \theta_{kl}}{\pi_{ij}(\theta)} = -\pi_{kl}(\theta)$$

where $k = 1, \cdots, n$ with $k \neq i$ and $l = 1, \cdots, m$ with $l \neq j$.

2.3.3 A Trajectory-Based EVaR Policy Gradient Algorithm

In this section, we propose a trajectory-based EVaR policy gradient algorithm, which is to descend in ν and θ according to the gradients of $L(\nu, \theta)$ w.r.t ν and θ, i.e.,

$$\nabla_\nu L(\nu, \theta) = \nu^{-1} \frac{\nabla_\nu \mathbb{E}[e^{J^\theta(x_0)\nu}]}{\mathbb{E}[e^{J^\theta(x_0)\nu}]} - \nu^{-2} \ln \frac{\mathbb{E}[e^{J^\theta(x_0)\nu}]}{1-\alpha} \tag{2.17}$$

and

$$\nabla_\theta L(\nu, \theta) = \nu^{-1} \frac{\nabla_\theta \mathbb{E}[e^{J^\theta(x_0)\nu}]}{\mathbb{E}[e^{J^\theta(x_0)\nu}]}. \tag{2.18}$$

In our trajectory-based algorithm, at each iteration, the algorithm will generate N trajectories by following the current policy π, then use these trajectories to estimate the gradients in (B.1) and (B.3) and update these parameters.

To generate these gradients, let $\xi = \{x_0, a_0, \ldots, x_{T-1}, a_{T-1}, x_T\}$ denote one trajectory where $x_T = x_{\text{tar}}$ and the corresponding cost function is $J(\xi) = \sum_{k=0}^{T-1} \gamma^k C(x_k, a_k)$. The probability of generating such a trajectory is $\mathbb{P}_\theta(\xi) = P_0(x_0) \prod_{k=0}^{T-1} \pi(a_k|x_k, \theta) P(x_{k+1}|x_k, a_k)$ and we can also have $\nabla_\theta \log \mathbb{P}_\theta(\xi) = \sum_{k=0}^{T-1} \nabla_\theta \log \pi(a_k|x_k, \theta) = \sum_{k=0}^{T-1} \nabla_\theta \pi(a_k|x_k, \theta)/\pi(a_k|x_k, \theta)$ whenever $\mathbb{P}_\theta(\xi) \neq 0$ and $\pi(a_k|x_k, \theta) \in (0, 1]$.

Proposition 1. By the definition of $\mathbb{P}_\theta(\xi)$ and $\nabla_\theta \log \mathbb{P}_\theta(\xi)$, we have

$$\begin{aligned}
\mathbb{P}_\theta(\xi) \nabla_\theta \log \mathbb{P}_\theta(\xi) &= P_0(x_0) \prod_{k=0}^{T-1} \pi(a_k|x_k, \theta) P(x_{k+1}|x_k, a_k) \sum_{k=0}^{T-1} \nabla_\theta \pi(a_k|x_k, \theta)/\pi(a_k|x_k, \theta) \\
&= P_0(x_0) \sum_{k=0}^{T-1} \prod_{i \neq k}^{T-1} \nabla_\theta \pi(a_k|x_k, \theta) \pi(a_k|x_k, \theta) P(x_{k+1}|x_k, a_k).
\end{aligned}$$

Combining Assumption 3 and the fact that the sum of products of Lipschitz function is Lipschitz, we can show that $\mathbb{P}_\theta(\xi) \nabla_\theta \log \mathbb{P}_\theta(\xi)$ and $\sum_\xi \mathbb{P}_\theta(\xi) \nabla_\theta \log \mathbb{P}_\theta(\xi)$ are Lipschitz in θ. Furthermore, since the gradient of Lipschitz function is bounded, we have $|\nabla_\theta (\sum_\xi \mathbb{P}_\theta(\xi) \nabla_\theta \log \mathbb{P}_\theta(\xi))| \leq K_1(\xi)$. Also, $\mathbb{E}[\nabla_\theta \log \mathbb{P}_\theta(\xi)] = \sum_\xi \mathbb{P}_\theta(\xi) \nabla_\theta \log \mathbb{P}_\theta(\xi) = 0$.

Proposition 2. By Assumption 3, $\nabla_\theta \log \mathbb{P}_\theta(\xi)$ is bounded, i.e., $|\nabla_\theta \log \mathbb{P}_\theta(\xi)| \leq K_2(\xi)$.

Since we have the representation of trajectories, we can now derive the estimated form of these gradients. First, the goal function $L(\nu, \theta)$ in the optimization problem (2.16) can be rewrite as

$$L(\nu, \theta) = \nu^{-1} \ln \frac{\sum_\xi \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}}{1-\alpha}.$$

Now, based on this function, we can estimate these gradients as follows.

Gradient w.r.t θ is

$$\nabla_\theta L(\nu, \theta) = \nu^{-1} \frac{\sum_\xi \nabla_\theta \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}}{\sum_\xi \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}}.$$

Note that $\nabla_\theta \mathbb{P}_\theta(\xi) = \mathbb{P}_\theta(\xi) \nabla_\theta \log \mathbb{P}_\theta(\xi)$ and insert this term to the above equation, we have

$$\nabla_\theta L(\nu, \theta) = \nu^{-1} \frac{\sum_\xi \mathbb{P}_\theta(\xi) \nabla_\theta \log \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}}{\sum_\xi \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}}. \tag{2.19}$$

Gradient w.r.t ν is

$$\nabla_\nu L(\nu, \theta) = -\nu^{-2} \ln \frac{\sum_\xi \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}}{1-\alpha} + \nu^{-1} \frac{\sum_\xi \mathbb{P}_\theta(\xi) J(\xi) e^{J(\xi)\nu}}{\sum_\xi \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}}. \tag{2.20}$$

Recall that in each iteration, we generate N trajectories. Therefore, use $\xi_{j,k}$ denote the j-th trajectory in k-th iteration. Then these trajectories can be used to estimate the gradients in (2.19) (2.20) and these updates rules can be written as:

ν-update

$$\begin{aligned} \nu_{k+1} &= \Gamma_{\mathcal{N}}[\nu_k - \zeta_2(k) \nabla_\nu L(\nu, \theta)|_{\nu=\nu_k, \theta=\theta_k}] \\ &= \Gamma_{\mathcal{N}} \left[\nu_k - \zeta_2(k) \left(-\nu_k^{-2} \ln \frac{\sum_{j=1}^N e^{J(\xi_{j,k})\nu_k}}{N(1-\alpha)} + \nu_k^{-1} \frac{\sum_{j=1}^N J(\xi_{j,k}) e^{J(\xi_{j,k})\nu_k}}{\sum_{j=1}^N e^{J(\xi_{j,k})\nu_k}} \right) \right]. \end{aligned} \tag{2.21}$$

θ-update

$$\begin{aligned}\theta_{k+1} &= \Gamma_\Theta[\theta_k - \zeta_1(k)\nabla_\theta L(\nu,\theta)|_{\nu=\nu_k,\theta=\theta_k}] \\ &= \Gamma_\Theta\left[\theta_k - \zeta_1(k)\left(\nu_k^{-1}\frac{\sum_{j=1}^N \nabla_\theta \log \mathbb{P}_\theta(\xi_{j,k})e^{J(\xi_{j,k})\nu_k}}{\sum_{j=1}^N e^{J(\xi_{j,k})\nu_k}}\right)\right].\end{aligned} \tag{2.22}$$

where

$$\Gamma_{\mathcal{N}}(\nu) = \text{argmin}_{\hat{\nu}\in[V_{\min},V_{\max}]}||\nu-\hat{\nu}||_2^2,$$

and

$$\Gamma_\Theta(\theta) = \text{argmin}_{\hat{\theta}\in\Theta}||\theta-\hat{\theta}||_2^2.$$

Note that these projections ensure the updated values are still in the bounded ranges, which are further used in the proof of the convergence. Algorithm 2.4 contains the pseudo-code of our proposed EVaR policy gradient algorithm.

Algorithm 2.4 Trajectory-based EVaR Policy Gradient

Input: confidence level α and parameterized policy $\pi(\cdot|\cdot,\theta)$.
Initialization: choose $\nu = \nu_0$, $\theta = \theta_0$ and initial state $x_0 = x_0$.
while TRUE **do**
  for $k = 0, 1, 2, \ldots$ **do**
    Generate N trajectories $\{\xi_{j,k}\}_{j=1}^N$ from x_0 by following the current policy parameterized by θ_k. Update (ν,θ) by

$$\nu_{k+1} = \Gamma_{\mathcal{N}}\left[\nu_k - \zeta_2(k)\left(-\nu_k^{-2}\ln\frac{\sum_{j=1}^N e^{J(\xi_{j,k})\nu_k}}{N(1-\alpha)} + \nu_k^{-1}\frac{\sum_{j=1}^N J(\xi_{j,k})e^{J(\xi_{j,k})\nu_k}}{\sum_{j=1}^N e^{J(\xi_{j,k})\nu_k}}\right)\right]$$

$$\theta_{k+1} = \Gamma_\Theta\left[\theta_k - \zeta_1(k)\left(\nu_k^{-1}\frac{\sum_{j=1}^N \nabla_\theta \log \mathbb{P}_\theta(\xi_{j,k})e^{J(\xi_{j,k})\nu_k}}{\sum_{j=1}^N e^{J(\xi_{j,k})\nu_k}}\right)\right]$$

  end for
  if $|\nu_k - V_{\max}| \leq \epsilon$ for some tolerance parameter $\epsilon > 0$ **then**
    Set $V_{\max} \leftarrow 2V_{\max}$.
  else
    return ν and θ, **break**
  end if
end while

We further make a typical assumption about the step sizes for policy gradient.

Assumption 4. The step size schedules $\zeta_1(k)$ and $\zeta_2(k)$ satisfy

$$\sum_k \zeta_1(k) = \sum_k \zeta_2(k) = \infty, \tag{2.23}$$

$$\sum_k \zeta_1^2(k), \sum_k \zeta_2^2(k) < \infty, \tag{2.24}$$

$$\zeta_1(k) = o(\zeta_2(k)), \;\; i.e., \;\; \lim_{k\to\infty} \frac{\zeta_1(k)}{\zeta_2(k)} = 0 \tag{2.25}$$

From (2.25), we know that ν updates at a faster timescale $\zeta_2(k)$ and θ updates at a slower timescale $\zeta_1(k)$. Note that the above assumption satisfies the standard condition of stochastic approximation algorithms.

In the following theorem, we prove our trajectory-based EVaR policy gradient algorithm converges to a local optimal policy for the EVaR optimization problem (2.16).

Theorem 5. Under Assumption 3 and Assumption 4, the policy sequence generated by Algorithm (2.4) converges almost surely to a locally optimal policy θ^* for the EVaR optimization problem as $k \to \infty$.

Proof. Please refer to Appendix A.5 for details. □

Here we give a high level overview of the proof technique in the following. First, we regard these updates (ν_k, θ_k) as a multi-time scale discrete stochastic approximation and show the sequences (ν_k, θ_k) converge to the solution of the corresponding continuous time systems with different speed. Then by using Lyapunov analysis, we show that the these sequences further converges to the local asymptotically stable points (ν^*, θ^*). Lastly, we show that (ν^*, θ^*) is a local minimum.

To illustrate this high level idea more, consider the following two-time scale stochastic approximation algorithm for updating (x_i, y_i):

$$x_{n+1} = x_n + \zeta_1(n)[h(x_n, y_n) + M_{n+1}^{(1)}], \tag{2.26}$$

$$y_{n+1} = y_n + \zeta_2[g(x_n, y_n) + M_{n+1}^{(2)}], \tag{2.27}$$

where h and g are two Lipschitz continuous functions, $M_{n+1}^{(1)}$, $M_{n+1}^{(2)}$ are two Martingale differences w.r.t the increasingσ-field $\mathcal{F}_n = \sigma(x_m, y_m, M_m^{(1)}, M_m^{(2)}, m \leq n),\ n \geq 0$, satisfying $\mathbb{E}[||M_{n+1}^{(i)}||^2|\mathcal{F}_n] \leq K(1 + ||x_n|| + ||y_n||)^2,\ i = 1, 2$ for $n \geq 0$. The step sizes $a(n)$ and $b(n)$ are non-summable and square summable. If $b(n)$ converges to zero faster than $a(n)$, then (2.26) is a faster recursion than (2.27) after some iteration n_0, which implies that (2.26) has uniformly larger increments than (2.27). Note that (2.27) can be rewritten as

$$y_{n+1} = y_n + \zeta_1 \frac{\zeta_2}{\zeta_1}[g(x_n, y_n) + M_{n+1}^{(2)}],$$

and by the fact that ζ_2 converges to zero faster than ζ_1, (2.26) and (2.27), it's instructive to consider the ODE $\dot{x} = h(x, y)$ and $\dot{y} = 0$. By Theorem 2 in Chapter 6 of [19], we can show that (x_n, y_n) converges to $(\lambda(y^*), y^*)$ as $n \to \infty$ almost surely, where $\lambda(y^*)$ is a globally asymptotically stable equilibrium of the ODE $\dot{x} = h(x, y)$ and λ is a Lipschitz continuous function, and y^* is a globally asymptotically equilibrium of the ODE $\dot{y} = g(\lambda(y), y)$.

2.4 Experiments

We provide some numerical examples to illustrate the algorithms developed in this chapter.

2.4.1 EVaR-VI

In the first experiment, we set the environment to be a rectangular grid world, where the state space is consisted of positions in the map. An agent starts at a safe position (i.e., the initial state) and its goal is to travel to a given destination. In each step, there are four available actions to take: left, right, up and down. After taking an action, the agent will move to the corresponding neighboring state with probability $1 - \delta$ while the agent will move to any of the other three neighboring states with equal probability $\delta/3$. In the grid world, there are some obstacles which differ from safe

positions in the following setup. The cost of each movement between safe regions is 1 while the cost of hitting an obstacle is 40. Also, the mission will be terminated if the agent hits obstacles. The goal here is to find a safe path with small cost.

In order to compare with the CVaR application in risk-sensitive decision making in [24], we use the same parameters for the grid world setup. We use a 64×53 grid world and put 80 obstacles (printed in bright yellow), which results in a total of $3,312$ states. The start point is $(60, 50)$ and the destination is $(60, 2)$. For the confidence level set, we choose the number of interpolated points be 21. In order to make the error smaller, here we use the update rule mentioned in the bounds, i.e., $y_{i+1} = \theta y_i$ for $i = 2, 3, \ldots, 20$. We choose $\delta = 0.05$ and a discount factor $\gamma = 0.95$ for an effective horizon of 200 steps [24]. For the initialization, we apply the standard value iteration process, i.e., use the risk-neutral method. In the EVaR value iteration, we use an optimization tool named Gurobi [17, 78]. Furthermore, considering the cases where the transition probability is unknown, we also validate the algorithm equipped with SAA (Algorithm 2.3) in the same setup. Note that the choice of N_k affects the accuracy of the approximation of the transition probability, thus further has influence on the near-optimal value function as well as the optimal policy. Here we choose the sample size $N_k = 100$, $N_k = 500$ and $N_k = 1000$ to compare the influence.

After applying Algorithm 2.2 and Algorithm 2.3 (with three different value of N_k), we plot the near-optimal value function and the corresponding optimal path at $\alpha = 0.01$, $\alpha = 0.11$ and $\alpha = 1.00$ in Figures 2.1, 2.2 and 2.3 respectively, to compare the agent's preference about risk. In the figures, we use bright yellow color to mark the positions of the obstacles, and use color bar to represent the value functions for different states. More specifically, as shown in the color bar, the bluer the color, the smaller the value function. From the figures, we can see that the closer the states are to the obstacles, the higher the cost are. Comparing the results generated by applying Algorithm 2.2 in Figures 2.1, 2.2 and 2.3, we can find that, with confidence level α increasing, the difference between the value function of safe states is getting smaller, i.e, the states near obstacles are becoming less risky, which leads to the case that the agent's strategy becomes more aggressive, i.e., the optimal path tends to be shorter and closer to the obstacles. For this

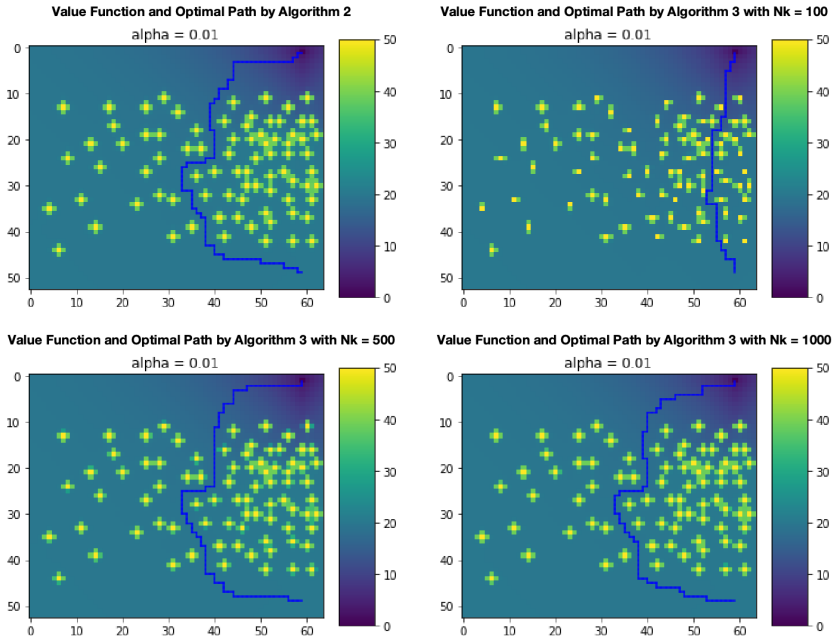


Figure 2.1: The value function and corresponding optimal path for $\alpha = 0.01$ generated by Algorithm 2.2 and Algorithm 2.3 (with different values of N_k) in the obstacle's setting.

part, we also reproduce the CVaR algorithm in [24] and the results are almost same with ours, which indicates that our approach is also practical in solving risk-sensitive RL. As for the results generated by Algorithm 2.3, when $N_k = 100$, the value function and the path are not near-optimal since the estimated transition probability is not accurate enough. But for $N_k = 500$ and $N_k = 1000$, the overall tendency is almost the same as the one in Algorithm 2.2 despite some minor difference that can be further alleviated by choosing larger N_k.

In the second experiment, we apply both Algorithm 2.2 and Algorithm 2.3 in Cliffwalk's setup. In this setting, we choose the map to be 14×16 and put 23 cliffs, which leads to a total of 201 states. The difference between cliff and obstacle in the first example is that hitting cliff will send the agent back to the start point while hitting an obstacle in the first example ends the mission. Similar to the first example, we use bright yellow color to mark the positions of the cliffs, and use color bar to represent the value functions for different states. As shown in Figures 2.4, 2.5, 2.6, we know that for both algorithms, with the confidence level increasing, the agent becomes more and more aggressive and the optimal path becomes shorter and closer to the cliffs. This tendency

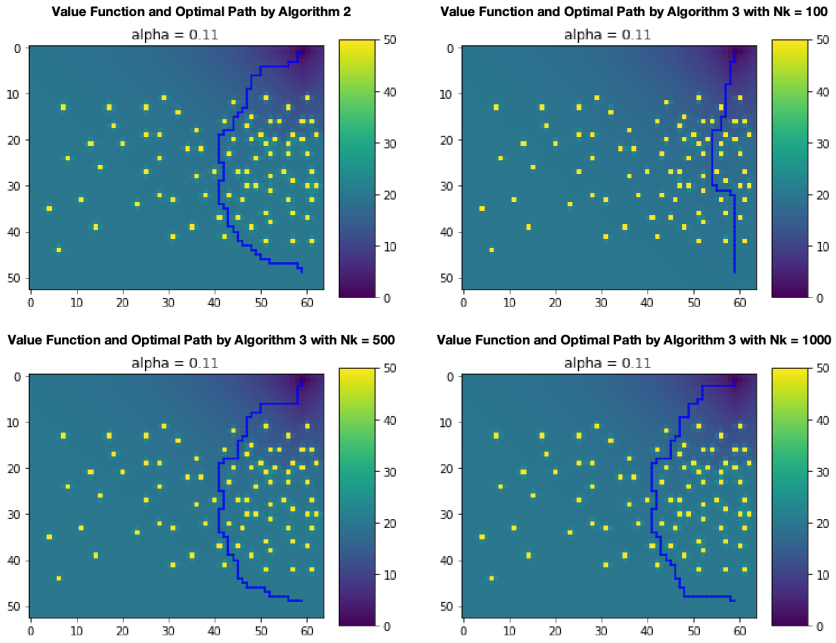


Figure 2.2: The value function and corresponding optimal path for $\alpha = 0.11$ generated by Algorithm 2.2 and Algorithm 2.3 (with different values of N_k) in the obstacle's setting.

is exactly the same as the one we get in the first experiment. Moreover, for the results generated by Algorithm 2.3, when $N_k = 100$, all these optimal policies generated by Algorithm 2.3 are quite different with these in Algorithm 2.2. For $N_k = 500$ and $N_k = 1000$, the optimal path is same when $\alpha = 0.11$ and $\alpha = 1.00$ while the optimal path is a little different when $\alpha = 0.01$.

2.4.2 EVaR-PG

In this section, we apply Algorithm 2.4 to RL following the similar setup with [23] to illustrate the practicality and efficiency of our approach.

We consider an optimal stopping problem of purchasing certain types of goods. Under this setup, the state at each time step $k \leq T$ is $x = (c_k, k)$, where c_k is the purchase cost and T is the upper bound of first-hitting time. The purchase cost sequence $\{c_k\}_{k=0}^{T}$ is randomly generated by a Markov chain with two modes. Specifically, at time k the random purchase cost at the next time step c_{k+1} either grows by a constant factor $f_u > 1$, i.e., $c_{k+1} = f_u c_k$ with probability p or drops by a constant factor $f_d < 1$, i.e., $c_{k+1} = f_d c_k$ with probability $1 - p$. The agent should decide either

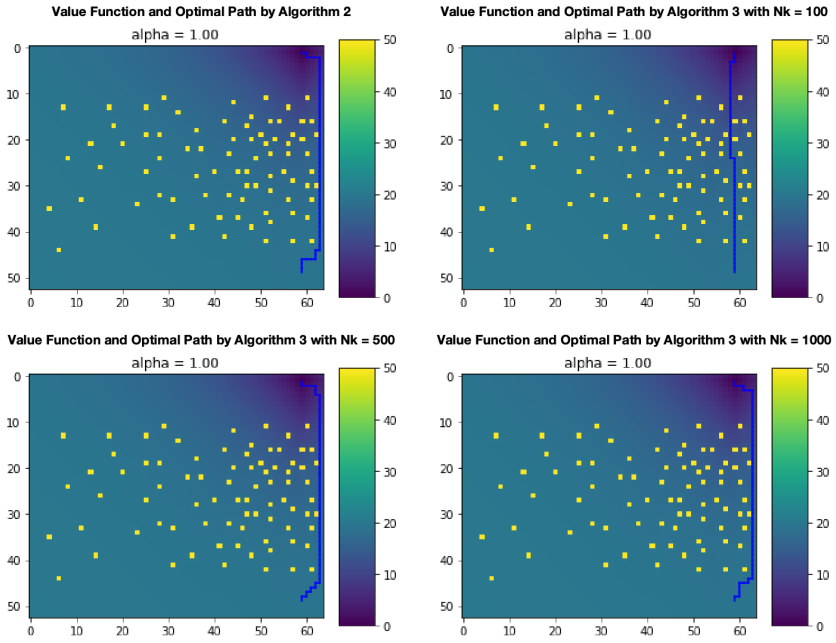


Figure 2.3: The value function and corresponding optimal path for $\alpha = 1.00$ generated by Algorithm 2.2 and Algorithm 2.3 (with different values of N_k) in the obstacle's setting.

to accept the present cost ($u_k = 1$) or wait ($u_k = 0$). If the agent accepts the cost or the system terminate at time $k = T$, the purchase cost is set at $\max(K, c_k)$, where K is the maximum cost threshold. Moreover, due to the steady rate of inflation, at each time step the agent will receive an extra cost of p_h, which is independent to the purchase cost. Also, account for the increase in the agent's affordability, there is a discount factor $\gamma \in (0, 1)$.

Therefore, the optimal stopping problem can be formulated as

$$\min_{\theta} \text{EVaR}_{\alpha}(J^{\theta}(x_0)) \tag{2.28}$$

where

$$J^{\theta}(x) = \sum_{k=0}^{T} \gamma^k \big(\mathbf{1}\{u_k = 1\} \max(K, c_k) + \mathbf{1}\{u_k = 0\} p_h\big) | x_0 = x, \pi.$$

Here we choose $x_0 = [1; 0]$ (this means $c_0 = 1$), $p_h = 0.1$, $T = 20$, $K = 5$, $\gamma = 0.95$, $f_u = 2$, $f_d = 0.5$ and $p = 0.65$. The number of trajectories $N = 500,000$ and $\Theta = [-20, 20]^{\kappa_1}$, where the dimension of the basis function is $\kappa_1 = 64$. We implement radial basis functions (RBFs) as feature

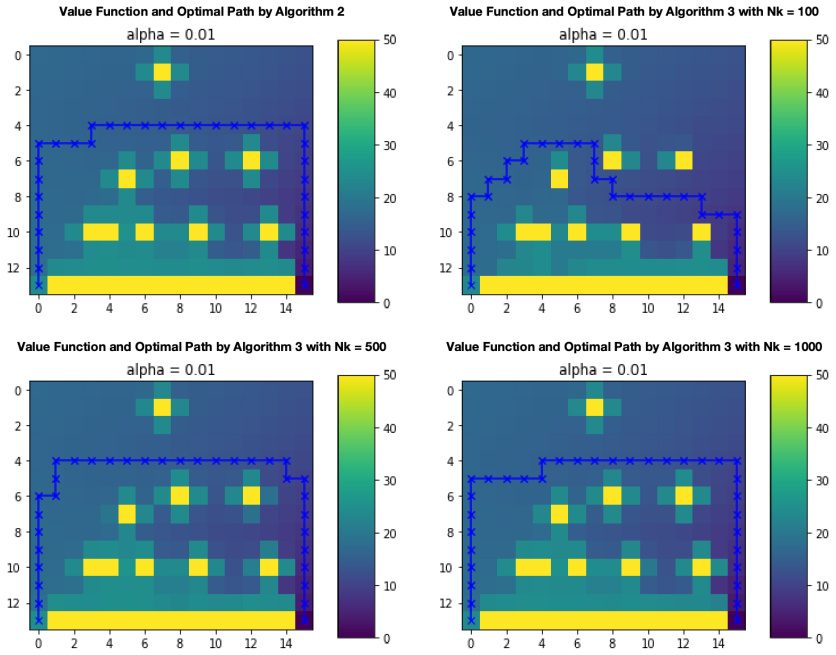


Figure 2.4: The value function and corresponding optimal path for $\alpha = 0.01$ generated by Algorithm 2.2 and Algorithm 2.3 (with different values of N_k) in the cliff's setting.

functions and search over the class of Boltzmann policies

$$\left\{\theta : \{\theta_{x,a}\}_{x\in\mathcal{X},a\in\mathcal{A}},\ \pi_\theta(a|x) = \frac{\exp(\theta_{x,a}^{\mathrm{T}} x_f(x))}{\sum_{a\in\mathcal{A}} \exp(\theta_{x,a}^{\mathrm{T}} x_f(x))}\right\},$$

where $x_f(x)$ is the feature chosen by RBF at each state x.

There are two phases in applying Algorithm 2.4 to this experiment:

- **Tunning phase**: We run the EVaR policy gradient algorithm and update the policy until (ν, θ) converges.
- **Converged run**: Having obtained a converged policy θ^* in the tunning phase, in the converged run phase, we perform a Monte Carlo simulation of $10,000$ trajectories and report the results as averages over these trials.

In order to better compare the results generated by applying the EVaR policy gradient algorithm to the optimal stopping problem, we implement Algorithm 2.4 at two different confidence level,

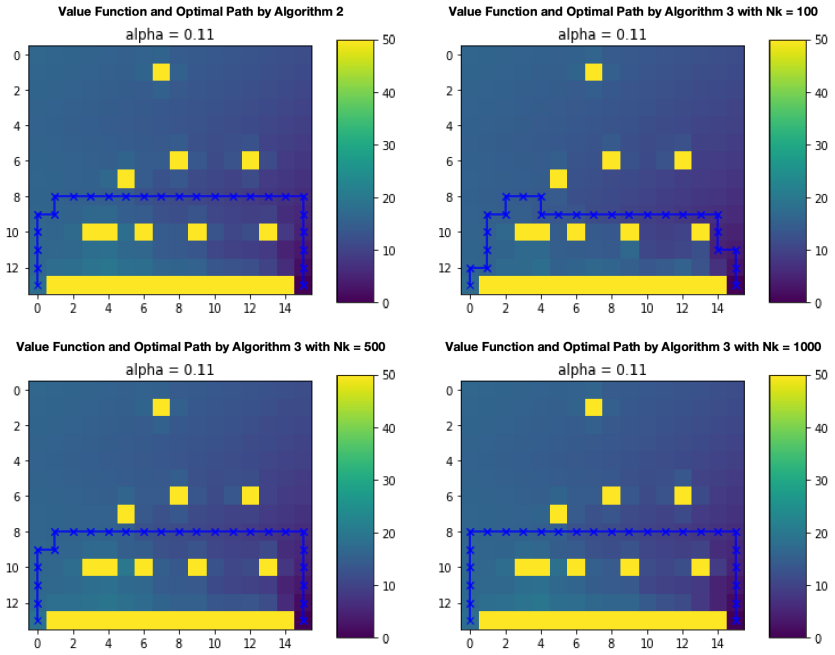


Figure 2.5: The value function and corresponding optimal path for $\alpha = 0.11$ generated by Algorithm 2.2 and Algorithm 2.3 (with different values of N_k) in the cliff's setting.

$\alpha = 0.05$ (agent will be more risk-averse) and $\alpha = 0.95$ (agent will be more risk-seeking). The results for $\alpha = 0.05$ is shown in Figure 2.7a and the result for $\alpha = 0.95$ is shown in Figure 2.7b. From these figures, we can see that the agent is more likely to wait longer and its tolerance towards larger cost is higher with confidence level $\alpha = 0.95$ while the agent prefers to accept the cost at earlier state to avoid larger cost at next state.

2.5 Conclusion

In this chapter, we have applied EVaR to risk-sensitive RL, introducing both value iteration and policy gradient methods based on the MDP framework. We proposed an EVaR value iteration algorithm and a more practical approximate version, proving convergence and bounding the approximation error. Additionally, for scenarios where the transition kernel of the MDP is unknown, we presented a sample-based EVaR synchronous Q-value update algorithm with convergence guarantees. We validated these approaches in simulation experiments, demonstrating

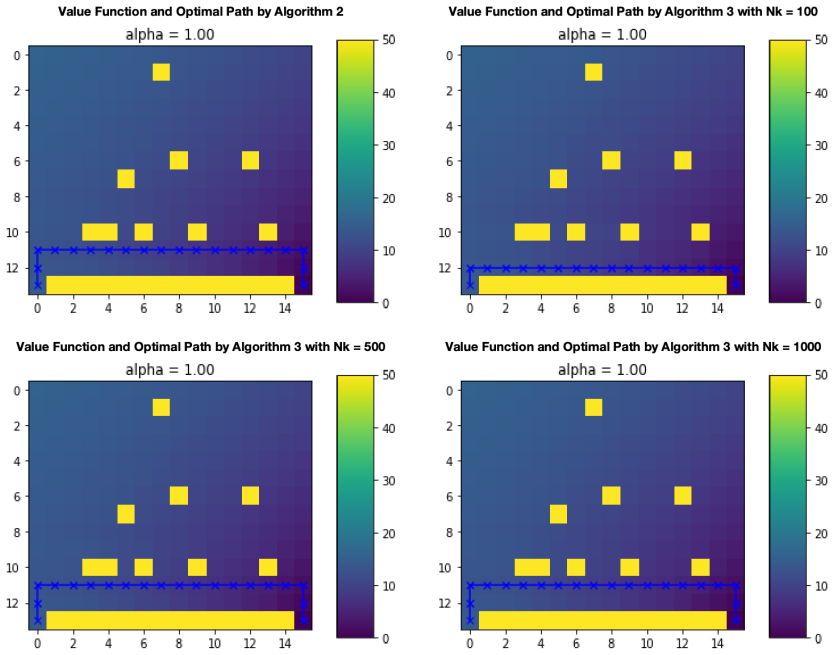
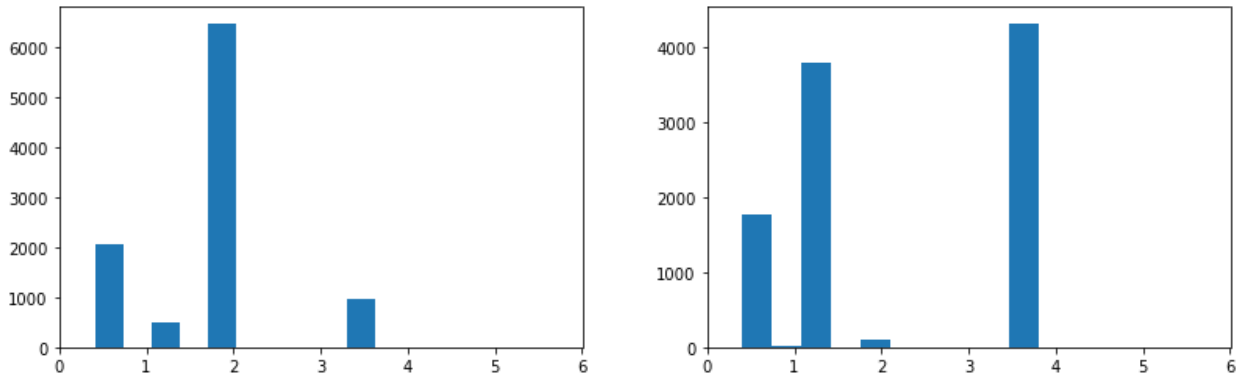


Figure 2.6: The value function and corresponding optimal path for $\alpha = 1.00$ generated by Algorithm 2.2 and Algorithm 2.3 (with different values of N_k) in the cliff's setting.

the effectiveness of our algorithms. Furthermore, we developed an EVaR policy gradient algorithm that learns a parameterized policy, providing a convergence guarantee to a locally optimal policy. We evaluated this approach through simulations using an optimal stopping problem setup, confirming its practical applicability.



(a) $\alpha = 0.05$. (b) $\alpha = 0.95$.

Figure 2.7: The total discounted cost distributions generated by applying the EVaR Policy gradient algorithm at different confidence levels.

Chapter 3

Risk-Sensitive Reinforcement Learning with ϕ-Divergence Risk Measure

3.1 Introduction

As discussed in Chapter 1, a multitude of risk measures have been studied in the literature and successfully applied to RL, such as VaR, CVaR, Entropic risk measure and EVaR et al [22–24, 33, 43, 44, 55, 70, 74, 95, 97]. The extensive exploration of risk measures in decision-making contexts often requires adopting specific algorithms tailored to each measure, potentially reducing decision-making efficiency. While some risk measures incorporate a risk-tolerance parameter that reflects decision-makers' preferences to an extent, their varied methodologies might not capture these preferences accurately due to different risk quantification approaches. [95] introduces a policy gradient method applicable to a wide range of coherent risk measures. However, this method assumes a structured form of the measures' envelope sets in their dual representation. Although the approach is comprehensive, it involves significant computation complexity, especially in identifying saddle points across four parameters. This complexity, arising from the specific constraints within the dual representation, represents a trade-off between generality and computational efficiency in risk-sensitive RL. Realizing these challenges in existing

work on risk sensitive RL, we are prompted to explore a critical question:

Is it possible to develop a class of coherent risk measures that cover popular risk measures and to design an accompanying algorithm that not only offers decision-makers greater flexibility in selecting risk measures but also ensures efficiency and robustness?

To address this question, we adopt a new class of risk measures named PhiD-R, whose uncertainty sets in their dual representations are defined by using ϕ-divergence [2], to RL problems. Our choice of this class of risk measures is motivated by several factors: 1) PhiD-R is coherent and includes many widely adopted risk measures such as CVaR and EVaR as special cases [2]. 2) ϕ-divergence has been thoroughly explored in the machine learning domain, particularly in policy optimization and robust RL [11,41,46,49,54,87]. This extensive research supports the potential of ϕ-divergence to foster innovative developments in risk measures for risk-sensitive RL. 3) Previous work by [24] illustrated that solving CVaR RL was equivalent to tackling risk-neutral RL when uncertainties in transition probabilities are defined by specific divergence measures. This finding motivates further exploration into the equivalence of PhiD-R RL and robust RL, aiming to address the robustness concerns identified. 4) The explicit generalized representation of PhiD-R in [2] allows for the development of a generalized policy gradient method applicable to all forms of ϕ-divergence, which ensures both flexibility and efficiency of the approach.

Chapter Contribution: In this chapter, we introduce a trajectory-based policy gradient method tailored to solve RL problems under this new class of risk measures, PhiD-R. The explicit representation of PhiD-R allows for efficient gradient estimation. Based on these gradient estimates, we propose specific update rule for each parameter. By using multi-time stochastic approximation technique [19, 23, 70], we demonstrate that our proposed method asymptotically converges to locally optimal policies. This approach is highly versatile, applying to the entire spectrum of ϕ-divergence, thereby broadening the scope beyond traditional risk measures such as CVaR. This extension also offers a new approach to address CVaR RL and explores novel approaches within risk-sensitive RL. Our approach benefits from the coherence property of PhiD-R and the dual presentation theorem, which ensures that solving PhiD-R RL is equivalent

to solving robust RL when uncertainties in transition probabilities are defined by ϕ-divergence. This connection between risk and robustness is particular valuable when decision-makers face scenarios with inherent uncertainty and wish to incorporate their risk preferences.

Related Work: Several works are closely related to our studies. [95] proposes a generalized method for solving RL problems with coherent risk measures, which aligns with PhiD-R, and demonstrates convergence to local optimality. Our approach also guarantees near-optimality, while providing a simplified solution for PhiD-R, requiring fewer assumptions and optimized parameters. We build on the well-established representation of PhiD-R from [2], offering a more efficient and practical method tailored to these risk measures. In particular, when applied to CVaR RL, our algorithm reduces the number of parameters without compromising local optimality. Compared to policy gradient-based CVaR RL approaches that extend the likelihood-ratio method for demonstrating local optimality [95, 97], our work estimates gradients directly using the explicit representation of PhiD-R. While our methodology and objectives differ from those of [95, 97], all algorithms achieve convergence to a locally optimal policy. Furthermore, our approach contrasts with existing policy gradient research on CVaR [23, 83, 104], which is typically limited to the constrained RL framework. Our method also diverges from [70], which focuses solely on EVaR.

Chapter Organization: The remainder of this chapter is organized as follows. Section 3.2 provides background on risk measures, ϕ-divergence and the new risk measure class PhiD-R, detailing their definition and drawing upon exiting properties from [2]. Section 3.3 outlines the notations and problem formulation of this work. In Section 3.4, we introduce the proposed trajectory-based policy gradient algorithm and establishes its asymptotic convergence towards local optima, utilizing the multi-time stochastic approximation technique from [19]. Section 3.5 presents empirical validation through various experimental setups. Finally, Section 3.6 offers concluding remarks.

3.2 Preliminaries

Recall that the dual representation theorem links different choices of uncertainty sets $\mathcal{U}$ to various risk measures. We extend this framework by constructing risk measures based on ϕ-divergence.

For two probability measures Q and P within the probability space, the ϕ-divergence is defined as:

$$D_\phi(Q, P) = \sum_{z \in \Omega} P(z) \phi\left(\frac{Q(z)}{P(z)}\right), \tag{3.1}$$

where ϕ is a closed and convex function satisfying $\phi(1) = 0$. The choice of the function ϕ directly determines the type of divergence, allowing for various risk measures to be modeled. Below, we present some common choices of ϕ and their corresponding divergences:

1). Total variation distance: $\phi(x) = \frac{1}{2}|x - 1|$.

2). KL divergence: $\phi(x) = x \log x$ for $x \geq 0$.

3). χ^2-divergence: $\phi(x) = (x - 1)^2$.

We now define the ϕ-Divergence-Risk, in which the uncertainty sets $\mathcal{U}$ are constructed based on ϕ-divergence, following the framework described in [2].

Definition 7. (ϕ-Divergence-Risk) Let ϕ be a closed and convex function with $\phi(1) = 0$, and $\beta > 0$. The ϕ-divergence risk measure with divergence level β for a random variable $Z \in \mathcal{Z}$ is defined as

$$\text{PhiD-R}_{\phi,\beta}(Z) := \sup_{Q \in \mathcal{U}} \mathbb{E}_Q[Z],$$

where $\mathcal{U} = \{Q \ll P : D_\phi(Q, P) \leq \beta\}$ with D_ϕ being defined in (3.1).

The definition via dual representation ensures two key outcomes: (1) PhiD-R is a coherent risk measure, as validated by Theorem 3.2 in [2]; and (2) building on insights from [22], solving PhiD-R RL aligns with robust RL, where uncertainties in transition probabilities are characterized by ϕ-divergence. Furthermore, Theorem 5.1 in [2] provides an explicit representation of PhiD-R, which plays a crucial role in developing the policy gradient method discussed in the following sections.

Theorem 6 (Theorem 5.1 of [2]). For any $Z \in \mathcal{Z}$, the ϕ-divergence risk measure has the following representation:

$$\text{PhiD-R}_{\phi,\beta}(Z) = \inf_{\nu>0,\omega\in\mathbb{R}} \left\{ \nu \left[\omega + \mathbb{E}_P \left(\phi^* \left(\frac{Z}{\nu} - \omega + \beta \right) \right) \right] \right\}, \tag{3.2}$$

where ϕ^* is the conjugate of ϕ (the Legendre–Fenchel transform).

It is important to note that the class of ϕ-divergence risk measures encompasses widely used risk measures in risk-sensitive RL, such as CVaR and EVaR, as special cases. For instance, by selecting $\phi(x) = 0$ for $0 \leq x \leq \frac{1}{1-\alpha}$ and $+\infty$ otherwise, we recover CVaR, with $\phi^*(x) = \frac{1}{1-\alpha}\max\{0, x\}$. Additionally, by setting $\beta = 0$, we obtain

$$\text{PhiD-R}_{\phi,\beta}(Z) = \inf_{t\in\mathbb{R}} \left\{ t + \frac{1}{1-\alpha}\mathbb{E}_P \left[(Z-t)^+ \right] \right\},$$

which exactly corresponds to the definition of CVaR as mentioned earlier.

Similarly, by selecting $\phi(x) = x\log x$ for $x \geq 0$, we recover EVaR, with $\phi^*(x) = e^{x-1}$. By setting $\beta = -\ln(1-\alpha)$ [2], we derive

$$\text{PhiD-R}_{\phi,\beta}(Z) = \inf_{\nu>0} \left\{ \nu \ln \mathbb{E}_P \left(e^{\frac{Z}{\nu}} \right) - \nu \ln(1-\alpha) \right\},$$

which corresponds to the representation formula for EVaR.

3.3 Problem Statement

Our goal is to solve PhiD-R RL by minimizing the objective function

$$\min_{\theta} \quad \text{PhiD-R}_{\phi,\beta}\left(J^\theta(x_0) \right) \tag{3.3}$$

for a given divergence level $\beta \geq 0$. This optimization problem seeks to find the optimal policy θ^* that minimizes the risk-sensitive objective. By incorporating the representation (3.2) of the ϕ-divergence risk measure, the optimization problem can be reformulated as follows:

$$\min_{\theta,\nu,\omega} L(\nu,\omega,\theta) := \nu \left[\omega + \mathbb{E}_P \left(\phi^* \left(\frac{J^\theta(x_0)}{\nu} - \omega + \beta\right)\right)\right]. \tag{3.4}$$

While similar formulations have been explored in the literature on risk measures and optimization, our application of this reformulation to the context of RL is novel. Our main idea to solve the optimization problem (3.4) is to adopt a gradient descent method, which will be detailed in the subsequent section. In this chapter, the policy should also satisfies Assumption 3 mentioned in Section 2.3.

3.4 Trajectory-Based Policy Gradient Method

In this section, we introduce a trajectory-based policy gradient algorithm that descends in ν, μ, and θ based on the gradients of $L(\nu,\omega,\theta)$ with respect to ν, ω, and θ, respectively.

The approach is similar to that proposed in Section 2.3. In each iteration, the algorithm generates N trajectories by executing the current policy π. These trajectories are then used to estimate the gradients, and the parameters ν, ω, and θ are updated using stepsizes that satisfy specific conditions. The cost function $J(\xi)$ and the probability $\mathbb{P}_\theta(\xi)$ of generating trajectory ξ are the same as in Section 2.3 and adhere to the same propositions.

We can now proceed to derive the estimated form of these gradients. The derivation details could be found in Appendix B.1.

Gradient estimate w.r.t ν

$$\widehat{\nabla_\nu} L(\nu,\omega,\theta) = \omega + \sum_\xi \mathbb{P}_\theta(\xi)\phi^*\left(\frac{J(\xi)}{\nu} - \omega + \beta\right) - \sum_\xi \mathbb{P}_\theta(\xi)\frac{J(\xi)}{\nu}\frac{\partial \phi^*}{\partial u}\bigg|_{u=\frac{J(\xi)}{\nu}-\omega+\beta}.$$

Gradient estimate w.r.t ω

$$\widehat{\nabla_\omega} L(\nu, \omega, \theta) = \nu - \nu \sum_\xi \mathbb{P}_\theta(\xi) \frac{\partial \phi^*}{\partial u}\bigg|_{u=\frac{J(\xi)}{\nu}-\omega+\beta}.$$

Gradient estimate w.r.t θ

$$\widehat{\nabla_\theta} L(\nu, \omega, \theta) = \nu \sum_\xi \mathbb{P}_\theta(\xi) \nabla_\theta \log \mathbb{P}_\theta(\xi) \phi^* \left(\frac{J(\xi)}{\nu} - \omega + \beta \right).$$

However, these estimates are not immediately usable due to the presence of the unknown transition probability $P(x_{k+1}|x_k, a_k)$ in the expression of $\mathbb{P}_\theta(\xi)$. To address this, we use empirical mean to estimate the sample mean. Moreover, it is important to note that when $\mathbb{P}_\theta(\xi) \neq 0$, the gradients $\nabla_\theta \mathbb{P}_\theta(\xi)$ and $\nabla_\theta \log \mathbb{P}_\theta(\xi)$ can be expressed as $\mathbb{P}_\theta(\xi) \nabla_\theta \log \mathbb{P}_\theta(\xi)$, and the latter is only dependent on π without any reliance on the unknown transition probability $P(x_{k+1}|x_k, a_k)$. By utilizing these insights and generating N trajectories per iteration, we obtain the gradient estimates as:

Gradient estimate w.r.t ν

$$\widetilde{\nabla_\nu} L(\nu, \omega, \theta) = \omega + \sum_\xi \frac{1}{N} \phi^* \left(\frac{J(\xi)}{\nu} - \omega + \beta \right) - \sum_\xi \frac{1}{N} \frac{J(\xi)}{\nu} \frac{\partial \phi^*}{\partial u}\bigg|_{u=\frac{J(\xi)}{\nu}-\omega+\beta}. \tag{3.5}$$

Gradient estimate w.r.t ω

$$\widetilde{\nabla_\omega} L(\nu, \omega, \theta) = \nu - \nu \sum_\xi \frac{1}{N} \frac{\partial \phi^*}{\partial u}\bigg|_{u=\frac{J(\xi)}{\nu}-\omega+\beta}. \tag{3.6}$$

Gradient estimate w.r.t θ

$$\widetilde{\nabla_\theta} L(\nu, \omega, \theta) = \nu \sum_\xi \frac{1}{N} \nabla_\theta \log \mathbb{P}_\theta(\xi) \phi^* \left(\frac{J(\xi)}{\nu} - \omega + \beta \right). \tag{3.7}$$

Based on these gradient estimates and let $\xi_{j,k}$ denote the j-th trajectory generated at iteration

k and properly chosen step sizes $\zeta_1(k)$, $\zeta_2(k)$ and $\zeta_3(k)$, we design the following update rules for parameter ν, ω, θ that will be utilized in our algorithm.

ν-update

$$\begin{aligned}
&\nu_{k+1} \\
&= \Gamma_{\mathcal{N}}\left[\nu_k - \zeta_1(k)\widetilde{\nabla}_\nu L(\nu,\omega,\theta)\big|_{\nu=\nu_k,\omega=\omega_k,\theta=\theta_k}\right] \\
&= \Gamma_{\mathcal{N}}\left[\nu_k - \zeta_1(k)\left(\omega_k + \sum_{j=1}^{N}\frac{1}{N}\phi^*\left(\frac{J(\xi_{j,k})}{\nu_k} - \omega_k + \beta\right) - \sum_{j=1}^{N}\frac{1}{N}\frac{J(\xi_{j,k})}{\nu_k}\frac{\partial\phi^*}{\partial u}\bigg|_{u=\frac{J(\xi_{j,k})}{\nu_k}-\omega_k+\beta}\right)\right].
\end{aligned} \tag{3.8}$$

ω-update

$$\begin{aligned}
\omega_{k+1} &= \Gamma_{\mathcal{R}}\left[\omega_k - \zeta_2(k)\widetilde{\nabla}_\omega L(\nu,\omega,\theta)\big|_{\nu=\nu_k,\omega=\omega_k,\theta=\theta_k}\right] \\
&= \Gamma_{\mathcal{R}}\left[\omega_k - \zeta_2(k)\cdot\left(\nu_k - \nu_k\sum_{j=1}^{N}\frac{1}{N}\frac{\partial\phi^*}{\partial u}\bigg|_{u=\frac{J(\xi_{j,k})}{\nu_k}-\omega_k+\beta}\right)\right].
\end{aligned} \tag{3.9}$$

θ-update

$$\begin{aligned}
\theta_{k+1} &= \Gamma_{\Theta}\left[\theta_k - \zeta_3(k)\widetilde{\nabla}_\theta L(\nu,\omega,\theta)\big|_{\nu=\nu_k,\omega=\omega_k,\theta=\theta_k}\right] \\
&= \Gamma_{\Theta}\left[\theta_k - \zeta_3(k)\left(\nu_k\sum_{j=1}^{N}\frac{1}{N}\nabla_\theta\log\mathbb{P}_\theta(\xi_{j,k})\cdot\phi^*\left(\frac{J(\xi_{j,k})}{\nu_k} - \omega_k + \beta\right)\right)\right].
\end{aligned} \tag{3.10}$$

The projections introduced in the update rules, i.e., $\Gamma_{\mathcal{N}}(\nu) = \text{argmin}_{\nu\in[V_{\min},V_{\max}]}||\nu - \hat{\nu}||_2^2$, $\Gamma_{\mathcal{R}}(\omega) = \text{argmin}_{\omega\in[W_{\min},W_{\max}]}||\omega - \hat{\omega}||_2^2$, $\Gamma_{\Theta}(\theta) = \text{argmin}_{\theta\in\Theta}||\theta - \hat{\theta}||_2^2$, are employed to enforce the updated values to remain within specified bounds, thereby ensuring the convergence of the policy gradient algorithm for PhiD-R. Additionally, we adopt a common assumption regarding the stepsizes utilized in the update rules (3.8) (3.9) (3.10).

Assumption 5. The stepsizes $\zeta_1(k)$, $\zeta_2(k)$ and $\zeta_3(k)$satisfy

$$\sum_k \zeta_1(k) = \sum_k \zeta_2(k) = \sum_k \zeta_3(k) = \infty, \tag{3.11}$$

$$\sum_k \zeta_1^2(k), \sum_k \zeta_2^2(k), \sum_k \zeta_3^2(k) < \infty, \tag{3.12}$$

$$\zeta_1(k) = o(\zeta_2(k)), \zeta_2(k) = o(\zeta_3(k)). \tag{3.13}$$

The first two conditions in Assumption 5 are common in RL problems. The third condition assumes that the stepsizes satisfy the standard requirements of stepsizes in multi-scale stochastic approximation algorithms. Moreover, from Eq (3.13), we observe that the update frequencies for ν, ω, and θ occur at different timescales, with ν updating at the fastest timescale $\zeta_1(k)$, ω updating at a second fast timescale $\zeta_2(k)$, and θ updating at the slowest timescale $\zeta_3(k)$.

Algorithm 3.1 outlines the proposed trajectory-based policy gradient method for PhiD-R. Line 5 details the collection of N trajectories by following the current parameterized policy with θ_k and line 5 updates the parameters. Lines 9 to 13 describe adjustments to the selected ranges for ν and ω. If no adjustments are needed, the iteration ceases, resulting in the local optimal θ.

Theorem 7 provides theoretical guarantees for Algorithm 3.1, establishing its convergence to a locally optimal policy for the optimization problem (3.4).

Theorem 7. (Local Optimality) Under Assumptions 3 and 5, as $k \to \infty$, the policy sequence generated by Algorithm 3.1 converges almost surely to a locally optimal policy θ^*.

Proof Sketch 1. Our proof is inspired by [22] and our EVaR-PG method proposed in Section 2.3. Initially, we treat the updates $(\nu_k, \omega_k, \theta_k)$ as a multi-time scale discrete stochastic approximation, under the condition that the stepsizes satisfy Assumption 5. We prove that the sequences $(\nu_k, \omega_k, \theta_k)$ converge to the solutions of the corresponding continuous-time systems, each with varying convergence rates. Subsequently, we apply Lyapunov analysis to demonstrate that the sequences $(\nu_k, \omega_k, \theta_k)$ further converge to local asymptotically stable points denoted as $(\nu^*, \omega^*, \theta^*)$. Finally, we establish that the attained points $(\nu^*, \omega^*, \theta^*)$ serve as local optimal solutions for the optimization problem (3.3). More details can be found in Appendix B.2.

Algorithm 3.1 PhiD-R RL: A Trajectory-based Policy Gradient Method

Given: divergence level β, parameterized policy $\pi(\cdot|\cdot,\theta)$, tolerance parameters $\epsilon_\nu, \epsilon_\omega$.
Initialization: choose $\nu = \nu_0$, $\omega = \omega_0$, $\theta = \theta_0$ and initial state x_0.
while TRUE **do**
  for $k = 0, 1, 2, \ldots$ **do**
    Generate N trajectories $\{\xi_{j,k}\}_{j=1}^N$ by following policy π_{θ_k} starting from the initial state x_0.
    Update (ν, ω, θ) by

$$\nu_{k+1} = \Gamma_{\mathcal{N}}\Bigg[\nu_k - \zeta_1(k)\Bigg(\omega_k + \sum_{j=1}^N \frac{1}{N}\phi^*\Bigg(\frac{J(\xi_{j,k})}{\nu_k} - \omega_k + \beta\Bigg) - \sum_{j=1}^N \frac{1}{N}\frac{J(\xi_{j,k})}{\nu_k}\frac{\partial \phi^*}{\partial u}\Bigg|_{u=\frac{J(\xi_{j,k})}{\nu_k}-\omega_k+\beta}\Bigg)\Bigg],$$

$$\omega_{k+1} = \Gamma_{\mathcal{R}}\Bigg[\omega_k - \zeta_2(k)\cdot\Bigg(\nu_k - \nu_k \sum_{j=1}^N \frac{1}{N}\frac{\partial \phi^*}{\partial u}\Bigg|_{u=\frac{J(\xi_{j,k})}{\nu_k}-\omega_k+\beta}\Bigg)\Bigg],$$

$$\theta_{k+1} = \Gamma_{\Theta}\Bigg[\theta_k - \zeta_3(k)\Bigg(\nu_k \sum_{j=1}^N \frac{1}{N}\nabla_\theta \log \mathbb{P}_\theta(\xi_{j,k})\cdot\phi^*\Bigg(\frac{J(\xi_{j,k})}{\nu_k} - \omega_k + \beta\Bigg)\Bigg)\Bigg].$$

  end for
  if ν_k lies within the ϵ_ν-neighborhood of the boundary **then**
    Extend the boundary for ν
  else if ω_k lies within the ϵ_ω-neighborhood of the boundary **then**
    Extend the boundary for ω
  else
    Return (ν, ω, θ) and **terminate**
  end if
end while

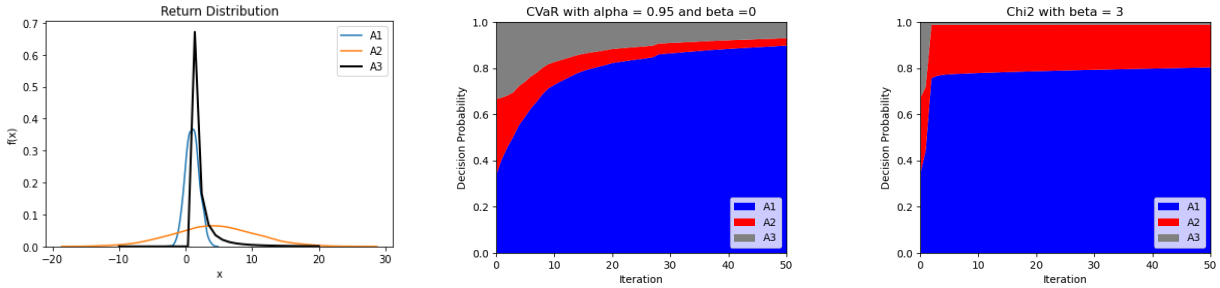
3.5 Experiments

In this section, we present numerical examples to demonstrate the practicality and efficiency of the proposed algorithms. We first validate our approach using an investment problem and the optimal stopping problem, as utilized in related work [22, 70, 95], highlighting comparison over existing methods. Additionally, we conduct a more comprehensive evaluation using OpenAI's Gym environment to further demonstrate the generalizability of our algorithms.

3.5.1 Investment Problem

We conduct a validation of our method using the same experimental setup as [95]. We examine a scenario involving a trading agent with options to invest in one of three assets. The returns of the first two assets, A_1 and A_2, follow normal distributions: A_1 is distributed as $\mathcal{N}(1, 1)$, and A_2 as $\mathcal{N}(4, 6)$. The third asset, A_3, exhibits a Pareto distribution characterized by $f(x) = \frac{\alpha}{x^{\alpha+1}}$ for $x > 1$ with a parameter $\alpha = 1.5$. This distribution results in a mean return of 3 for A_3, but with an infinite variance, reflecting the heavy-tailed distributions commonly employed in financial modeling [93]. The agent's investment decisions are randomized, with the probability of choosing asset A_i denoted as $P(A_i) \sim \exp(\theta_i)$, where $\theta \in \mathbb{R}^3$ represents the policy parameters. Here we plot the results of running 50 iterations, with $10,000$ trajectories to estimate gradients in each iteration.

In the experiment, we choose the Radon-Nikodym derivative and χ^2-divergence as examples. Figure 3.1 illustrates how the probabilities of choosing A_1, A_2, and A_3 change over iterations. For Radon-Nikodym derivative, the agent is highly risk-averse at $\alpha = 0.95$, favoring A_1 and the agent is less risk-averse at$\alpha = 0.05$, resulting in shifts in probabilities. For χ^2, $P(A_i)$ also changes with different β. Notably, different ϕ-divergence reflects different risk preferences as the probability distribution differs. These results align with our theoretical analysis. Moreover, in comparison to the experimental results in [95], our method exhibits enhanced efficiency, achieving convergence with a small number of iterations, even when applied to more complex forms of risk measures.



(a) Distribution of asset return. (b) PhiD-R with RN derivative (c) PhiD-R with χ^2-divergence

Figure 3.1: Probability of selecting each asset versus training iterations, for policies generated by solving PhiD-R RL based on Radon-Nikodym derivative and χ^2-divergence.

3.5.2 Optimal Stopping Problem

In this section, we consider a more complex setup similar to the CVaR and EVaR policy gradient work [22] [70]. The environment is designed as an optimal stopping problem, where the state at each time step k is represented by $x = [k, c_k]$. Here, c_k denotes the cost at time k. The cost sequence $\{c_k\}_{k=0}^{T}$ is generated as follows: at each time step, the cost at the next time step either increases by a constant factor $f_u > 1$ (i.e., $c_{k+1} = f_u c_k$) with probability p, or decreases by a constant factor $f_d < 1$ (i.e., $c_{k+1} = f_d c_k$) with probability $1-p$. The agent's task is to decide whether to accept the current cost ($a_k = 1$) or wait ($a_k = 0$) at each time step. If the agent chooses to accept the cost or the time step reaches $k = T$, the cost is set to $\min(K, c_k)$, where K represents the cost threshold. However, if the agent chooses to wait, an additional cost of p_h is incurred. Hence, the discounted cost can be expressed as $J^{\theta}(x) = \sum_{k=0}^{T} \gamma^k \left(\mathbf{1}\{a_k = 1\} \min(K, c_k) + \mathbf{1}\{a_k = 0\} p_h\right)$.

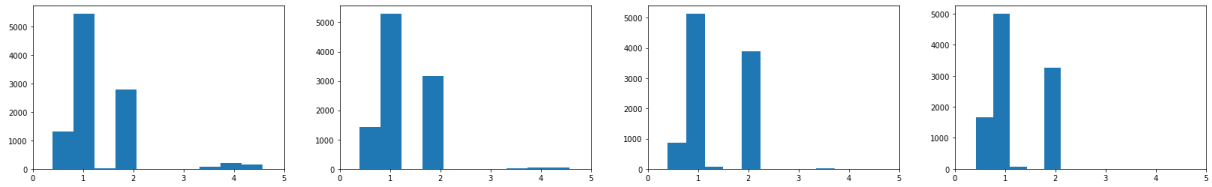
Here we choose $x_0 = [1; 0]$, $p_h = 0.1$, $T = 20$, $K = 5$, $\gamma = 0.95$, $f_u = 2$, $f_d = 0.5$, $p = 0.65$, $N = 500,000$ and $\Theta = [-20, 20]^{\kappa_1}$, where the dimension of the basis function is $\kappa_1 = 64$. Furthermore, we use Boltzmann policies

$$\left\{\theta : \{\theta_{x,a}\}_{x\in\mathcal{X}, a\in\mathcal{A}},\ \mu_\theta(a|x) = \frac{\exp(\theta_{x,a}^{\mathrm{T}} x_f(x))}{\sum_{a\in\mathcal{A}} \exp(\theta_{x,a}^{\mathrm{T}} x_f(x))}\right\},$$

where $x_f(x)$ is the feature chosen by RBF at state x.

We evaluate the effectiveness of our algorithm using various ϕ-divergences. First, we employ the Radon-Nikodym derivative as the ϕ-divergence, corresponding to the widely-used CVaR measure. Next, we consider the KL divergence, corresponding to the EVaR, a relatively recent risk measure adopted in risk-sensitive RL [71]. These first two choices demonstrate our approach's efficiency with popular risk measures, offering fresh perspectives on tackling these risk measures in risk-sensitive RL. Furthermore, we explore the χ^2 divergence, a common divergence in RL, yet without a designated risk measure defined by this divergence. This experiment highlights our algorithm's potential in addressing less clear or undefined risk measures, potentially inspiring new research on innovative risk measures. Finally, we utilize the squared Hellinger distance to

underscore our algorithm's necessity and advantages over other policy gradient methods. The frequency distribution of costs under PhiD-R with different choices of ϕ-divergence is presented in Figure 3.2.

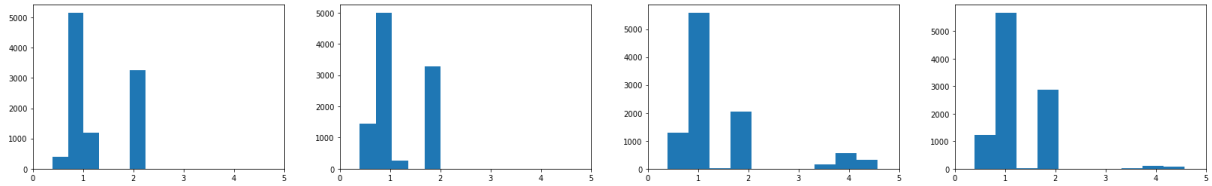


(a) $\alpha = 0.05$, $\beta = 0$, $\mathbb{E}[Z] = 1.35$ and CVaR $= 1.42$

(b) $\alpha = 0.95$, $\beta = 0$, $\mathbb{E}[Z] = 1.29$ and CVaR $= 2.49$

(c) $\alpha = 0.05$, $\beta = 3$, $\mathbb{E}[Z] = 1.36$ and EVaR $= 1.54$

(d) $\alpha = 0.95$, $\beta = 0.05$, $\mathbb{E}[Z] = 1.25$ and EVaR $= 2.67$



(e) $\beta = 0.95$, $\mathbb{E}[Z] = 1.31$ and $\text{PR}_{\chi^2,\beta} = 1.79$

(f) $\beta = 3$, $\mathbb{E}[Z] = 1.27$ and $\text{PR}_{\chi^2,\beta} = 1.83$

(g) $\beta = 0.95$, with $\mathbb{E}[Z] = 1.46$ and $\text{PR}_{\text{shd},\beta} = 1.54$

(h) $\beta = 3$, with $\mathbb{E}[Z] = 1.29$ and $\text{PR}_{\text{shd},\beta} = 1.97$

Figure 3.2: Frequency distribution of costs under PhiD-R defined by: 1). Radon-Nikodym derivative (CVaR); 2) KL divergence (EVaR); 3) χ^2 divergence and 4) Squared Hellinger Distance with different choices of parameters.

Radon-Nikodym Derivative (CVaR)

We begin by selecting the ϕ-divergence as the Radon-Nikodym derivative, where $\phi(x) = 0$ for $0 \leq x \leq \frac{1}{1-\alpha}$ and $+\infty$ otherwise. In this case, the conjugate function $\phi^*(x)$ is

$$\frac{1}{1-\alpha}\max\{0, x\} = \frac{1}{\alpha}(0, x)^+.$$

By setting $\beta = 0$, the corresponding ϕ-divergence risk measure is CVaR and we obtain the following expressions.

$$\text{CVaR}_\alpha(Z) = \inf_{\nu>0,\omega\in\mathbb{R}} \left\{ \nu\omega + \frac{1}{1-\alpha}\mathbb{E}_P\left((Z-\nu\omega)^+\right) \right\}.$$

Notice that $\frac{d}{dx}\phi^*(x) = \frac{1}{1-\alpha}\mathbb{I}\{x > 0\}$, where $\mathbb{I}$ is the indicator function and $\frac{d^2}{dx^2}\phi^*(x) = 0$.

By employing Algorithm 3.1 with CVaR update rules at various confidence levels α, we obtain the results in Figures 3.2a and 3.2b. The mean of discounted costs generated by following the optimal policy at $\alpha = 0.05$ exceeds the mean at $\alpha = 0.95$, whereas the opposite holds true for the CVaR value. The observed results align with the theoretical properties of CVaR. Specifically, when the risk aversion parameter (α) is set to 0.05, the agent exhibits a risk-averse behavior, opting for a safer strategy that results in higher costs but reduced risk exposure. Conversely, for $\alpha = 0.95$, the agent demonstrates risk-seeking tendencies, prioritizing lower costs despite the associated higher level of risk.

KL Divergence (EVaR)

In this case, we choose the ϕ-divergence to be the KL divergence, denoted as $\phi(x) = x \log x$ for $x \geq 0$. Consequently, we have

$$\phi^*(x) = e^{x-1}$$

and $\beta = -\ln(1-\alpha)$ according to [2]. With this selection, the resulting ϕ-divergence risk measure corresponds to EVaR, given by

$$\text{EVaR}_\alpha(Z) = \inf_{\nu>0,\omega\in\mathbb{R}} \left\{ \nu \left[\omega + \mathbb{E}_P \left(e^{\frac{Z}{\nu} - \omega + \beta} \right) \right] \right\}.$$

By employing Algorithm 3.1 and incorporating EVaR update rules, we obtained results for two specific risk parameter settings: $\alpha = 0.05$ ($\beta = 3$) and $\alpha = 0.95$ ($\beta = 0.05$). For the case where $\alpha = 0.05$, the agent demonstrates a risk-averse preference by selecting higher costs to mitigate potential high risks. Conversely, for $\alpha = 0.95$, the agent exhibits a more aggressive behavior, seeking to minimize costs even in the presence of higher risks. Furthermore, the observation that EVaR consistently exceeds CVaR under the same distribution of a random variable aligns with the theoretical facts in [2].

χ^2 Divergence

In this case, we utilize the χ^2 divergence and set $\phi(x) = (x-1)^2$. Consequently, we obtain

$$\phi^*(x) = \frac{x^2}{4} + x$$

and $\beta > 0$. Thus, we have

$$\text{PR}_{\chi^2,\beta}(Z) = \inf_{\nu>0,\omega\in\mathbb{R}} \left\{ \nu \mathbb{E}_P \left(\frac{\left(\frac{Z}{\nu} - \omega + \beta\right)^2}{4} + \frac{Z}{\nu} + \beta \right) \right\}.$$

Applying Algorithm 3.1 with corresponding update rules for χ^2 divergence, we obtain the following results. Figure 3.2e and 3.2f illustrate that when $\beta = 0.95$, the mean of discounted costs exceeds the mean in the case where $\beta = 3$, while the risk value is lower. The selection of β indeed reflects the decision-maker's attitude towards risk.

Squared Hellinger Distance

In [70], the authors propose a two-update-rules trajectory-based policy gradient method to solve EVaR in risk-sensitive RL. However, the two-update-rules algorithm is not applicable to the entire class of ϕ-divergences. Here, we illustrate the necessity and practicality of our Algorithm 3.1 by employing the squared Hellinger distance as an example.

We choose $\phi(x) = (\sqrt{x} - 1)^2$ and the conjugate function is

$$\phi^*(x) = \frac{1}{1-x} - 1$$

for $x > 0$. Hence, this ϕ-divergence risk measure is given by

$$\text{PR}_{\phi,\beta}(Z) = \inf \left\{ \nu \left[\omega + \mathbb{E}_P \left(\frac{1}{1 - \frac{Z}{\nu} + \omega - \beta} - 1 \right) \right] \right\},$$

where $'\inf'$ is taken over the set $\left\{\nu, \omega : \nu > 0, \omega \in \mathbb{R}, \frac{Z}{\nu} - \omega + \beta > 0\right\}$.

Applying Algorithm 3.1 for the square Hellinger distance with varying values of β, we obtained the results shown in Figures 3.2g and 3.2h. The figures clearly indicate that when $\beta = 0.95$, the sample mean of the discounted cost is 1.46, which exceeds the mean in the case where $\beta = 3$, while the risk is lower. The selection of the parameter β directly reflects the risk preference exhibited by the agent.

We present a summary of numerical results for PhiD-R using various ϕ-divergences under different parameter settings in Table 3.1, wher RN derivative means Radon-Nikodym derivative and SH distance means squared Hellinger distance. The data shows that all risk values exceed the mean and vary with parameter choices, validating the algorithm and demonstrating its alignment with established risk concerns. Additionally, these results offer insights into interpreting new risk measures, such as PhiD-R with χ^2-divergence and squared Hellinger distance, especially when supported by extensive simulations across diverse parameters. This adaptability in parameter selection highlights the flexibility of our approach, allowing decision-makers to align with their risk preferences while maintaining local optimality and efficiency.

Table 3.1: Numerical results of different choices of ϕ-divergence.

ϕ-divergence	Parameters	Mean	PhiD-R
Radon-Nikodym derivative	$\alpha = 0.05, \beta = 0$	1.35	1.42
	$\alpha = 0.95, \beta = 0$	1.29	2.49
KL-divergence	$\alpha = 0.05, \beta = 3$	1.36	1.54
	$\alpha = 0.95, \beta = 0.05$	1.25	2.67
χ^2-divergence	$\beta = 0.95$	1.31	1.79
	$\beta = 3$	1.27	1.83
Squared Hellinger distance	$\beta = 0.95$	1.46	1.54
	$\beta = 3$	1.29	1.79

3.5.3 Experiments on Gym

In this section, we validate our approach using OpenAI's Gym [20]. Specifically, we choose the `CartPole-v1` environment, which involves a pole attached to a cart moving along a frictionless track. The goal is to prevent the pole from falling over by applying force to the cart. The action

space is discrete, with two possible actions: pushing the cart to the left or to the right. Since we consider cost in this work, we design the environment such that the agent receives a cost of 0 for every time step the pole remains upright and a cost of $+1$ for failing to keep the pole upright. The goal is to keep the pole balanced for as many time steps as possible, up to a maximum of T steps. The episode terminates if the pole angle exceeds $\pm 15^{\circ}$ or the cart moves more than 2.4 units from the center. We run both risk-neutral RL policy gradient and our approach with different choices of ϕ-divergence. For these experiments, we set $\gamma = 0.99$ and run $N = 10,000$ episodes with a time step of $T = 1,000$.

As shown in Figure 3.3, the results are plotted with episode length on the y-axis and episodes on the x-axis. Longer episode lengths indicate better performance, and changes in episode length over time illustrate the convergence speed of the algorithm. The upper panel presents the episode length for each individual episode, while the lower panel shows the mean episode length over the past 50 episodes using a sliding window. This results in a smoother blue curve (mean) in the lower panel, with the yellow shaded area around the blue curve providing a visual indication of the variability around the moving average. A smaller shaded area suggests more consistent and robust algorithm performance, while a larger shaded area indicates greater variability and less consistency. This visualization aids in understanding the stability of the training process over time. Figure 3.3 shows the result of running policy gradient method for risk-neutral RL.

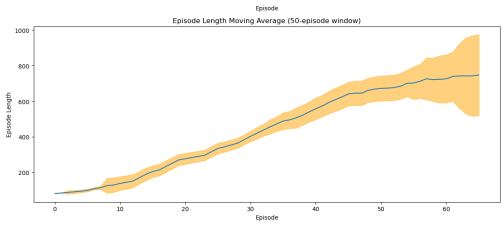
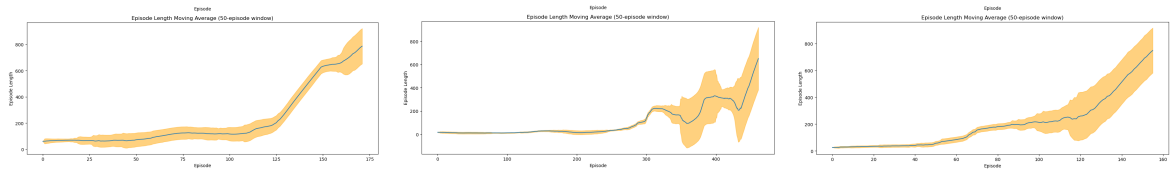


Figure 3.3: Episode length versus episodes for risk-neutral RL.

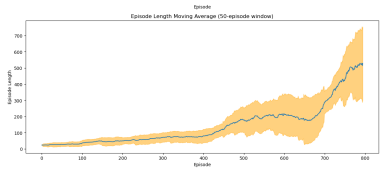
We then apply our approach to the same environment using three different choices of divergence: 1) Radon-Nikodym derivative (Figure 3.4a, 3.4b), 2) KL divergence (Figure 3.4c, 3.4d), and 3) χ^2-divergence (Figure 3.4e, 3.4f). Although the training processes vary with different divergences and parameters, the overall trends are similar. When the divergence level β is smaller,



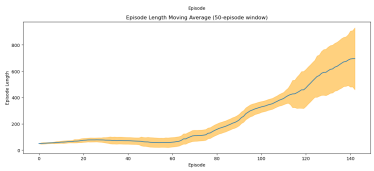
(a) RN derivative with $\beta = 0, \alpha = 0.95$.

(b) RN derivative with $\beta = 0, \alpha = 0.1$

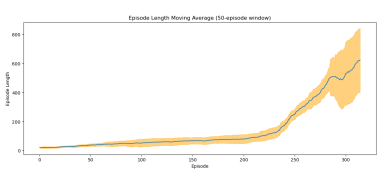
(c) KL divergence with $\beta = 0.1$



(d) KL divergence with $\beta = 3$



(e) χ^2 with $\beta = 0.1$



(f) χ^2 with $\beta = 3$

Figure 3.4: Episode length versus episodes for PhiD-R defined by: 1) Radon-Nikodym derivative (CVaR); 2) KL divergence (EVaR); 3) χ^2-divergence with different choices of parameters.

the process converges more quickly since the agent is more risk-seeking and prefers taking more aggressive actions to balance the cart pole, as shown in Figure 3.4a, 3.4c, 3.4e. These processes are similar to risk-neutral RL, as all agents were more risk-seeking (risk-neutral implies risk-seeking behavior). Conversely, with larger β values, the agent exhibite more risk-averse behavior, indicated by a flatter curve during the initial phase, leading to stable episode lengths compared to the smaller β case (shown as Figure 3.4b, 3.4d, 3.4f). This behavior also suggests that the agent is more likely to be trapped in a local optimum.

3.6 Conclusion

In this chapter, we have applied a new class of risk measures named PhiD-R to risk-sensitive RL. We have proposed a trajectory-based policy gradient method tailored to this class of risk measures, utilizing an explicit representation that accommodates all forms of ϕ-divergence. Our approach has extended upon previous methods targeting specific risk measures and provided a comprehensive solution that encompasses the entire range of ϕ-divergence. Furthermore, we have demonstrated the convergence of our algorithms using a multi-time stochastic approximation approach. Through numerical simulation results, we have validated the efficiency and practicality of our algorithms.

Chapter 4

Robust Risk-Sensitive Reinforcement Learning with CVaR

4.1 Introduction

Real-world applications of RL frequently encounter uncertainties in MDP elements, such as transition probabilities and reward/cost functions, leading to estimation errors in RL algorithms and subsequent sensitivity to model inaccuracies, thus impairing performance [59, 102, 106]. In light of these challenges, RMDPs have been developed to focus on optimal policies that accommodate worst-case transition probabilities within an ambiguity set [45], with most studies assuming known and rectangular ambiguity sets due to computational considerations [12, 45, 46, 48, 76, 101].

The existing RMDP research has largely focused on risk-neutral objectives that minimize the expected total discounted costs. Although risk-sensitive RL is widely popular, its robustness within the RMDP framework is not clear. While Chow et al. (2015) [24] roughly mention how solving CVaR can enhance the robustness of risk-neutral RL in certain uncertainty sets, there is a noticeable gap in understanding how CVaR's robustness fares against various types of uncertainty sets.

Chapter Contribution: This chapter presents a novel and comprehensive investigation into the robustness of risk-sensitive RL within RMDP. The primary goal is to determine an optimal

policy that minimizes the robust CVaR value. This value is characterized as the highest CVaR of the total discounted cost across transition probabilities within a defined rectangular ambiguity set. We initially explore scenarios where the uncertain budget is fixed, and utilize the coherent properties of CVaR and the dual representation theorem to convert the optimization challenge into a manageable risk-sensitive RL problem, facilitating the use of existing algorithms. Furthermore, considering that in many real-world applications, ambiguity sets are often dynamic and influenced by decision-making processes [77], we delve deep into a more challenging setup about designing robust CVaR optimization under decision-dependent uncertainty. To tackle this problem, we introduce a new coherent risk measure NCVaR and propose a crucial decomposition theorem. We develop value iteration algorithms for NCVaR and validate our methods through simulation experiments. Based on these results, the emergence of NCVaR not only enhances the robustness of CVaR RL under decision-dependent uncertainty but also brings insights to risk-sensitive RL. Adopting NCVaR as the risk measure for risk-sensitive RL provides strong robustness compared to risk-neutral RL while rationally capturing risk. This makes NCVaR promising for potential future research and also shed lights on solving decision-dependent uncertainty for RL.

Chapter Organization: The structure of this chapter is as follows. In Section 4.2, we outline mathematical foundations and problem formulation. Section 4.3 discusses solutions utilizing predetermined ambiguity sets and risk-sensitive RL methods. Section 4.4 focuses on undetermined ambiguity sets and corresponding value iteration algorithms. Section 4.5 validates our approaches through experimental simulations and presents the numerical results. Conclusions are drawn in Section 4.6.

4.2 Preliminaries

4.2.1 RMDP and Ambiguity Set

Addressing robustness, the transition probability P is known to belong to a non-empty, compact set $\mathcal{P}$, with the uncertain transition probability denoted as $\tilde{P} \in \mathcal{P}$. The robust policy evaluation

over non-rectangular ambiguity sets $\mathcal{P}$ is known to be NP-hard, even with a fixed policy π [102]. Therefore, robust RL research often focuses on rectangular ambiguity sets. In this work, we examine a specific rectangular ambiguity set:

$$\mathcal{P} = \left\{ \tilde{P} : \sum_{x' \in \mathcal{X}} \tilde{P}(x'|x,a) = 1, \;\; D(\tilde{P}, P) \leq K \right\},$$

where K is the non-negative uncertain budget and the divergence measure $D(\tilde{P}, P)$ satisfies

$$D(\tilde{P}, P) = \sum_{x' \in \mathcal{X}} P(x'|x,a) \phi \left(\frac{\tilde{P}(x'|x,a)}{P(x'|x,a)} \right) \leq K. \tag{4.1}$$

In (4.1), ϕ represents the ϕ-divergence measure.

4.3 Robust CVaR RL with Predetermined Ambiguity Set

In this section, the robust CVaR value is defined as the worst-case CVaR value of a policy π when starting from the initial state x_0 and traversing through transition probabilities specified in the ambiguity set. The objective is to minimize this robust CVaR value across all history-dependent policies, as expressed by the following optimization problem:

$$\min_{\pi \in \Pi_H} \max_{\tilde{P} \in \mathcal{P}} \text{CVaR}_\alpha \left(\lim_{T \to \infty} J^\pi(x_0) \right). \tag{4.2}$$

The sets Π_H and $\mathcal{P}$ are both non-empty and compact. Additionally, the objective function is finite due to $\gamma < 1$. Thus, the minimum and maximum values can be achieved, as guaranteed by the Weierstrass theorem in optimization theory [46]. This theorem ensures that the optimization problem is well-defined and can be effectively solved to obtain the desired policy that minimizes the robust CVaR value under the given constraints. Contrasting with the robustness analysis of CVaR in [24], our approach evaluates the inner CVaR objective in Equation (4.2) across the entire set $\mathcal{P}$, instead of limiting the analysis to the true transition probabilities P alone. This broader

evaluation provides a more comprehensive analysis of the robustness of CVaR in diverse uncertain environments.

Recalling the coherent nature of CVaR as a risk measure and leveraging the dual representation theorem, the original optimization problem (4.2) can be reformulated as follows:

$$\min_{\pi\in\Pi_H}\max_{\tilde{P}\in\mathcal{P}}\max_{Q\in\mathcal{U}_{\text{CVaR}}}\mathbb{E}_Q\left(\lim_{T\to\infty}J^\pi(x_0)\right). \tag{4.3}$$

where $\mathcal{U}_{\text{CVaR}} = \{Q \ll \tilde{P} : 0 \leq Q(x'|x,a)/\tilde{P}(x'|x,a) \leq \frac{1}{\alpha}\}$. Notice that the $'\sup'$ has been replaced by $'\max'$ since $\mathcal{U}_{\text{CVaR}}$ is convex and compact and the objective function is continuous in Q.

We first focus on solving problem (4.3) with a predetermined ambiguity set, where the uncertain budget remains fixed for every state and action. Our approach involves combining two inner maximization problems by analyzing the divergence $D(Q,P)$. Under the assumption that the function ϕ in (4.1) is chosen such that $D(Q,P)$ remains bounded, i.e., $D(Q,P) \leq \tilde{K}$ (a condition satisfied by the divergence measure used in this study), we show that problem (4.3) can be reformulated to:

$$\min_{\pi\in\Pi_H}\max_{Q\in\mathcal{Q}}\mathbb{E}_Q\left(\lim_{T\to\infty}J^\pi(x_0)\right), \tag{4.4}$$

where $\mathcal{Q} = \left\{Q : D(Q,P) \leq \tilde{K}\right\}$ represents the uncertain transition problem set.

This approach effectively addresses robust CVaR across diverse uncertainty sets by combining the set's divergence measure with the Radon-Nikodym derivative, forming a new envelope set for risk-sensitive RL. This strategy not only links the robustness of risk-sensitive RL with its intrinsic transformation but also provides a universal framework for evaluating CVaR's robustness. We further illustrate this approach by analyzing two specific ϕ-divergence measures.

4.3.1 Radon-Nikodym Derivative

Firstly, we consider the scenario where ϕ-divergence is Radon-Nikodym derivative, subject to a fixed uncertain budget for all states and actions: $D_{\text{RN}}(\tilde{P},P) = \frac{\tilde{P}(x'|x,a)}{P(x'|x,a)} \in [0,K]$, where $K \geq 0$ is

a predetermined constant.

Consequently, we obtain: $D_{\text{RN}}(Q,P) \in \left[0, \frac{K}{\alpha}\right]$. In this context, the original optimization problem (4.3) transforms into:

$$\min_{\pi\in\Pi_H} \max_{Q\in\mathcal{U}_{\text{RN}}} \mathbb{E}_Q \left(\lim_{T\to\infty} J^\pi(x_0) \right), \tag{4.5}$$

where $\mathcal{U}_{\text{RN}} = \left\{ Q \ll P : D_{\text{RN}}(Q,P) \in [0, \frac{K}{\alpha}] \right\}$.

Notice that solving problem (4.5) is equivalent to solving the following CVaR optimization problem with confidence level $\alpha' = \frac{\alpha}{K}$:

$$\min_{\pi\in\Pi_H} \text{CVaR}_{\alpha'} \left(\lim_{T\to\infty} J^\pi(x_0) \right),$$

which can be solve by employing CVaR value iteration algorithms proposed in [24].

4.3.2 KL Divergence

In this scenario, we consider that the uncertain transition probability $\tilde{P}$ is defined in the neighborhood of the true transition probability P using the KL divergence, given by: $D_{\text{KL}}(\tilde{P}, P) = \sum_{x'\in\mathcal{X}} \tilde{P}(x'|x,a) \log\left(\frac{\tilde{P}(x'|x,a)}{P(x'|x,a)}\right) \leq K$, where $K \geq 0$ is a fixed value. Without loss of generality, we set $K = \ln\kappa$ with $\kappa \geq 1$. We can combine the two inner maximization problems into one, as the KL divergence of Q and P satisfies: $D_{\text{KL}}(Q,P) \leq -\ln\alpha + 1/\alpha \ln\kappa = -\ln(\alpha/\kappa^{\frac{1}{\alpha}})$. Then, the original optimization problem (4.3) is transformed into:

$$\min_{\pi\in\Pi_H} \max_{Q\in\mathcal{U}_{\text{KL}}} \mathbb{E}_Q \left(\lim_{T\to\infty} J^\pi(x_0) \right), \tag{4.6}$$

where $\mathcal{U}_{\text{KL}} = \left\{ Q \ll P : D_{\text{KL}}(Q,P) \leq -\ln\frac{\alpha}{\kappa^{\frac{1}{\alpha}}} \right\}$.

Notice that solving problem (4.6) is equivalent to solving the following EVaR optimization

problem with confidence level $\alpha' = \alpha/\kappa^{\frac{1}{\alpha}}$:

$$\min_{\pi \in \Pi_H} \text{EVaR}_{\alpha'} \left(\lim_{T \to \infty} J^{\pi}(x_0) \right).$$

The problem could be solved by the approaches we proposed in Chapter 2.

4.4 Robust CVaR RL with Decision-Dependent Uncertainty

In real-world scenarios, ambiguity sets can dynamically change due to decisions made during optimization, introducing endogenous uncertainty [63]. This variability means that the uncertain budget can fluctuate over time, adding complexity to robust CVaR optimization analysis. To tackle this decision-dependent uncertainty, we focus on the Radon-Nikodym derivative, i.e.,

$$D_{\text{RN}}(\tilde{P}, P) = \frac{\tilde{P}(x'|x,a)}{P(x'|x,a)} \in [0, \vec{\kappa}(x,a)], \forall (x,a) \in \mathcal{X} \times \mathcal{A},$$

where $\vec{\kappa} := \{\vec{\kappa}(x,a), \forall s \in \mathcal{S}, a \in \mathcal{A}\}$ is the decision-dependent uncertainty budget vector.

By combining the dual representation theorem of CVaR, we obtain the following expression:

$$D_{\text{RN}}(Q, P) = \frac{Q(x'|x,a)}{P(x'|x,a)} \in \left[0, \frac{\vec{\kappa}(x,a)}{\alpha}\right], \forall (x,a) \in \mathcal{X} \times \mathcal{A}.$$

The problem at hand cannot be straightforwardly addressed by treating it as a fixed confidence level CVaR optimization. To overcome this challenge, we introduce a novel risk measure called NCVaR, which incorporates both the confidence level α and an undetermined uncertain budget vector $\vec{\kappa}$. Before delving into its definition, we set forth an assumption to ensure that both NCVaR and the uncertain budget are meaningful.

Assumption 6. The undetermined uncertain budget satisfies $1 \leq \vec{\kappa}(x,a) \leq K_{\text{max}}, \forall x \in \mathcal{X}$ and $a \in \mathcal{A}$. Here $K_{\max} \geq 1$ is a real value.

Definition 8. For a random variable $Z : \Omega \to \mathbb{R}$ with probability mass function (p.m.f.) P, the

NCVaR at a given confidence level $\alpha \in (0, 1]$ with an undetermined uncertain budget $\vec{\kappa}$ is defined as follows:

$$\text{NCVaR}_{\alpha,\vec{\kappa}}(Z) = \sup_{Q\in\mathcal{Q}} \mathbb{E}_Q[Z], \tag{4.7}$$

where $\mathcal{Q} = \left\{Q : D_{\text{RN}}(Q, P) = \frac{Q(\omega)}{P(\omega)} \in \left[0, \frac{\vec{\kappa}(\omega)}{\alpha}\right], \forall\omega \in \Omega\right\}$.

It's easy to observe that when $P(\omega) = 0$, it implies $Q(\omega) = 0$, indicating that Q is absolutely continuous with respect to P (i.e., $Q \ll P$). By leveraging Theorem 3.2 in [2], we can demonstrate that NCVaR is a coherent risk measure, which provides a solid theoretical foundation for employing NCVaR in practical applications and risk-sensitive RL scenarios.

As a consequence of the coherency property, solving problem (4.4) with an undetermined uncertain budget defined by the Radon-Nikodym derivative is equivalently transformed into:

$$\min_{\pi\in\Pi_H} \text{NCVaR}_{\alpha,\vec{\kappa}}\left(\lim_{T\to\infty} J^\pi(x_0)\right). \tag{4.8}$$

Given the computational challenges associated with directly computing NCVaR, as it requires knowledge of the entire distribution of the total discounted cost, we present a decomposition theorem for NCVaR, which is key to simplifying NCVaR computation and the proof is detailed in Theorem 21 of [82].

Theorem 8. (NCVaR Decomposition) For any $\alpha \in (0, 1]$ and $\vec{\kappa}$ satisfying Assumption 6, the $\text{NCVaR}_{\alpha,\vec{\kappa}}$ has the following decomposition

$$\text{NCVaR}_{\alpha,\vec{\kappa}}(Z|H_t, \pi) = \max_{\xi\in\mathcal{U}_{\text{NCVaR}}(\alpha,\vec{\kappa}(x_t,a_t),P(\cdot|x_t,a_t))} \mathbb{E}_P\left[\xi_{x_{t+1}} \cdot \text{NCVaR}_{\alpha\xi,\vec{\kappa}}(Z|H_{t+1}, \pi)|H_t, \pi\right],$$

where $\xi(x_{t+1}) = \frac{Q(x'|x,a)}{P(x'|x,a)} \geq 0$ is in the set

$$\begin{aligned}&\mathcal{U}_{\text{NCVaR}}(\alpha, \vec{\kappa}(x_t, a_t), P(\cdot|x_t, a_t))\\ &= \left\{\xi : \xi(x_{t+1}) \in \left[0, \frac{\vec{\kappa}(x_t, a_t)}{\alpha}\right), \sum_{x_{t+1}\in\mathcal{X}} \xi(x_{t+1})P(x_{t+1}|x_t, a_t) = 1\right\}.\end{aligned}$$

This decomposition theorem provides a valuable insight to NCVaR computation, effectively linking the risk measure between different states, and facilitates a more tractable approach to handling the complexity of NCVaR evaluation within risk-sensitive RL under the RMDP framework. In light of the distinct confidence levels on both sides of equation (8), we introduce an augmented continuous space $\mathcal{Y} = (0, 1]$ to represent the domain of confidence levels.

Accordingly, the value-function $V(x, y)$ for every $(x, y) \in \mathcal{X} \times \mathcal{Y}$ is defined as:

$$V(x, y) = \min_{\pi \in \Pi_H} \text{NCVaR}_{y,\vec{\kappa}} \left(\lim_{T \to \infty} J^{\pi}(x) \right).$$

The Bellman operator $\mathbf{T} : \mathcal{X} \times \mathcal{Y} \to \mathcal{X} \times \mathcal{Y}$ is defined as:

$$\mathbf{T}[V](x, y) = \min_{a \in \mathcal{A}} \left[C(x, a) + \gamma \max_{\xi \in \mathcal{U}_{\text{NCVaR}}(y, \vec{\kappa}(x,a), P(\cdot|x,a))} \sum_{x' \in \mathcal{X}} \xi(x') V(x', y\xi(x')) P(x'|x, a) \right].$$

Lemma 2 introduces some important properties for the NCVaR Bellman operator.

Lemma 2. The Bellman operator $\mathbf{T}$ has the following properties: P1) *Monotonicity*; P2) *Transition Invariance*; P3) *Contraction*; P4) *Concavity preserving*: Suppose $yV(x, y)$ is concave in $y \in \mathcal{Y}, \forall x \in \mathcal{X}$. Then the maximization problem in (4.8) is concave and $y\mathbf{T}[V](x, y)$ is also concave in y.

Properties P1-P3 are similar to standard dynamic programming [15], and are key to design a convergent value iteration method. P4 ensures that value-iteration updates involve concave, and thus tractable, optimization problems. More details can be found in Appendix C.1.

Based on Lemma 2, we are able to propose the following theorem, which demonstrates the existence of a unique fixed-point solution and outline a method for deriving an optimal policy.

Theorem 9. The unique fixed-point solution $V^*(x, y)$ of $\mathbf{T}[V](x, y) = V(x, y)$ exists and equals to the optimal value of optimization problem (4.8), i.e.,

$$V^*(x, y) = \min_{\pi \in \Pi_H} \text{NCVaR}_{y,\vec{\kappa}} \left(\lim_{T \to \infty} J^{\pi}(x) \right).$$

Proof. Please refer to Appendix C.2 for details. □

Although the problem is optimized over history-dependent policies, we demonstrate that an optimal Markov policy exists, from which the optimal history-dependent policy can be derived. Considering the easier implementation of the Markov policy, we adopt the greedy policy w.r.t $V^*(x, y)$ as the optimal policy.

We introduce Algorithm 4.1 to effectively solve the NCVaR optimization problem. This solution is equivalent to addressing the original problem incorporating an undetermined uncertain budget defined by the Radon-Nikodym derivative.

Algorithm 4.1 Value Iteration for NCVaR

1: **Initialization:** for each $x \in \mathcal{X}$ and $y \in \mathcal{Y}$, arbitrarily initialize $V_0(x, y)$.
2: **for** $t = 0, 1, 2, \ldots$ **do**
3: **for** $x \in \mathcal{X}$ and $y \in \mathcal{Y}$ **do**
4: $V_{t+1}(x, y) = \mathbf{T}[V_t](x, y)$
5: **end for**
6: **end for**
7: Set $V^*(x, y) = \lim_{t\to\infty} V_t(x, y)$, then construct π^* as the greedy policy w.r.t $V^*(x, y)$

However, implementing Algorithm 4.1 directly can be challenging due to the continuous nature of the set $\mathcal{Y}$. To address this issue, we employ a sampling approach, where we select multiple points in $\mathcal{Y}$ and subsequently utilize linear interpolation to derive the value function V. However, to guarantee convergence, we need to satisfy the following assumption for the initial value function V_0.

Assumption 7. The initial value function $V_0(x, y)$ is continuous and bounded in $y \in \mathcal{Y}$ for any $x \in \mathcal{X}$. Also, $yV_0(x, y)$ is concave in $y \in \mathcal{Y}$.

Let $N(x)$ denote the number of sample points, and $Y(x) = y_1, y_2, \ldots, y_{N(x)} \in [0, 1]^{N(x)}$ be the corresponding confidence level set. Notably, we have $y_1 = 0$ and $y_{N(x)} = 1$. To perform linear interpolation of $yV(x, y)$, we define the interpolation function $\mathcal{I}_x V$ as follows:

$$\mathcal{I}_x[V](y) = y_i V(x, y_i) + \frac{y_{i+1}V(x, y_{i+1}) - y_i V(x, y_i)}{y_{i+1} - y_i}(y - y_i), \tag{4.9}$$

where y_i and y_{i+1} are the closest points such that $y \in [y_i, y_{i+1}]$. With this, we introduce the interpolated Bellman operator for NCVaR, denoted as $\mathbf{T}_{\mathcal{I}} V$:

$$\mathbf{T}_{\mathcal{I}}[V](x,y) = \min_{a\in\mathcal{A}} \left[C(x,a) + \gamma \max_{\xi \in \mathcal{U}_{\text{NCVaR}}(y,P(\cdot|x,a))} \sum_{x'\in\mathcal{X}} \frac{\mathcal{I}_{x'}[V](y\xi(x'))}{y} P(x'|x,a) \right]. \tag{4.10}$$

An essential observation is that the interpolated Bellman operator shares the properties of Lemma 2, which can be shown using a similar proof. Additionally, Algorithm 4.2 provides a more practical value iteration method, utilizing the interpolated Bellman operator and linear interpolation to achieve near-optimal value functions and policies.

Algorithm 4.2 NCVaR Value Iteration with Linear Interpolation

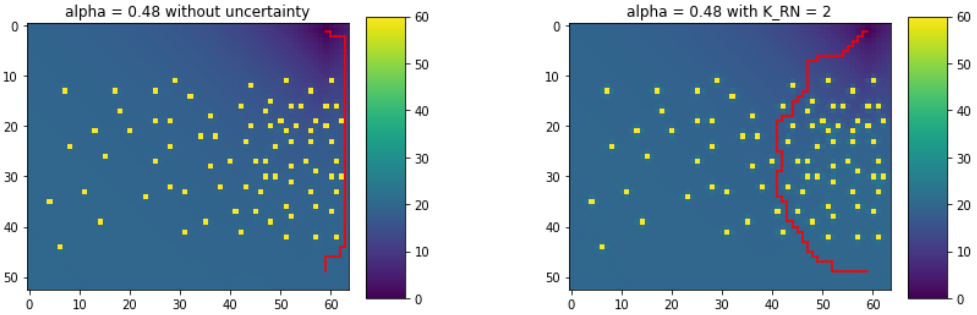
Initialization: choose $Y(x)$, $V_0(x,y)$ satisfying Assumption 7
for $t = 0, 1, 2, \ldots$ **do**
  for $x \in \mathcal{X}$ and $y \in \mathcal{Y}$ **do**
    $V_{t+1}(x,y) = \mathbf{T}_{\mathcal{I}}[V_t](x,y)$
  end for
end for
Set $V^*(x,y) = \lim_{t\to\infty} V_t(x,y)$, then construct π^* as the greedy policy w.r.t $V^*(x,y)$

4.5 Experiments

In this section, we use the grid world setup adopted for EVaR-VI in Chapter 2. Here, we omit a reintroduction of the setup and proceed directly to the results and analysis.

We first validate our approach for a fixed uncertain budget using Radon-Nikodym derivative and KL divergence. This involves visualizing the optimal value function with color variations (a bluer color indicates a lower risk while a yellower color indicates a higher risk) and tracing the optimal path as a red line (Figure 4.1a). In Figure 4.1a, 4.1b and 4.1c, we select a confidence level of $\alpha = 0.48$ and an uncertain budget of $K = 2$ for both RN derivative and KL divergence. Consequently, we obtain $\alpha'_{\text{CVaR}} = 0.24$ and $\alpha'_{\text{EVaR}} = 0.03$, which indicates that the new optimal policy will exhibit a more risk-averse behavior compared to the original one. Accordingly, the

optimal path becomes longer and is positioned closer to obstacles, aligning with the result that the value function is larger. We further assess Algorithm 4.2 for decision-dependent cases, setting the uncertain budget range to $[1, 2]$. As a result, for a fixed current state x, the new confidence level on the right side of the decomposition theorem significantly deviates from the fixed case. This increased deviation leads to the agent becoming more risk-averse as shown in Figure 4.1d. In conclusion, our algorithms effectively induce risk-averse policies, equipping agents to navigate more cautiously in uncertain environments. The experiments validate our methodology's efficacy in guiding agents towards safer decision-making strategies.



(a) $\alpha = 0.48$, no uncertainty

(b) $\alpha = 0.48$, $K_{\text{RN}} = 2$

(c) $\alpha = 0.48$, $K_{\text{KL}} = 2$

(d) $\alpha = 0.48$, $K_{\text{unfix}} \in [1, 2]$

Figure 4.1: Optimal value function and path in robust CVaR optimization across various uncertainty sets.

4.6 Conclusion

In this chapter, we have conducted a comprehensive and novel analysis of robust CVaR-based risk-sensitive RL within the framework of RMDP. We have successfully addressed robust CVaR

optimization in the presence of fixed uncertain budgets while adopting a rectangular ambiguity set. We have introduced a novel risk measure NCVaR and devised NCVaR value iteration algorithms to solve the challenges associated with state-action dependent uncertainty. Furthermore, we have demonstrated the convergence of our algorithms through theoretical analysis. We have validated the proposed approaches through simulation experiments, and the results showcased the effectiveness and practicality of our methods.

Chapter 5

Risk-Sensitive Reward-Free Reinforcement Learning

5.1 Introduction

In RL, agents learn optimal actions by iteratively interacting with the environment and leveraging feedback from reward signals. A critical part of this learning process is exploration, where agents navigate through states to effectively gather environment information. Despite exploration being widely recognized as a vital aspect of RL, simple randomized exploration strategies often fail due to high sample complexity [60]. While research by [6, 26, 27, 50] demonstrates that stochastic exploration can be sample-efficient, applying these algorithms across different reward functions can lead to inefficiencies. To address this, [51] introduces the concept of reward-free RL, in which the goal is to approximate the near optimal policy under any reward function after a single phase of exploration, enhancing the efficiency and adaptability of the learning process. [51] also derives upper and lower bounds of the sample complexity of the risk-free approach.

Building on these insights, subsequent studies such as [21, 53, 67, 68, 100, 107] have sought tighter upper bounds and more practical algorithms. The focus of these existing reward-free RL research has been predominantly on the risk-neutral approach, in which the goal is to maximize

the average total (discounted) reward.

In risk-sensitive RL, the objective function is shaped by applying risk measures to reward functions [8, 28, 32, 58, 84, 90, 104], thus is also significantly dependent on the exploration phase. However, efficient exploration, particularly in contexts without a predefined reward function, remains an under-explored area. Existing studies on sample complexity and algorithm performance in risk-sensitive RL typically target specific reward functions, potentially limiting their effectiveness in varied reward settings [7, 30, 31, 34, 99]. This situation underscores the urgency for developing efficient exploration methods in risk-sensitive RL, crucial for its practical deployment and success in diverse stochastic environments.

In this chapter, we study risk-sensitive RL in the reward-free setting, and aim to answer the following question:

Is it possible to design provably efficient risk-sensitive reward-free RL algorithm?

In this chapter, we design an algorithm with near-optimal sample complexity to the above question.

Chapter Contribution: This chapter introduces a CVaR-based risk-sensitive reward-free RL framework (CVaR-RF RL). For the exploration phase, we propose CVaR-RF-UCRL to efficiently explore environments with unknown reward functions. The number of trajectories collected in the exploration phase is upper bounded by $\tilde{\mathcal{O}}\left(\frac{S^2AH^4}{\epsilon^2\alpha^2}\right)$, where S is the number of states, A is the action count, H is the horizon length, ϵ is the targeted accuracy, and α the risk tolerance level for CVaR. We also prove a lower bound of $\Omega\left(\frac{S^2AH^2}{\epsilon^2\alpha}\right)$ for any CVaR-RF exploration algorithm. Subsequently, we introduce the CVaR-RF-planning algorithm equipped with CVaR-VI, which is able to solve CVaR RL for given reward function but without interacting with the environment. We also propose CVaR-VI-DISC, a discretized version of CVaR-VI for direct implementation in real-world settings while maintaining an optimization error within $\epsilon/3$. These developments ensure the efficiency and applicability of our CVaR-RF framework in advancing the field of risk-sensitive RL.

Challenges: 1). Compared to risk-neutral reward-free RL [51], CVaR-RF RL focuses only

on the tail distribution related to the risk tolerance parameter α. But in a reward-free setup, we can't access reward information, including the reward distribution. Therefore, we must adjust our exploration strategy based on α. To address this, we define an adaptive stopping rule for different α values during the exploration phase. Moreover, while the optimal policy in risk-neutral RL is Markovian, the optimal policy for risk-sensitive RL is history-dependent, which makes it more complex. To simplify this, we propose a planning algorithm with CVaR-VI that can construct a Markovian policy as the optimal policy for CVaR RL, reducing the added complexity.

2). Compared with CVaR RL [22–24, 43, 55, 70, 71, 95, 97], CVaR-RF RL faces challenges due to the absence of immediate feedback on risks associated with actions during the exploration phase. In CVaR RL, with rewards given, the agent doesn't need to explore every state or action, as it can immediately adjust its strategy based on the reward. However, in CVaR-RF RL, where rewards are unknown during the exploration, it's necessary to thoroughly explore the environment by visiting all possible states and actions. This extensive exploration gathers enough information for the planning phase, allowing the agent to adjust its strategy effectively. To facilitate this, we introduce CVaR-RF-UCRL, a method that efficiently explores all states.

Chapter Organization: In Section 5.2, we introduce the preliminaries essential for the understanding of CVaR-RF RL. Section 5.3 presents the formal problem statement of CVaR-RF RL. In Section 5.4, we present the CVaR-RF-UCRL for exploration and CVaR-RF-planning algorithms, and present the upper bound for sample complexity. Section 5.5 provides our analysis of the lower bound of sample complexity specifically for CVaR-RF exploration. Section 5.6 provides numerical examples. Section 5.7 offer concluding remarks.

5.2 Preliminaries

In this chapter, we use $\mathcal{S}$ to denote the state space and let S represent the number of states (to avoid notation conflicts with the random variable X). The probability of reaching state s under policy π is denoted by $P^{\pi}(s)$. Unlike the MDP process described in Section 1.1, this process begins with an

initial state s_1 selected from an unknown initial distribution $P_1(\cdot)$ and ends when the agent reaches the state s_{H+1}, where H denotes the time horizon length (distinct from the original setup). Here, we also consider a reward r, which can be interpreted as $-C$ and we assume $r(s,a) \in [0,1]$ for all (s,a).

Since the reward function is deterministic, its cumulative sum is bounded by $[0,H]$. Given this constraint and acknowledging that the optimal b aligns with the VaR (see Lemma 14), and considering $\text{VaR}_\alpha \in [0,H]$, we can appropriately restrict the range of b as follows:

$$\text{CVaR}_\alpha(X) := \sup_{b\in[0,H]} \left(b - \alpha^{-1}\mathbb{E}[(b-X)^+]\right). \tag{5.1}$$

Reward-Free RL: The RF-RL framework, as proposed by [51], is structured into two distinct phases: exploration and planning. In the exploration phase, the goal is to design algorithms that can efficiently explore the environment without reward information. Formally, in the exploration phase, each episode commences with an exploration policy π^t, based solely on data from previous episodes. An episode ξ_t captures a sequence of states and actions $(s_1^t, a_1^t, \ldots, s_H^t, a_H^t)$, starting at initial state s_1^t. Actions are chosen as $a_h^t = \pi_h^t(s_h^t)$, with subsequent states determined as $s_h^t \sim P_h(s_{h-1}^t, a_{h-1}^t)$. Each trajectory ξ_t is added to the dataset $\mathcal{D}_t$. Data collection ends at a random stopping time t_{stop}, resulting in dataset $D_{t_{\text{stop}}}$. Based on the dataset, we are able to get the empirical transition kernel $\hat{P}$.

In the planning phase, the agent's exploration strategy is critically assessed. During this phase, the agent is no longer permitted to interact with the environment. Instead, a specific reward function r is given, and the primary goal is to derive a near-optimal policy tailored to this r using the dataset $D_{t_{\text{stop}}}$ gathered during the exploration phase. The efficiency of the exploration approach is quantified based on the number of trajectories needed to consistently reach this objective, effectively measuring the algorithm's ability to prepare the agent for diverse reward scenarios without direct interaction with the MDP.

Our Goal: This chapter focuses on establishing an efficient CVaR based reward-free RL

framework, including:

1). Develop a CVaR-RF-Exploration algorithm that efficiently explores the environment without requiring any reward function and is adaptive to different α.

2). Propose a CVaR-RF-Planning algorithm, which computes near-optimal policies based on the dataset acquired during the exploration phase and a specified reward function, without further interaction with the environment.

3). Ensure the efficiency and reliability by analyzing the sample complexity of exploration algorithm and the optimization error of planning algorithm.

5.3 Problem Statement

To address the inner objective of CVaR outlined in (5.1), which depends on the variable b, we consider an augmented MDP, in which an augmented state is defined as $(s, b) \in \mathcal{S}^{\text{Aug}} := \mathcal{S} \times [0, H]$. The initial state for a given $b_1 \in [0, H]$ is set to (s_1, b_1). Then, for each timestep $h = 1, \ldots, H$, the agent selects action a_h based on policy π_h, and updates b_{h+1} to $b_h - r_h$.

For any history-dependent policy $\pi \in \Pi_H$, timestep $h \in [H]$, state $s_h \in \mathcal{S}$, budget $b_h \in [0, H]$, and history H, we define the value function as:

$$V_h^{\pi}(s_h, b_h; H_h) = \mathbb{E}_{\pi}\left[\left(b_h - \sum_{h'=h}^{H} r_{h'}(s_{h'}, a_{h'})\right)^{+} \middle| s_h, H_h\right].$$

The CVaR objective following policy π, starting at s_1, is then expressed as:

$$\text{CVaR}_{\alpha}^{\pi}(s_1) = \max_{b_1 \in [0,H]} \{b_1 - \alpha^{-1} V_1^{\pi}(s_1, b_1)\},$$

and the optimal CVaR objective is formulated as:

$$\begin{aligned} \text{CVaR}_\alpha^\star(s_1) &= \max_{\pi\in\Pi_H} \max_{b_1\in[0,H]} \{b - \alpha^{-1} V_1^\pi(s_1, b_1)\} \\ &= \max_{b_1\in[0,H]} \{b_1 - \alpha^{-1} \min_{\pi\in\Pi_H} V_1^\pi(s_1, b_1)\}. \end{aligned} \tag{5.2}$$

The work of [8] significantly advances our understanding by establishing the existence of an optimal policy $\rho^\star : \mathcal{S}^{\text{Aug}} \to \mathcal{A}$, which is deterministic and Markovian within the augmented MDP, denoted by $\mathcal{S}^{\text{Aug}} = \mathcal{S} \times [0, H]$. With a starting point of $b_1 \in [0, H]$ and initial state (s_1, b_1), the process unfolds as follows: for each $h = 1, 2, \ldots, H$, the action a_h is determined as $\rho^\star(s_h, b_h)$, the reward r_h as $r_h(s_h, a_h)$, the next state s_{h+1} evolves according to $P_h^\star(s_h, a_h)$, and the budget b_{h+1} is updated to $b_h - r_h$. The additional state b_h effectively tracks the residual budget from b_1, serving as a comprehensive summary of historical decisions for the CVaR RL problem.

The adoption of deterministic Markovian policies simplifies the decision-making process in MDPs, directly associating states with actions, thereby facilitating implementation and analytical processes. Consequently, without loss of optimality, the optimization problem in (5.2) simplifies to:

$$\text{CVaR}_\alpha^\star(s_1) = \max_{b_1\in[0,H]} \{b_1 - \alpha^{-1} \min_{\rho\in\Pi^{\text{Aug}}} V_1^\rho(s_1, b_1)\}, \tag{5.3}$$

where Π^{Aug} is the class of deterministic Markovian policies.

We now introduce the function definitions and the Bellman equations for the augmented MDP proposed in [10, 99]. For any policy $\rho \in \Pi^{\text{Aug}}$, we define:

$$V_h^\rho(s_h, b_h) = \mathbb{E}_\rho \left[\left(b_h - \sum_{h'=h}^{H} r_{h'}(s_{h'}, a_{h'}) \right)^+ \middle| s_h, b_h \right], \tag{5.4}$$

and

$$Q_h^\rho(s_h, b_h, a_h) = \mathbb{E}_\rho \left[\left(b_h - \sum_{h'=h}^{H} r_{h'}(s_{h'}, a_{h'}) \right)^+ \middle| s_h, b_h, a_h \right]. \tag{5.5}$$

For notation convenience, we introduce the following definition:

$$[P_h V_{h+1}]\,(s_h, b_h, a_h) = \mathbb{E}_{s_{h+1}\sim P(\cdot|s_h,a_h)}[V_{h+1}(s_{h+1}, b_{h+1})].$$

These functions adhere to the following Bellman equations:

$$\begin{aligned} V_h^\rho(s_h, b_h) &= \mathbb{E}_{a_h\sim\rho_h(s_h,b_h)}\left[Q_h^\rho(s_h, b_h, a_h)\right], \\ Q_h^\rho(s_h, b_h, a_h) &= [P_h V_{h+1}]\,(s_h, b_h, a_h), \end{aligned} \tag{5.6}$$

where $b_{h+1} = b_h - r_h$ and $V_{H+1}^\rho(s, b) = b_1^+ := \max(0, b_1)$. Similarly, we define the optimal conditions as:

$$\begin{aligned} V_h^\star(s_h, b_h) &= \min_{a\in\mathcal{A}} Q_h^\star(s_h, a_h, b_h), \\ \rho_h^\star(s_h, b_h) &= \text{argmin}_{a\in\mathcal{A}} Q_h^\star(s_h, b_h, a_h)], \\ Q_h^\star(s_h, b_h, a_h) &= \left[P_h V_{h+1}^\star\right](s_h, b_h, a_h), \end{aligned} \tag{5.7}$$

where $b_{h+1} = b_h - r_h$ and $V_{H+1}^\star(s, b) = b_1^+ = \max(0, b_1)$.

Here we introduce a key fact shown in [100], which shows the optimality of Π^{Aug}.

Theorem 10. (Optimality) For any $b_1 \in [0, 1]$, $V_1^\star(s_1, b_1) = V_1^{\rho^\star}(s_1, b_1) = \inf_{\pi\in\Pi_H} V_1^\pi(s_1, b_1)$.

Theorem 10 suggest that we could compute $V_1^\star$ and $\rho^\star$ using dynamic programming (DP) if the true transitions P were known, following the classical Value Iteration procedure in standard RL. By executing $\rho^\star$ starting from $(s_1, b_1^\star)$ where $b_1^\star := \arg\max_{b_1\in[0,H]}\{b_1 - \alpha^{-1}V_1^\star(s_1, b_1)\}$, we can attain the optimal CVaR value.

Based on these arguments, the goal of our work is to identify a probably approximately correct (PAC) algorithm for CVaR-RF RL, characterized by specific performance and accuracy parameters (ϵ, δ), which is defined as follows:

Definition 9. (PAC algorithm for CVaR-RF) A CVaR-RF exploration algorithm is (ϵ, δ)-PAC with

a given risk tolerance α if for any reward function r,

$$P\left(\mathbb{E}_{s_1\sim P_1}\left[\text{CVaR}_\alpha^\star(s_1;r)-\text{CVaR}_\alpha^{\hat{\rho}}(s_1;r)\right]\leq\epsilon\right)\geq 1-\delta.$$

Here, $\text{CVaR}_\alpha^\star(s_1;r)$ is derived by executing optimal policy $\rho^\star$ starting from $(s_1,b_1^\star)$ under the true transition probabilities P and the reward function r with optimal initial budget $b_1^\star := \arg\max_{b_1\in[0,H]}\{b_1-\alpha^{-1}V_1^\star(s_1,b_1;r)\}$. Conversely, $\text{CVaR}_\alpha^{\hat{\rho}}(s_1;r)$ is derived by executing the output policy in the planning phase $\hat{\rho}$ starting from $(s_1,\hat{b}_1)$ under the empirical transition probabilities $\hat{P}$ and the reward function r with the near optimal initial budget obtained in the planning phase.

5.4 Main Results

In this section, we first analyze the exploration phase by assuming the optimization error during the planning phase is bounded. Inspired by [35, 53], we propose the CVaR-RF-UCRL, which is an (ϵ,δ)-PAC algorithm for CVaR-RF exploration, with the sample complexity upper bounded by $\tilde{\mathcal{O}}(S^2AH^4/\epsilon^2\alpha^2)$. Then, in the planning phase, we propose a CVaR-RF-planning algorithm, adopting CVaR-VI and CVaR-VI-DISC, which satisfy the optimization error assumption.

Notation: Consider (s_h^i,a_h^i,s_{h+1}^i) as the state, action, and next state observed by an algorithm at step h of episode i. For any step $h\in[H]$ and any state-action pair $(s,a)\in\mathcal{S}\times\mathcal{A}$, we define $n_h^t(s,a)=\sum_{i=1}^t\mathbb{I}\{(s_h^i,a_h^i)=(s,a)\}$ as the count of visits to the state-action pair (s,a) at step h in the first t episodes, and $n_h^t(s,a,s')=\sum_{i=1}^t\mathbb{I}\{(s_h^i,a_h^i,s_{h+1}^i)=(s,a,s')\}$. The empirical transitions are defined as:

$$\hat{P}_h^t(s'|s,a)=\begin{cases}\frac{n_h^t(s,a,s')}{n_h^t(s,a)}, & \text{if } n_h^t(s,a)>0\\ \frac{1}{S}, & \text{otherwise}\end{cases}.$$

We denote by $\hat{V}_h^{t,\rho}(s_h,b_h;r)$ and $\hat{Q}_h^{t,\rho}(s_h,b_h,a_h;r)$ the value and action-value functions of a policy π in the MDP with transition kernels $\hat{P}_t$ and reward function r.

5.4.1 Exploration Phase

We first present a lemma that will be useful for the study of the objective within the CVaR-RF exploration context. Prior to delving into this lemma, we make an assumption regarding the planning phase.

Assumption 8. The optimization error during the planning phase is bounded, i.e.,

$$\left|\widehat{\text{CVaR}}_{\alpha}^{\hat{\rho}^\star}(s_1; r) - \widehat{\text{CVaR}}_{\alpha}^{\hat{\rho}}(s_1; r)\right| \leq \epsilon\alpha/3,$$

where $\widehat{\text{CVaR}}_{\alpha}^{\hat{\rho}^\star}(s_1; r)$ is derived by executing the optimal policy $\hat{\rho}^\star$ starting from $(s_1, \hat{b}_1^\star)$ under the empirical transition probabilities $\hat{P}$ and the reward function r with optimal initial budget $\hat{b}_1^\star := \arg\max_{b_1 \in [0,H]}\{b_1 - \alpha^{-1}\hat{V}_1^\star(s_1, b_1; r)\}$.

Notice that, Assumption 8 focuses on the optimization error based on same empirical transition probability $\hat{P}$ and given r. This assumption is not about the error with respect to the optimal policy for the true underlying MDP. Theorem 10 justifies the existence of an optimal policy $\hat{\rho}^*$ for MDP specified by $\hat{P}$ and given reward function (more technical details could be found in Appendix D.1.1). Furthermore, there exist many CVaR RL works capable of generating such a near-optimal policy $\hat{\rho}$ that satisfies this assumption, such as [24, 97, 99]. We also propose our algorithms in the planning phase that satisfy this assumption. Therefore, Assumption 8 could be immediately satisfied based on these facts.

The following lemma is useful for subsequent discussions and analyses related to our primary objective.

Lemma 3. An algorithm is (ϵ, δ)-PAC for CVaR-RF exploration with a given risk tolerance α if for any reward function r and for any $b_1 \in [0, H]$, $\left|V_1^\rho(s_1, b_1; r) - \hat{V}_1^\rho(s_1, b_1; r)\right| \leq \epsilon\alpha/3$.

Proof. Please refer to Appendix D.1.1 for more details. □

For simplifying the exposition of our algorithm, we posit that the initial state distribution P_0 is supported solely on a singular state s_1. This assumption incurs no loss of generality [35]. If this is

not the case, one might contemplate an augmented MDP that includes an additional initial state s_0, paired with a unique action a_0 yielding a null reward. Thus, $b_0 = b_1$. In this scenario, the transitions from s_0 using a_0 are defined as $P_0(\cdot|s_0, a_0) = P_0$.

The error upper bounds in the CVaR-RF-UCRL algorithm are derived from an upper bound on the estimation error for each policy ρ, each initial budget $b \in [0, H]$ and each reward function r. The complete procedure is outlined in Algorithm 5.1. Before discussing the details of this algorithm, we introduce the definitions for the estimation error and its upper confidence bound.

Definition 10. For a given policy ρ, reward function r, and episode t, we define this error for any $(s_h, b_h, a_h) \in \mathcal{S}^{\text{Aug}} \times \mathcal{A}$ as

$$\hat{e}_h^{t,\rho}(s_h, b_h, a_h; r) := \left| \hat{Q}_h^{t,\rho}(s_h, b_h, a_h; r) - Q_h^{\rho}(s_h, b_h, a_h; r) \right|.$$

Definition 11. The upper confidence bound $E_h^t(s_h, a_h)$ for the error, recursively defined as follows: $E_{H+1}^t(s,a) = 0$ for all (s, a), and for all $h \in [H]$, with the convention $\frac{1}{0} = +\infty$,

$$E_h^t(s_h, a_h) = \min \left\{ H, H \sqrt{\frac{2\beta(n_h^t(s,a), \delta)}{n_h^t(s,a)}} + \sum_{s'} \hat{P}_h^t(s'|s,a) \max_a E_{h+1}^t(s', a) \right\}, \tag{5.8}$$

where $\beta(n, \delta)$ is a threshold function, an input to the algorithm, the choice of which will be discussed later.

It is important to note that the error upper bound only depends on the state s and action a, and is independent of the policy ρ, initial budget b_1 and reward function r. Lemma 4 establishes that $E_h^t(s,a)$ serves as the upper bound on the error $\hat{e}_h^{t,\rho}(s, b, a; r)$ for any ρ, b, r with a high probability.

Lemma 4. With the KL divergence between two distributions over $\mathcal{S}$ defined as $\text{KL}(p \parallel q) = \sum_{s \in \mathcal{S}} p(s) \log \frac{p(s)}{q(s)}$, consider the event

$$\mathcal{E} = \left\{ \forall t \in \mathbb{N}, \forall h \in [H], \forall (s,a), \text{KL}(\hat{P}_h^t(\cdot|s,a), P^h(\cdot|s,a)) \leq \frac{\beta(n_h^t(s,a), \delta)}{n_h^t(s,a)} \right\},$$

it is established that for any policy ρ, any reward function r and any b, $\hat{e}_h^{t,\rho}(s,b,a;r) \leq E_h^t(s,a)$ holds on event $\mathcal{E}$.

Proof. Please refer to the Appendix D.1.2 for more details. □

We now introduce the proposed CVaR-RF-UCRL algorithm, which operates on the principle of uniformly reducing the estimation error across all budgets, policies and potential reward functions by adopting a greedy approach based on the upper bounds E^t on these errors. The stopping criterion for CVaR-RF-UCRL is defined as reaching an error in step 1 that is smaller than $\epsilon\alpha/3$:

- **Sampling rule:** the exploration policy π^{t+1} is the greedy policy with respect to $E^t(s,a)$, such that for all $s \in \mathcal{S}$ and $h \in [H]$:

$$\pi_h^{t+1}(s_h) = \text{argmax}_a E_h^t(s,a). \tag{5.9}$$

- **Stopping rule:** the algorithm stops at

$$t_{\text{stop}} = \inf\{t : E_h^t(s_1, \pi_1^{t+1}(s_1)) \leq \epsilon\alpha/3\}.$$

Algorithm 5.1 CVaR-RF-UCRL

Given: risk tolerance $\alpha \in (0,1]$
Initialization: $t=1$, $\mathcal{D}_0 = \emptyset$, initialize E^0 with (5.8) and π_h^1 with (5.9)
while $E_h^{t-1}(s_1, \pi_1^t(s_1)) \geq \epsilon\alpha/3$ **do**
  Observe the initial state $s_1^t \sim P_0$
  for $h = 1, \ldots, H-1, H$ **do**
    Play $a_h^t \sim \pi_h^t(s_h^t)$
    Observe the next state s_{h+1}
  end for
  Compute E^t according to (5.8) and π^{t+1} according to (5.9)
  $D_t = D_{t-1} \cup (s_1^t, a_1^t, \ldots, s_H^t, a_H^t)$
  $t = t+1$
end while
Return the dataset $\mathcal{D}_{t_{\text{stop}}}$

Now, we have the following Lemma showing that CVaR-RF-UCRL is an algorithm with (ϵ, δ)-PAC.

Lemma 5. (Correctness) On the event $\mathcal{E}$, given α, for any r, ρ and b_1,

$$\left|V_1^{t_{\text{stop}},\rho}(s_1, b_1; r) - \hat{V}_1^{t_{\text{stop}},\rho}(s_1, b_1; r)\right| \leq \epsilon\alpha/3,$$

which implies $\text{CVaR}_\alpha^\star(s_1; r) - \text{CVaR}_\alpha^{\hat{\rho}^\star}(s_1; r) \leq \epsilon$.

Proof. By definition of the stopping rule and the sampling rule, we have for all $a \in \mathcal{A}$, $E_1^{t_{\text{stop}}}(s_1, a) \leq \epsilon/3$. Hence, by Lemma 4 on the event $\mathcal{E}$, for all ρ, b_1, r, and all a, $\hat{e}_1^{t_{\text{stop}},\rho}(s_1, b_1, a; r) \leq \epsilon\alpha/3$. In particular, for all ρ, b_1, and r, $\left|V_1^{t_{\text{stop}},\rho}(s_1, b_1; r) - \hat{V}_1^{t_{\text{stop}},\rho}(s_1, b_1; r)\right| \leq \epsilon\alpha/3$. The conclusion follows from Lemma 3 by choosing ρ to be $\hat{\rho}^\star$. □

We are now able to state our main results for CVaR-RF-UCRL, which show that with a proper chosen threshold $\beta(n, \delta)$, CVaR-RF-UCRL achieves (ϵ, δ)-PAC for CVaR RL. Furthermore, an upper bound on its sample complexity can be established under these conditions.

Theorem 11. (Upper Bound for Sample Complexity) Using threshold $\beta(n, \delta) = \log(2SAH/\delta) + (S-1)\log(e(1 + n/(S-1)))$, the CVaR-RF-UCRL is (ϵ, δ)-PAC for CVaR-RF exploration. The number of trajectories collected in the exploration phase is bounded by $\tilde{\mathcal{O}}\left(\frac{S^2AH^4}{\epsilon^2\alpha^2}\right)$.

Proof. Please refer to Appendix D.1.3 for more details. □

Compared with the risk-neutral reward-free approaches [51, 53, 67], the denominator of the bound we obtained is related to the risk tolerance parameter α. This is expected since CVaR is interpreted as the mean of the tail distribution with an area under the curve equal to α, it requires more trajectories for smaller α values and fewer trajectories for larger α values.

5.4.2 Planning Phase

In the planning phase, the reward function is provided, and the goal is to find a near-optimal policy based on the given reward function and the dataset generated during the exploration phase.

Following a similar approach to [51], we now introduce our planning algorithm, as outlined in Algorithm 5.2.

Algorithm 5.2 CVaR-RF-Planning

Input: a dataset of transition $\mathcal{D}_{t_{\text{stop}}}$, reward function r, accuracy ϵ, risk tolerance α.
for all $(s, a, s', h) \in \mathcal{S} \times \mathcal{A} \times \mathcal{S} \times [H]$ **do**
  $N_h(s, a, s') \leftarrow \sum_{(s_h, a_h, s_{h+1}) \in \mathcal{D}} \mathbb{I}[s_h = s, a_h = a, s_{h+1} = s']$.
  $N_h(s, a) \leftarrow \sum_{s'} N_h(s, a, s')$.
  $\hat{P}_h(s'|s, a) = N_h(s, a, s')/N_h(s, a)$.
end for
$\hat{\rho}, \hat{b} \leftarrow$ APPROXIMATE-CVaR-SOLVER$(\hat{P}, r, \epsilon, \alpha)$.
return policy $\hat{\rho}$, and initial budget $\hat{b}$.

In Algorithm 5.2 , we first compute the empirical transition matrix $\hat{P}$ based on the collected dataset $\mathcal{D}_{t_{\text{stop}}}$. Then, for each reward function r, we find a near-optimal policy by employing a APPROXIMATE-CVaR-SOLVER that utilizes transitions $\hat{P}$, the given reward function r, an accuracy parameter ϵ and the given risk tolerance α. It's worth noting that the solver can be any algorithm designed to find an $\epsilon/3$-suboptimal policy $\hat{\rho}$ for CVaR RL when both the transition matrix and the reward are known. One straightforward approach to achieve this is by using the Value Iteration algorithm, which iteratively solves the Bellman optimality equation (5.6) in a dynamic programming manner. The greedy policy induced by the resulting $Q^\star$ yields the optimal optimal policy without errors. We present Algorithm 5.3, which generates an optimal policy exactly according to Theorem 10 [99]. This algorithm satisfies our Assumption 8 about the optimization error.

Discretization

Algorithm 5.3 faces computational challenges due to the dynamic programming step, which requires optimization over all $b \in [0, H]$, involving the maximization of a non-concave function [99]. Following the approach proposed in [7,99], we introduce a discretization of rewards, which allows the mentioned steps to be performed over a finite grid. This offers computational efficiency while preserving the same statistical guarantees.

Algorithm 5.3 CVaR-VI

Input: transition matrix P, reward function r, risk tolerance α
for all $s \in \mathcal{S}, b \in [0, H]$ **do**
  Set $V_{H+1}(s, b) = b^+$
  for $h = H, H-1, \ldots, 1$ **do**
    $Q_h(s_h, b_h, a_h) = [P_h V_{h+1}](s_h, b_h, a_h)$, where $b_{h+1} = b_h - r_h$
    $\rho_h^\star(s_h, b_h) = \text{argmin}_a Q_h(s_h, b_h, a_h)$
    $V_h^\star(s_h, b_h) = \min_a Q_h(s_h, b_h, a_h)$
  end for
end for
Calculate $b^\star = \text{argmax}_{b_1 \in [0,1]} \{b - \alpha^{-1} V_1(s_1, b)\}$
return policy $\rho^\star$ and $b^\star$

We fix a precision $\eta \in (0, 1)$ and define $\phi(r) = \eta\lceil r/\eta \rceil \wedge 1$. This rounding function maps $r \in [0, 1]$ to an η-net of the interval $[0, 1]$, commonly referred as the grid. The discretized MDP dis($\mathcal{M}$) is an exact replica of the true MDP $\mathcal{M}$ with one exception: its rewards are post-processed using ϕ, i.e., $r(s, a; \text{disc}(\mathcal{M})) = \phi(r(s, a; \mathcal{M}))$. We now introduce the CVaR value iteration with discretization algorithm.

Algorithm 5.4 CVaR-VI-DISC

Input: transition matrix P, reward function r, precision parameter η, risk tolerance α.
Discretize the reward funtion r by

$$\hat{r} = \phi(r) = \eta\lceil r/\eta \rceil \wedge 1$$

for all $s \in \mathcal{S}, \hat{b} = n \cdot \eta, n = 0, 1, \ldots$ **do**
  Set $\hat{V}_{H+1}(s, \hat{b}) = \hat{b}^+ := \max(0, \hat{b})$
  for $h = H, H-1, \ldots, 1$ **do**
    $\hat{Q}_h(s_h, \hat{b}_h, a_h) = \left[P_h \hat{V}_{h+1}\right](s_h, \hat{b}_h, a_h)$, where $\hat{b}_{h+1} = \hat{b}_h - \hat{r}_h$
    $\hat{\rho}_h^\star(s_h, \hat{b}_h) = \text{argmin}_a \hat{Q}_h(s_h, \hat{b}_h, a_h)$
    $\hat{V}_h^\star(s_h, \hat{b}_h) = \min_a \hat{Q}_h(s_h, \hat{b}_h, a_h)$
  end for
end for
Calculate $\hat{b}^\star = \text{argmax}_{\hat{b}} \left\{\hat{b} - \alpha^{-1} \hat{V}_1(s_1, \hat{b})\right\}$
return policy $\hat{\rho}^\star$ and $\hat{b}^\star$

Computational Complexity

In disc($\mathcal{M}$), the α-th quantile of the returns distribution (the argmax of the CVaR objective) will be a multiple of η. Therefore, it suffices to compute $V_1(s_1, b_1)$ and maximize line 9 over the grid. Since b_1 transitions by subtracting rewards, which are multiples of η, b_h will always stay on the grid. Hence, the entire dynamic programming procedure only needs to occur on the grid. This approach demonstrates that CVaR value iteration via discretization is computationally tractable.

Theorem 12. The CVaR-VI-DISC has a run time of $\mathcal{O}(S^2AH\eta^{-2})$ in the discretized MDP. Setting $\eta = \epsilon\alpha/3H$, as suggested in Theorem 13, the run time is $\mathcal{O}(\frac{S^2AH^3}{\epsilon^2\alpha^2})$.

Proof. Please refer to Appendix for more details. □

Discretization Error

Next, we evaluate the impact of errors resulting from the discretization step. Following a similar method as previous works [99], we can relate the errors within disc($\mathcal{M}$) to equivalent errors within $\mathcal{M}$ using a coupling argument. This leads us to introduce the CVaR-VI-DISC algorithm, which is tailored for practical applications.

The following theorem guarantees that the optimization error assumption is met when when Algorithm 5.4 is employed.

Theorem 13. By selecting $\eta \leq \epsilon\alpha/3H$, we ensure that

$$\left|\text{CVaR}_\alpha^{\rho^\star}(s_1; r) - \text{CVaR}_\alpha^{\hat{\rho}}(s_1; r)\right| \leq \epsilon/3, \tag{5.10}$$

where $\rho^\star$ represents the policy generated by Algorithm 5.3 and $\hat{\rho}$ is the output of Algorithm 5.4. Consequently, the optimization error is bounded by $\epsilon/3$, which satisfies Assumption 8.

Proof. Please refer to the Appendix for more details. □

5.4.3 Adaptability to Varying Risk Tolerances

We further introduce an important proposition that underscores the adaptability of our exploration process to different levels of risk tolerance α:

Proposition 3. For any $\alpha' \geq \alpha$, the exploration dataset obtained through Algorithm 5.1 at risk tolerance α contains the requisite information for conducting CVaR-RF RL with any higher risk tolerance α'. Consequently, the planning phase is also compatible with any given $\alpha' \geq \alpha$.

Proof. Utilizing Lemma 3, we observe that as $\epsilon\alpha/3 \leq \epsilon\alpha'/3$, the CVaR-RF exploration algorithm configured with a risk tolerance of α also satisfies the (ϵ, δ)-PAC criterion for CVaR-RF RL when operating under a higher risk tolerance $\alpha' \geq \alpha$. Furthermore, invoking Theorem 13, we have that the stipulated optimization error condition is met since $\eta \leq \eta'$. This implies that the planning phase remains efficacious under these adjusted parameters. □

5.5 Lower Bound

In this section, we develop a lower bound of the sample complexity for CVaR-RF exploration. We present a theorem that delineates this lower bound, applicable to any algorithm operating within the CVaR-RF exploration framework.

Theorem 14. Consider a universal constant $C > 0$. For a given risk tolerance $\alpha \in (0, 1]$, if the number of actions $A \geq 2$, the number of states $S \geq C \log_2 A + 2$, the horizon $H \geq C \log_2 S + 1$, and the accuracy parameter $\epsilon \leq \min\{1/4\alpha, H/48\alpha\}$, then any CVaR-RF exploration algorithm that can output ϵ-optimal policies for an arbitrary number of adaptively chosen reward functions with a success probability $\delta = 1/2$ must collect at least $\Omega(S^2AH^2/\alpha\epsilon^2)$ trajectories in expectation.

Proof Sketch 2. Here we highlight the main idea of our lower bound proof, while the detailed proof can be found in the Appendix. Our proof is inspired by the lower bound construction in for the reward-free RL [51]. The key idea is that any reward-free risk neutral problem can be transformed into a CVaR-RF RL problem. If a CVaR-RF exploration algorithm that can output ϵ-optimal

policies in the transformed CVaR-RF RL problem, it can also solve the original reward-free risk neutral problem. Specifically, for a MDP $\mathcal{M}$ with initial state s_1, we consider a new MDP $\mathcal{M}'$ with an initial state s_0. For any action a, $P(s_1|s_0, a) = \alpha$, $P(s'|s_0, a) = 1 - \alpha$, $P(s'|s', a) = 1$, and $r(s', a) = 1$. For any adaptively chosen reward function for $\mathcal{M}$ and a policy π, the CVaR with tolerance α following policy in the new MDP $\mathcal{M}'$ is equal to the cumulative rewards in the original MDP $\mathcal{M}$. [51] shows that any reward-free exploration algorithm that output ϵ-optimal policy from initial state s_1 must collect at least $\Omega(S^2AH^2/\alpha\epsilon^2)$ trajectories in expectation. Thus, from the initial state s_0, the CVaR-RF exploration algorithm must collect at least $\Omega(S^2AH^2/\alpha\epsilon^2)$ trajectories in expectation. □

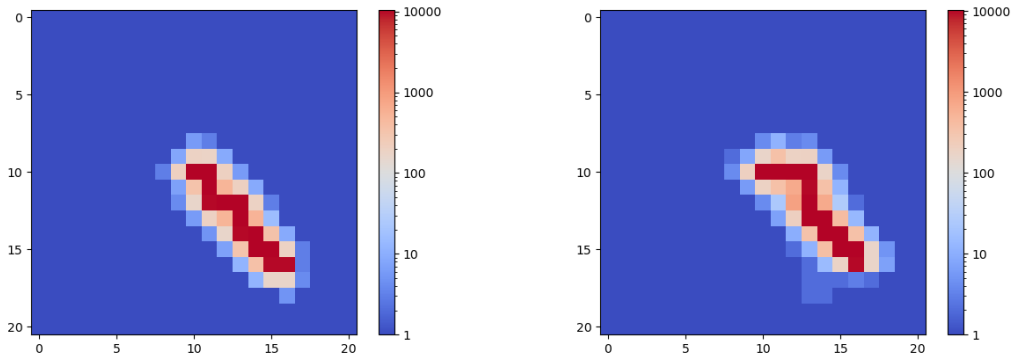
This theorem illustrates that, compared with the lower bound, the upper bound established in Theorem 11 has by an additional factor of H^2 and $1/\alpha$, while being tight with respect to the parameters S, A, ϵ. If α is a constant, our result is nearly minimax-optimal with an additional factor on H^2. An interesting direction of the future work is utilizing the empirical Bernstein inequality to further improve the sample complexity. The H factor can potentially be optimized by adopting an approach similar to [67] by introducing an empirical Bernstein inequality derived from a control of the transition probability. As shown in [99], the Bernstein inequality could also potentially improve the dependence on α under a continuity assumption. Furthermore, compared with the risk-neutral reward-free RL, our derived lower bound for any CVaR-RF exploration algorithm includes an additional α in the denominator. This is because CVaR focuses on the α worst outcomes. Additionally, the CVaR setting poses challenges due to non-Markovianity, requiring more efforts in achieving a minimax optimal sample complexity bound.

5.6 Experiments

In this section, we provide numerical examples to evaluate the proposed CVaR-RF RL framework. In these examples, we use similar experimental setup as in [53]. Our environment is configured as a grid-world consisting of 21×21 states, where each state offers four possible actions (up,

down, left, right), and actions leading to the boundary result in remaining in the current state. The agent will move to the correct state with a probability of 0.95. However, there is an equal probability of $\frac{0.05}{3}$ for the agent to move in any one of the other three directions. Initially, the exploration algorithm CVaR-RF-UCRL runs without reward information, collecting $n = 30,000$ transitions. The empirical transition probability $\hat{P}$ is then estimated. We use the $\beta(n, \delta)$ threshold from Theorem 11 with $\delta = 0.1$ and set a time horizon H of 20. Using the obtained dataset and $\hat{P}$, the planning algorithm derives near-optimal policies, employing CVaR-VI-DISC as the solver.

Reward Setup 1: The first one is similar with [53], where the agent starts at position $(10, 10)$. The reward structure is primarily set at 0 for most states, except at $(16, 16)$ where it is 1.0. Here we choose $\epsilon = 0.1$. Then we executing the output policy of CVaR-VI-DISC in the same grid-world for $K = 10,000$ trajectories and plot the number of state visits following the policy. For comparison, we also generate the optimal policy using true transition probability. Figures 5.1a displays the number of visits to each state following the policy generated from P, while Figure 5.1b shows for $\hat{P}$. Additionally, Table 5.1 presents the CVaR values under both true and empirical transition probabilities.



(a) Optimal policy (b) CVaR-VI-DISC

Figure 5.1: Number of state visits following policies generated under P and $\hat{P}$ in reward setup 1 with risk tolerance $\alpha = 0.05$.

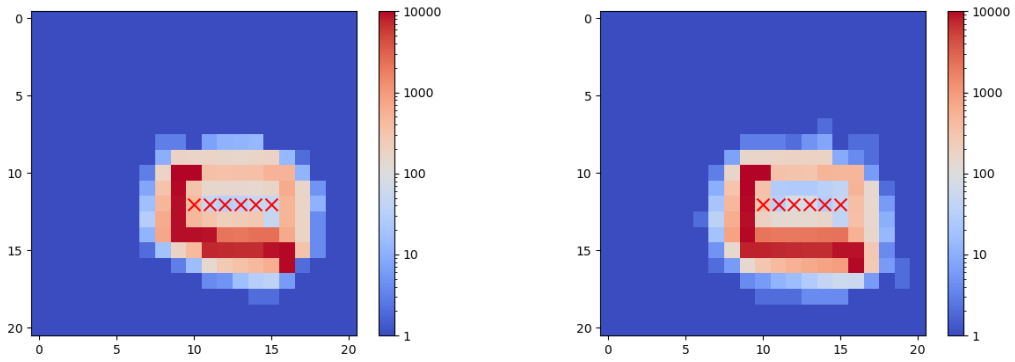
These visitation patterns, shown in Figures 5.1a and 5.1b, are notably similar, indicating that the agent tends to favor states with higher rewards. This behavior is consistent with the objective of

ϵ, α	CVaR_P	$\text{CVaR}_{\hat{P}}$	Error
$0.1, 0.05$	4.308	4.258	0.05
$0.1, 0.95$	4.960	4.954	0.006

Table 5.1: CVaR values under reward setup 1 with different α.

maximizing CVaR. The similarity in patterns under both true and empirical transition probabilities underscores the reliability of the data collected during the exploration phase. Moreover, with $\epsilon = 0.1$ and $\alpha = 0.05$, the difference between true and empirical CVaR is 0.05, which is below the anticipated error threshold of $\epsilon = 0.1$. Similarly, with $\epsilon = 0.1$ and $\alpha = 0.95$, the error is only 0.006, again less than the threshold of 0.1. These results align with our theoretical analysis.

Reward Setup 2: We consider a more complex case as the reward structure is primarily set at 0.5 for most states, except at $(16, 16)$ where it is 1.0, and a zero-reward zone marked 'x' from $(12, 10)$ to $(12, 16)$.



(a) Optimal policy (b) CVaR-VI-DISC

Figure 5.2: Number of state visits following policies generated under P and $\hat{P}$ in reward setup 2 with risk tolerance $\alpha = 0.05$.

ϵ, α	CVaR_P	$\text{CVaR}_{\hat{P}}$	Error
$0.1, 0.05$	1.852	1.829	0.023
$0.1, 0.95$	1.993	1.990	0.003

Table 5.2: CVaR values under reward setup 2 with different α.

Figure 5.2 and Table 5.2 illustrate that CVaR-RF RL effectively avoids traversing zero-reward

regions, and the observed errors remain within the pre-defined thresholds. These outcomes are also consistent with the CVaR's property as the agent is more risk-averse compared to risk-neutral case.

5.7 Conclusion

In this chapter, we have introduced a novel risk-sensitive reward-free RL framework based on CVaR, which is able to solve CVaR RL for given any reward function after a singular reward-free exploration phase. We have proposed CVaR-RF-UCRL as the exploration algorithm and established upper and lower bounds for the sample complexity. We have developed a CVaR-RF-planning algorithm, equipped with CVaR-VI and CVaR-VI-DISC to generate near-optimal Markov policies solely based on the exploration dataset and given reward function. Through our numerical experiments, we have validated the effectiveness and practicality of this CVaR-RF-RF framework.

Chapter 6

Conclusion

This thesis presents advancements in risk-sensitive RL, addressing critical challenges in managing uncertainty, optimizing for risk-sensitive criteria, and enhancing decision-making robustness. By leveraging novel risk measures, developing generalized frameworks, ensuring robustness in uncertain environments, and designing efficient exploration methods, this work contributes new approaches to foundational problems in risk-sensitive RL.

In Chapter 2, we introduced EVaR as a coherent, interpretable risk measure that offers computational advantages over traditional measures such as CVaR. We developed two algorithms—EVaR-VI and EVaR-PG—to solve the EVaR optimization problem, establishing the practical and theoretical viability of EVaR for RL applications. This approach enables RL agents to minimize risks using a measure that is both coherent and naturally interpretable through its dual KL-divergence representation.

Chapter 3 extended risk-sensitive RL by proposing a novel framework based on PhiD-R, a class of coherent risk measures that encompasses widely used risk measures, including CVaR and EVaR. By introducing a trajectory-based policy gradient method, we demonstrated how PhiD-R can be used to efficiently solve RL problems while providing flexibility in risk measure selection. This approach ensures that decision-makers can customize risk preferences to suit specific applications without sacrificing computational efficiency.

In Chapter 4, we focused on the robustness of risk-sensitive RL within the RMDP framework, examining CVaR's robustness under uncertainty. We developed algorithms to optimize for the worst-case CVaR within a predefined ambiguity set of transition probabilities and introduced NCVaR to handle decision-dependent uncertainty. These contributions reinforce the robustness of risk-sensitive RL under dynamic uncertainty conditions, making it a more effective tool for applications where model parameters are not fully known or are subject to change.

Finally, in Chapter 5, we addressed the problem of efficient exploration in risk-sensitive RL without a predefined reward function, developing a CVaR-based reward-free framework. By proposing the CVaR-RF-UCRL algorithm, which achieves near-optimal sample complexity, and additional planning algorithms tailored for CVaR RL, we demonstrated how risk-sensitive exploration can be conducted effectively in diverse, unknown environments. This contribution is particularly valuable for safety-critical applications, where efficient and safe exploration is essential.

Appendix A

Technical Results in Chapter 2

A.1 Proof of Lemma 1

The monotonicity and constant shift properties can be directly obtained from the definition of EVaR Bellman operator since $\sum_{x' \in \mathcal{X}} \xi(x') P(x'|x,a) = 1$ holds for any $\xi \in \mathcal{U}_{\text{EVaR}}(y, P(\cdot|x,a))$ and $\xi(x')P(x'|x,a)$ is non-negative.

For the contraction property, by the definition of sup norm, for any $x \in \mathcal{X}, y \in \mathcal{Y}$, we have

$$-||V_1 - V_2||_\infty \leq V_1(x,y) - V_2(x,y) \leq ||V_1 - V_2||_\infty.$$

Using the monotonicity and constant shift property, we obtain

$$-\gamma||V_1 - V_2||_\infty \leq \mathbf{T}[V_1](x,y) - \mathbf{T}[V_2](x,y) \leq \gamma||V_1 - V_2||_\infty.$$

This further implies that

$$|\mathbf{T}[V_1](x,y) - \mathbf{T}[V_2](x,y)| \leq \gamma||V_1 - V_2||_\infty$$

and the contraction property holds.

It remains to prove the concavity preserving property. Assume that $yV(x,y)$ is concave in $y \in \mathcal{Y}$. Let $y_1, y_2 \in \mathcal{Y}$ and $\lambda \in [0,1]$ and define $y_\lambda = (1-\lambda)y_1 + \lambda y_2$. Then,

$$
\begin{aligned}
&(1-\lambda)y_1 \mathbf{T}[V](x,y_1) + \lambda y_2 \mathbf{T}[V](x,y_2) \\
&= (1-\lambda)y_1 \min_{a_1 \in \mathcal{A}} \big[C(x,a_1) + \gamma \max_{\xi_1 \in \mathcal{U}_{\text{EVaR}}(y_1, P(\cdot|x,a_1))} \sum_{x' \in \mathcal{X}} \xi_1(x') V(x', y_1 \xi(x')) P(x'|x,a_1)\big] \\
&\qquad + \lambda y_2 \min_{a_2 \in \mathcal{A}} \big[C(x,a_2) + \gamma \max_{\xi_2 \in \mathcal{U}_{\text{EVaR}}(y_2, P(\cdot|x,a_2))} \sum_{x' \in \mathcal{X}} \xi_2(x') V(x', y_2 \xi(x')) P(x'|x,a_2)\big] \\
&\overset{(1)}{\leq} \min_{a \in \mathcal{A}} \big[y_\lambda C(x,a) + \gamma \max_{\substack{\xi_1 \in \mathcal{U}_{\text{EVaR}}(y_1, P(\cdot|x,a)) \\ \xi_2 \in \mathcal{U}_{\text{EVaR}}(y_2, P(\cdot|x,a))}} \sum_{x' \in \mathcal{X}} P(x'|x,a)\big((1-\lambda)y_1 \xi_1(x') V(x', y_1 \xi_1(x')) \\
&\qquad + \lambda y_2 \xi_2(x') V(x', y_2 \xi_2(x'))\big)\big] \\
&\overset{(2)}{\leq} \min_{a \in \mathcal{A}} \big[y_\lambda C(x,a) + \gamma \max_{\substack{\xi_1 \in \mathcal{U}_{\text{EVaR}}(y_1, P(\cdot|x,a)) \\ \xi_2 \in \mathcal{U}_{\text{EVaR}}(y_2, P(\cdot|x,a))}} \sum_{x' \in \mathcal{X}} P(x'|x,a)\big((1-\lambda)y_1 \xi_1(x') + \lambda y_2 \xi_2(x')) \\
&\qquad V(x, (1-\lambda)y_1 \xi_1(x') + \lambda y_2 \xi_2(x'))\big].
\end{aligned}
\tag{A.1}
$$

The inequality (1) is by the concavity of min and (2) is by the assumption of concavity of $yV(x,y)$. Now define

$$
\xi = \frac{(1-\lambda)y_1 \xi_1 + \lambda y_2 \xi_2}{y_\lambda} = \frac{(1-\lambda)y_1 \xi_1 + \lambda y_2 \xi_2}{(1-\lambda)y_1 + \lambda y_2}.
$$

To prove the the concavity preserving property, it remains to show that $\xi \in \mathcal{U}_{\text{EVaR}}(y_\lambda, P(\cdot|x,a))$. Note that $\xi_1 \in \mathcal{U}_{\text{EVaR}}(y_1, P(\cdot|x,a))$ and $\xi_2 \in \mathcal{U}_{\text{EVaR}}(y_2, P(\cdot|x,a))$, we obtain

$$
\textstyle\sum_{x' \in \mathcal{X}} \xi(x') P(x'|x,a) = \sum_{x' \in \mathcal{X}} \frac{(1-\lambda)y_1 \xi_1 + \lambda y_2 \xi_2}{(1-\lambda)y_1 + \lambda y_2} P(x'|x,a) = 1.
$$

It remains to show that

$$
\sum_{x' \in \mathcal{X}} \xi(x') P(x'|x,a) \log \xi(x') \leq -\ln y_\lambda.
$$

Recall that ξ is the ratio of two PMFs, then we have

$$
Q = \xi P = \frac{(1-\lambda)y_1 Q_1 + \lambda y_2 Q_2}{(1-\lambda)y_1 + \lambda y_2},
$$

where $Q_1 = \xi_1 P$ and $Q_2 = \xi_2 P$.

Then it is equivalent to show

$$D_{KL}(Q \parallel P) \leq -\ln y_\lambda.$$

Since KL divergence is convex when P is fixed, we have

$$\begin{aligned} D_{KL}(Q \parallel P) &= D_{KL}\left(\frac{(1-\lambda)y_1 Q_1 + \lambda y_2 Q_2}{(1-\lambda)y_1 + \lambda y_2} \parallel P\right) \\ &= D_{KL}\left(\frac{(1-\lambda)y_1}{(1-\lambda)y_1 + \lambda y_2} Q_1 + \frac{\lambda y_2}{(1-\lambda)y_1 + \lambda y_2} Q_2 \parallel P\right) \\ &\leq \frac{(1-\lambda)}{(1-\lambda)y_1 + \lambda y_2} y_1 D_{KL}(Q_1 \parallel P) + \frac{\lambda}{(1-\lambda)y_1 + \lambda y_2} y_2 D_{KL}(Q_2 \parallel P). \end{aligned}$$

Since $D_{KL}(Q_1 \parallel P) \leq -\ln y_1$ and $D_{KL}(Q_2 \parallel P) \leq -\ln y_2$, we obtain

$$\big((1-\lambda)y_1 + \lambda y_2\big) D_{KL}(Q \parallel P) \leq -\big((1-\lambda)y_1 \ln y_1 + \lambda y_2 \ln y_2\big).$$

We will also use the fact that $y \ln y$ is convex, i.e,

$$\big((1-\lambda)y_1 \ln y_1 + \lambda y_2 \ln y_2\big) \geq \big((1-\lambda)y_1 + \lambda y_2\big) \ln((1-\lambda)y_1 + \lambda y_2).$$

Combining these two inequalities, we can get

$$\big((1-\lambda)y_1 + \lambda y_2\big) D_{KL}(Q \parallel P) \leq -\big((1-\lambda)y_1 + \lambda y_2\big) \ln((1-\lambda)y_1 + \lambda y_2),$$

i.e,

$$D_{KL}(Q \parallel P) \leq -\ln((1-\lambda)y_1 + \lambda y_2) = -\ln y_\lambda.$$

Thus, we have shown that ξ also belongs to $\mathcal{U}_{\text{EVaR}}(y_\lambda, P(\cdot|x,a))$. Then, combining this fact

with (C.1), we obtain

$$
\begin{aligned}
&(1-\lambda)y_1\mathbf{T}[V](x,y_1)+\lambda y_2\mathbf{T}[V](x,y_2)\\
&\leq \min_{a\in\mathcal{A}}\left[y_\lambda C(x,a)+\gamma \max_{\xi\in\mathcal{U}_{\text{EVaR}}(y_\lambda,P(\cdot|x,a))}\sum_{x'\in\mathcal{X}}P(x'|x,a)y_\lambda\xi(x')V(x,y_\lambda\xi(x'))\right]\\
&= y_\lambda\mathbf{T}[V](x,y_\lambda).
\end{aligned}
$$

We have shown that $y\mathbf{T}[V](x,y)$ is concave in y under the assumption that $yV(x,y)$ is concave. Finally, to show that the inner maximization problem in (2.4) is concave, we need to show the following function:

$$
G_{x,y,a}(z) := \begin{cases} zV(x',z)P(x'|x,a)/y & \text{if} \quad y\neq 0 \\ 0 & \text{otherwise} \end{cases}
$$

is concave in $z\in\mathbb{R}$ for any given $x\in\mathcal{X}, y\in\mathcal{Y}$ and $a\in\mathcal{A}$. Suppose $zV(x,z)$ is a concave function in z, then for $y=0$, the function is concave in z. For $y\in\mathcal{Y}\backslash\{0\}$, since $P(x'|x,a)\geq 0$, we also have that $G_{x,y,a}(z)$ is concave in z. This further implies

$$
\sum_{x'\in\mathcal{X}}\xi(x')V(x',y\xi(x'))P(x'|x,a)=\sum_{x'\in\mathcal{X}}G_{x,y,a}(y\xi(x'))
$$

is concave in ξ. Combining this result with the fact that the envelope set of ξ is a polytope, we can prove the Property 4.

A.2 Proof of Theorem 2

The proof of Theorem 2 follows the idea in the proof of Theorem 4 in [24].

Let $\mathcal{C}_{0,T}=\sum_{t=0}^{T}\gamma^t C(x_t,a_t)$ denotes the total discounted cost from time 0 up to time T. For any $(x,y)\in\mathcal{X}\times\mathcal{Y}$, $V_0(x,y)$ is the bounded arbitrarily selected initial value. We divide the proof

into three parts and the first part is to show that for any $(x, y) \in \mathcal{X} \times \mathcal{Y}$,

$$\begin{aligned} V_n(x,y) &:= \mathbf{T}^n[V_0](x,y) \\ &= \min_{\pi \in \Pi_M} \text{EVaR}_y\big(\mathcal{C}_{0,n} + \gamma^n V_0(x_n, y_n) | x_0 = x, \pi\big), \end{aligned} \tag{A.2}$$

where $x_0 = x$, $y_0 = y$ and $a_t = \pi(x_t, y_t)$.

By induction hypothesis, firstly we need to verify (C.2) holds when $n = 1$. For $n = 1$, let (x_1, y_1) denotes $(x', y\xi(x'))$, from definition we have

$$V_1(x,y) = \mathbf{T}[V_0](x,y) = \min_{\pi \in \Pi_M} \big[C(x_0, a_0) + \gamma EVaR_y(C(x_1, a_1) + V_0(x_1, y_1) | x_0 = x, \pi)\big]$$

Note that when $n = 1$, a_1 only depends on x_1 and y_1, therefore, π is a Markovian policy, i.e., $\pi \in \Pi_M$. Hence, we obtain $V_1(x,y) = \min_{\pi \in \Pi_M} \text{EVaR}_y\big(\mathcal{C}_{0,1} + \gamma V_0(x_1, y_1) | x_0 = x, \pi\big)$.

Next, we assume that (C.2) holds at $n = k$.

Then for $n = k + 1$,

$$\begin{aligned} V_{k+1}(x,y) &:= \mathbf{T}^{k+1}[V_0](x,y) = \mathbf{T}[V_k](x,y) \\ &\overset{(1)}{=} \min_{a \in \mathcal{A}} \left[C(x,a) + \gamma \max_{\xi \in \mathcal{U}_{\text{EVaR}}(y, P(\cdot|x,a))} \sum_{x' \in \mathcal{X}} \xi(x') V_k(x', y\xi(x')) P(x'|x,a) \right] \\ &\overset{(2)}{=} \min_{a \in \mathcal{A}} \left[C(x,a) + \gamma \max_{\xi \in \mathcal{U}_{\text{EVaR}}(y, P(\cdot|x,a))} \sum_{x' \in \mathcal{X}} \xi(x') P(x'|x,a) \min_{\pi \in \Pi_M} \text{EVaR}_{y\xi(x')}(\mathcal{C}_{0,k} + \gamma^k V_0 | x_0 = x', \pi) \right] \\ &\overset{(3)}{=} \min_{a \in \mathcal{A}} [C(x,a) + \gamma \max_{\xi \in \mathcal{U}_{\text{EVaR}}(y, P(\cdot|x,a))} \sum_{x' \in \mathcal{X}} \xi(x') P(x'|x,a) \min_{\pi \in \Pi_M} \text{EVaR}_{y_1}(\mathcal{C}_{0,k} + \gamma^k V_0 | x_0 = x_1, \pi)] \\ &= \min_{a \in \mathcal{A}} \left[C(x,a) + \max_{\xi \in \mathcal{U}_{\text{EVaR}}(y, P(\cdot|x,a))} \sum_{x' \in \mathcal{X}} \xi(x') P(x'|x,a) \min_{\pi \in \Pi_M} \text{EVaR}_{y_1}(\gamma \mathcal{C}_{0,k} + \gamma^{k+1} V_0 | x_0 = x_1, \pi) \right] \\ &\overset{(4)}{=} \min_{a \in \mathcal{A}} \left[C(x,a) + \max_{\xi \in \mathcal{U}_{\text{EVaR}}(y, P(\cdot|x,a))} \mathbb{E}_{\xi P}\big[\min_{\pi \in \Pi_M} \text{EVaR}_{y_1}(\mathcal{C}_{1,k+1} + \gamma^{k+1} V_0 | x_1, \pi)\big] \right] \\ &\overset{(5)}{=} \min_{a \in \mathcal{A}} \big[\min_{\pi \in \Pi_M} \text{EVaR}_y(\mathcal{C}_{0,k+1} + \gamma^{k+1} V_0 | x_0 = x, \pi)\big] \\ &= \min_{\pi \in \Pi_M} \text{EVaR}_y(\mathcal{C}_{0,k+1} + \gamma^{k+1} V_0 | x_0 = x, \pi), \end{aligned} \tag{A.3}$$

where $x_0 = x$ and $y_0 = y$. The equality (1) is by the definition of $\mathbf{T}$, (2) is by plugging in the induction that (C.2) holds at $n = k$, (3) is by denoting $(x', y\xi(x')) = (x_1, y_1)$, (4) is by the definition of $\mathcal{C}_{0,k}$, i.e,

$$\begin{aligned}
&\gamma\mathcal{C}_{0,k}|x_0 = x_1, \pi \\
&= \gamma C(x_1, a_1) + \gamma^2 C(x_2, a_2) + \cdots + \gamma^{k+1} C(x_{k+1}, a_{k+1}) \\
&= \sum_{t=1}^{k+1} \gamma^t C(x_t, a_t) \\
&= \mathcal{C}_{1,k+1},
\end{aligned}$$

and (5) is by the EVaR decomposition theorem. Thus, (A.3) is proved by induction.

The second part of the proof is to show that

$$V^*(x_0, y_0) = \min_{\pi \in \Pi_M} \text{EVaR}_{y_0} \left(\lim_{n\to\infty} \mathcal{C}_{0,n}|x_0, \pi \right). \tag{A.4}$$

Recall the contraction property of $\mathbf{T}$ and the boundedness of V_0, for any $(x, y) \in \mathcal{X} \times \mathcal{Y}$, we can get the result that

$$\begin{aligned}
V^*(x, y) &= \mathbf{T}[V^*](x, y) \\
&= \lim_{n\to\infty} \mathbf{T}^n[V_0](x, y) = \lim_{n\to\infty} V_n(x, y).
\end{aligned}$$

The first equality is by the definition of V^*. The second equality can be obtained by Proposition 2.2 in [15]. The third equation is derived from the definition of V_n. Combining the above results, we have

$$\begin{aligned}
V^*(x_0, y_0) &= \lim_{n\to\infty} V_n(x_0, y_0) \\
&= \min_{\pi \in \Pi_M} \text{EVaR}_{y_0}\big(\lim_{n\to\infty} (\mathcal{C}_{0,n} + \gamma^n V_0(x_n, y_n))|x_0, \pi \big).
\end{aligned}$$

The second equality is due to the boundedness of both state-wise cost and V_0. Recall the

subadditivity property of EVaR, we obtain

$$\begin{aligned} V^*(x_0, y_0) &\leq \min_{\pi \in \Pi_M} \left[\text{EVaR}_{y_0}(\lim_{n\to\infty} \mathcal{C}_{0,n} | x_0, \pi) + \lim_{n\to\infty} \gamma^n \parallel V_0 \parallel_\infty \right] \\ &\leq \min_{\pi \in \Pi_M} \text{EVaR}_{y_0}(\lim_{n\to\infty} \mathcal{C}_{0,n} | x_0, \pi) + \lim_{n\to\infty} \gamma^n \parallel V_0 \parallel_\infty \\ &\leq \min_{\pi \in \Pi_M} \text{EVaR}_{y_0}(\lim_{n\to\infty} \mathcal{C}_{0,n} | x_0, \pi) + \left| \lim_{n\to\infty} \gamma^n \parallel V_0 \parallel_\infty \right| \end{aligned}$$

which implies

$$- \lim_{n\to\infty} \gamma^n \parallel V_0 \parallel_\infty \leq V^*(x_0, y_0) - \min_{\pi \in \Pi_M} \text{EVaR}_{y_0} \left(\lim_{n\to\infty} \mathcal{C}_{0,n} | x_0, \pi \right) \leq \lim_{n\to\infty} \gamma^n \parallel V_0 \parallel_\infty .$$

Since $\gamma \in (0, 1]$, the term $\lim_{n\to\infty} \gamma^n \parallel V_0 \parallel_\infty \to 0$ when $n \to \infty$. Thus, we obtain that

$$V^*(x_0, y_0) = \min_{\pi \in \Pi_M} \text{EVaR}_{y_0} \left(\lim_{n\to\infty} \mathcal{C}_{0,n} | x_0, \pi \right)$$

holds for any $(x_0, y_0) \in \mathcal{X} \times \mathcal{Y}$.

So far, we have established the optimal value over Markovian policies, the third part is to get the optimal value over all historic-dependent policies, i.e., for the initial conditions (x_0, y_0), we have that

$$V^*(x_0, y_0) = \min_{\pi \in \Pi_H} \text{EVaR}_{y_0}(\lim_{n\to\infty} \mathcal{C}_{0,n} | x_0, \pi).$$

For each $(x_t, y_t) \in \mathcal{X} \times \mathcal{Y}$, we first define the t^{th} tail-subproblem as follow:

$$\mathbb{V}(x_t, y_t) = \min_{\pi \in \Pi_H} \text{EVaR}_{y_t}(\lim_{n\to\infty} \mathcal{C}_{t,n} | x_t, \pi)$$

where the tail policy sequence is equal to $\pi = \{\pi_t, \pi_{t+1}, \dots\}$ and the action is given by $a_j = \pi_j(h_j)$ for $j \geq t$.

For any history depend policy $\tilde{\pi} \in \Pi_H$, we also define the $\tilde{\pi}$-induced value function as $\text{EVaR}_{y_t}(\lim_{n\to\infty} \mathcal{C}_{t,n} | x_t, \tilde{\pi})$ where $\tilde{\pi} = \{\tilde{\pi}_t, \tilde{\pi}_{t+1}, \dots\}$ and $a_j = \tilde{\pi}_j(h_j)$ for $j \geq t$.

Let π^* denote the optimal policy of the t^{th}-subproblem mentioned above, then the policy $\tilde{\pi} =$

$\{\pi^*_{t+1}, \pi^*_{t+2}, \dots\}$ is a feasible policy for the $(t+1)^{th}$-subproblem for any state x_{t+1} and confidence level y_{t+1}:

$$\min_{\pi \in \Pi_H} \text{EVaR}_{y_{t+1}}(\lim_{n\to\infty} \mathcal{C}_{t+1,n} | x_{t+1}, \pi).$$

Combining all the above results, for any $(x_t, y_t) \in \mathcal{X} \times \mathcal{Y}$ with $a_t = \pi^*_t(x_t)$, we can write

$$\begin{aligned}
\mathbb{V}(x_t, y_t) &= \min_{\pi \in \Pi_H} \text{EVaR}_{y_t}(\lim_{n\to\infty} \mathcal{C}_{t,n} | x_t, \pi) \\
&= \text{EVaR}_{y_t}(\lim_{n\to\infty} \mathcal{C}_{t,n} | x_t, \pi^*) \\
&= C(x_t, a_t) + \gamma \text{EVaR}_{y_t}(\lim_{n\to\infty} \mathcal{C}_{t+1,n} | x_{t+1}, \tilde{\pi}) \\
&\overset{(1)}{=} C(x_t, a_t) + \gamma \max_{\xi \in \mathcal{U}_{\text{EVaR}}(y_t, P(\cdot|x_t,a_t))} \mathbb{E}\big[\xi(x_{t+1}) \cdot \text{EVaR}_{y_{t+1}}(\lim_{n\to\infty} \mathcal{C}_{t+1,n} | x_{t+1}, \tilde{\pi})\big] \\
&\overset{(2)}{=} C(x_t, a_t) + \gamma \max_{\xi \in \mathcal{U}_{\text{EVaR}}(y_t, P(\cdot|x_t,a_t))} \mathbb{E}_\xi\big[\mathbb{V}^{\tilde{\pi}}(x_{t+1}, y_t\xi(x_t+1)) | x_t, y_t, a_t\big] \\
&\overset{(3)}{\geq} C(x_t, a_t) + \gamma \max_{\xi \in \mathcal{U}_{\text{EVaR}}(y_t, P(\cdot|x_t,a_t))} \mathbb{E}_\xi\big[\mathbb{V}(x_{t+1}, y_t\xi(x_t+1)) | x_t, y_t, a_t\big] \\
&\overset{(4)}{\geq} \mathbf{T}[\mathbb{V}](x_t, y_t)
\end{aligned}$$

where (1) is by the decomposition theorem, (2) is by defining $\mathbb{V}^{\tilde{\pi}}(x_t, y_t) = \text{EVaR}_{y_t}(\lim_{n\to\infty} \mathcal{C}_{t,n} | x_t, \tilde{\pi})$, (3) is by $\mathbb{V}^{\tilde{\pi}}(x, y) \geq \mathbb{V}(x, y)$ for any $(x, y) \in \mathcal{X} \times \mathcal{Y}$ and (4) is by the definition of $\mathbf{T}$.

On the other hand, for any state x_{t+1} and confidence level y_{t+1}, let $\pi^* = \{\pi^*_{t+1}, \pi^*_{t+2}, \dots\} \in \Pi_H$ be an optimal policy for the $(t+1)^{th}$ tail subproblem. Given $(x_t, y_t) \in \mathcal{X} \times \mathcal{Y}$, we can construct policy $\tilde{\pi} = \{\tilde{\pi}_t, \tilde{\pi}_{t+1}, \dots\} \in \Pi_H$ for the t^{th} subproblem from π^* by $\tilde{\pi}_t(x_t) = u^*(x_t, y_t)$ and $\tilde{\pi}_j(h_j) = \pi^*_j(h_j)$, where

$$u^*(x_t, y_t) \in \arg\min_{a \in \mathcal{A}} \big[C(x_t, a) + \gamma \max_{\xi \in \mathcal{U}_{\text{EVaR}}(y_t, P(\cdot|x_t,a))} \mathbb{E}_\xi[\mathbb{V}(x_{t+1}, y_t\xi x_{t+1}) | x_t, y_t, a]\big],$$

with y_t is the given confidence level to the t^{th} tail-subproblem and the transition from y_t to y_{t+1} is given by $y_{t+1} = y_t\xi^*(x_{t+1})$ where

$$\xi^* \in \arg \max_{\xi \in \mathcal{U}_{\text{EVaR}}(y_t, P(\cdot|x_t,a^*))} \mathbb{E}\big[\xi(x_{t+1}) \text{EVaR}_{y_t\xi(x_{t+1})}(\lim_{n\to\infty} \mathcal{C}_{t+1,n} | x_{t+1,n}, \tilde{\pi})\big].$$

Notice that π^* is an optimal and hence is a feasible policy for the tail subproblem from time $t+1$. Then the policy $\tilde{\pi} \in \Pi_H$ is a feasible policy for the tail subproblem from time t: $\min_{\pi\in\Pi_H} \text{EVaR}_{y_t}(\lim_{n\to\infty} \mathcal{C}_{t+1,n}|x_t, \pi)$. Hence,

$$\mathbb{V}(x_t, y_t) \leq C(x_t, \tilde{\pi}_t(x_t)) + \gamma \text{EVaR}_{y_t}(\lim_{n\to\infty} \mathcal{C}_{t+1,n}|x_t, \tilde{\pi}).$$

Recall the definition of π^*, we can immediately get

$$\begin{aligned}
&\mathbb{V}(x_t, y_t) \\
&\leq C(x_t, u^*(x_t, y_t)) \\
&\quad + \gamma \max_{\xi\in\mathcal{U}_{\text{EVaR}}(y_t, P(\cdot|x_t, u^*(x_t, y_t)))} \mathbb{E}\big[\xi(x_{t+1}) \cdot \text{EVaR}_{y_t\xi(x_{t+1})}(\lim_{n\to\infty} \mathcal{C}_{t+1,n}|x_{t+1}, \tilde{\pi})|x_t, y_t, u^*(x_t, y_t)\big] \\
&\leq C(x_t, u^*(x_t, y_t)) + \gamma \max_{\xi\in\mathcal{U}_{\text{EVaR}}(y_t, P(\cdot|x_t, u^*(x_t, y_t)))} \mathbb{E}_\xi\big[\mathbb{V}(x_{t+1}, y_t\xi(x_{t+1}))|x_t, y_t, u^*(x_t, y_t)\big] \\
&= \mathbf{T}[\mathbb{V}](x_t, y_t).
\end{aligned}$$

Combining the result $\mathbb{V}(x_t, y_t) \geq \mathbf{T}[\mathbb{V}](x_t, y_t)$ and $\mathbb{V}(x_t, y_t) \leq \mathbf{T}[\mathbb{V}](x_t, y_t)$, we show that $\mathbb{V}$ is a fixed-point solution of $\mathbb{V}(x_t, y_t) = \mathbf{T}[\mathbb{V}](x_t, y_t)$ for any $(x, y) \in \mathcal{X} \times \mathcal{Y}$. Since the fixed-point solution is unique, we can obtain $V^*(x, y) = \mathbb{V}(x, y)$ for any $(x, y) \in \mathcal{X} \times \mathcal{Y}$. Therefore, we have

$$V^*(x, y) = \mathbb{V}(x, y) = \min_{\pi\in\Pi_H} \text{EVaR}_y(\lim_{n\to\infty} \mathcal{C}_{0,n}|x_0 = x, \pi).$$

Equipped with the results from the above three parts, this claim is proved.

A.3 Proof of Theorem 3

The proof follows the similar idea with the proof of Theorem 5 in [24].

Firstly, for any $u \in \Pi_{M,S}$, we define the policy induced Bellman operator $\mathbf{T}_u$ as follows:

$$\mathbf{T}_u[V](x, y) = C(x, u(x, y)) + \gamma \max_{\xi\in\mathcal{U}_{\text{EVaR}}(y, P(\cdot|x, u(x,y)))} \sum_{x'\in\mathcal{X}} \xi(x')V(x', y\xi(x'))P(x'|x, u(x, y)).$$

Following the arguments in the proof of Theorem 2, we can show that the unique fixed-point solution to $T_u[V](x,y) = V(x,y)$ exists. Therefore, we need to show that the stationary Markovian policy u^* is optimal if and only if for any (x,y) in $\mathcal{X} \times \mathcal{Y}$

$$\mathbf{T}[V^*](x,y) = \mathbf{T}_{u^*}[V^*](x,y), \tag{A.5}$$

where $V^*(x,y)$ is the unique fixed-point solution of $\mathbf{T}[V](x,y) = V(x,y)$.

The first step is to show that, if $u^* \in \Pi_{M,S}$ is optimal, equation (A.5) holds. From Theorem 2, we know that

$$V^*(x,y) = \min_{\pi \in \Pi_H} \text{EVaR}_y \left(\lim_{T\to\infty} \mathcal{C}_{0,T} | x_0 = x, \pi \right).$$

Let V_{u^*} be the fixed-point solution to $\mathbf{T}_{u^*}[V](x,y) = V(x,y)$ for any (x,y) and combine the definition of u^* as described in Theorem 3, we can obtain $V^*(x,y) = V_{u^*}(x,y)$. Then, we have

$$\mathbf{T}[V^*](x,y) = V^*(x,y) = V_{u^*}(x,y) = \mathbf{T}_{u^*}[V_{u^*}](x,y).$$

The second step is to assume that equation (A.5) holds, we need to show $u^* \in \Pi_{M,S}$ is optimal. Recall that $\mathbf{T}[V^*](x,y) = V^*(x,y)$ holds for any (x,y), we obtain $V^*(x,y) = \mathbf{T}_{u^*}(x,y)$. Due to the uniqueness of fixed-point solution and the result from Theorem 2, we have

$$\mathbf{T}[V^*](x,y) = V^*(x,y) = V_{u^*}(x,y) = \min_{\pi \in \Pi_H} \text{EVaR}_y(\lim_{T\to\infty} \mathcal{C}_{0,T} | x_0 = x, \pi).$$

A.4 Proof of Theorem 4

The proof is inspired by the idea of the proof of Theorem 7 in [24].

We can rewrite the (2.11) as

$$Q_{k+1}(x,y,a) = (1-\zeta_k(x,y,a))Q_k(x,y,a) + \zeta_k(x,y,a)\cdot \left(\gamma \max_{\xi\in\mathcal{U}_{\text{EVaR}}(y,P(\cdot|x,a))} \sum_{x'\in\mathcal{X}} \frac{\mathcal{I}_{x'}[V_k](y\xi(x'))}{y} P(x'|x,a) + \gamma M_k(x,y,a) + C(x,a)\right),$$

where the noise term is given by

$$M_k(x,y,a)$$
$$= \max_{\xi\in\mathcal{U}_{\text{EVaR}}(y,P_{N_k}(\cdot|x,a))} \frac{1}{N_k}\sum_{i=1}^{N_k} \frac{\mathcal{I}_{x',i}[V_k](y\xi(x'^{,i}))}{y} - \max_{\xi\in\mathcal{U}_{\text{EVaR}}(y,P(\cdot|x,a))} \sum_{x'\in\mathcal{X}} \frac{\mathcal{I}_{x'}[V_k](y\xi(x'))}{y} P(x'|x,a)$$

for which $M_k(x,y,a) \to 0$ almost surely as $N_k \to \infty$ (consistency property of SAA shown in Chapter 5 of [88]) and for any $k \in \mathbb{N}$, let

$$T_1 = C(x,a) + \max_{\xi\in\mathcal{U}_{\text{EVaR}}(y,P_{N_k}(\cdot|x,a))} \frac{1}{N_k}\sum_{i=1}^{N_k} \frac{\mathcal{I}_{x',i}[V_k](y\xi(x'^{,i}))}{y},$$
$$T_2 = C(x,a) + \max_{\xi\in\mathcal{U}_{\text{EVaR}}(y,P(\cdot|x,a))} \sum_{x'\in\mathcal{X}} \frac{\mathcal{I}_{x'}[V_k](y\xi(x'))}{y} P(x'|x,a).$$

We can rewrite the noise term as

$$M_k(x,y,a) = T_1 - T_2 \le |T_1 - T_2|.$$

Then

$$M_k^2(x,y,a) \le |T_1-T_2|^2 \le |T_1|^2 + |T_2|^2 \le 2\max_{x,y,a} Q_k^2(x,y,a).$$

Then the assumptions in Proposition 4.5 in [15] on the noise term $M_k(x,y,a)$ are verified.

Now, we need to show that the operator $\mathbf{F}_{\mathcal{I}}$ is contraction. Firstly, we prove the monotonicity property. Based on the definition of $I_x[V](y)$, if $V_1(x,y) \ge V_2(x,y), \forall x \in \mathcal{X}, y \in \mathcal{Y}$, we have that

for $y \in \mathbf{I}_{i+1}(x)$

$$\mathcal{I}_x[V_1](y) = \frac{y_{i+1}V_1(x, y_{i+1})(y - y_i) + y_i V_1(x, y_i)(y_{i+1} - y)}{y_{i+1} - y_i}.$$

Since $y_i, y_{i+1} \in \mathcal{Y}$ and $(y_{i+1} - y), (y - y_i) \geq 0$, we can easily see that $I_x[V_1](y) \geq I_x[V_2](y)$. As $y \in \mathcal{Y}$ and $\xi(\cdot)P(\cdot|x, a) \geq 0$ for any $\xi \in \mathcal{U}_{\text{EVaR}}(y, P(\cdot|x, a))$, this further implies $\mathbf{F}_{\mathcal{I}}[V_1](x, y) \geq \mathbf{F}_{\mathcal{I}}[V_2](x, y)$.

Next we prove the constant shift property. From the definition of $I_x[V](y)$ that for a constant K, we have that

$$\begin{aligned}
\mathcal{I}_x[V + K](y) &= y_i(V(x, y_i) + K) + \frac{y_{i+1}(V(x, y_{i+1}) + K) - y_i(V(x, y_i) + K)}{y_{i+1} - y_i}(y - y_i) \\
&= yK + y_i V(x, y_i) + \frac{y_{i+1}V(x, y_{i+1}) - y_i V(x, y_i)}{y_{i+1} - y_i}(y - y_i) \\
&= yK + \mathcal{I}_x[V](y).
\end{aligned}$$

Therefore, by definition of $\mathbf{F}_I[V](x, y)$, the constant shift property:

$$\mathbf{T}_{\mathcal{I}}[V + K](x, y) = \mathbf{T}_{\mathcal{I}}[V](x, y) + \gamma K, \forall x \in \mathcal{X}, y \in \mathcal{Y}$$

follows directly from the above arguments. Based on these two properties, we can prove the contraction of $\mathbf{F}_{\mathcal{I}}$ directly follow steps in Lemma 1, which means, for any two state-action value function $Q_1(x, y, a)$ and $Q_2(x, y, a)$ such that $V_1(x, y) = \min_{a \in \mathcal{A}} Q_1(x, y, a)$ and $V_2(x, y) = \min_{a \in \mathcal{A}} Q_2(x, y, a)$, we have that $||\mathbf{F}_{\mathcal{I}}[Q_1] - \mathbf{F}_{\mathcal{I}}[Q_2]|| \leq \gamma ||Q_1 - Q_2||_\infty$.

By combining these arguments, all assumptions in Proposition 4.5 in [15] are justified. This in turns implies the convergence of $\{Q_k(x, y, a)\}_{k \in \mathbb{N}}$ to $Q^*(x, y, a)$ component-wise, where Q^* is the unique fixed-point solution of $\mathbf{F}_{\mathcal{I}}[Q](x, y, a) = Q(x, y, a)$.

A.5 Proof of Theorem 5

Here we follow the same idea used in the proof of Theorem 7 in [23]. First, we can regard these updates as a multi-time scale discrete stochastic approximation and show the sequences $\{\nu_k\}$ and $\{\theta_k\}$ converge to the solution of the corresponding continuous time systems with different speed by [19], which further converge to the local asymptotically stable point (ν^*, θ^*) by applying Lyapunov analysis in Chapter 4 of [56].

Step 1. Show ν-update converges.

Following Chapter 6 in Borkar's book [19], since ν converges faster, when analyzing the convergence of ν-update, we can view θ as constant, then the ν-update rule can be rewritten as

$$\nu_{k+1} = \Gamma_{\mathcal{N}}\left[\nu_k - \zeta_2(k)\left(- \nu_k^{-2} \ln \frac{\sum_{j=1}^N e^{J(\xi_{j,k})\nu_k}}{N(1-\alpha)} + \nu_k^{-1} \frac{\sum_{j=1}^N J(\xi_{j,k}) e^{J(\xi_{j,k})\nu_k}}{\sum_{j=1}^N e^{J(\xi_{j,k})\nu_k}}\right)\right].$$

Considering the following dynamic system

$$\dot{\nu} = \Upsilon_{\nu}[-\nabla_{\nu} L(\nu, \theta)] \tag{A.6}$$

where

$$\Upsilon_{\nu}[-K(\nu)] = \lim_{0<\eta\to 0} \frac{\Gamma_{\mathcal{N}}(\nu + \eta K(\nu)) - \Gamma_{\mathcal{N}}(\nu)}{\eta}.$$

Now consider the following equation,

$$\nu_{k+1} = \Gamma_{\mathcal{N}}\left[\nu_k + \zeta_2(k)\left(\nu_k^{-2} \ln \frac{\sum_{j=1}^N e^{J(\xi_{j,k})\nu_k}}{N(1-\alpha)} - \nu_k^{-1} \frac{\sum_{j=1}^N J(\xi_{j,k}) e^{J(\xi_{j,k})\nu_k}}{\sum_{j=1}^N e^{J(\xi_{j,k})\nu_k}} + \delta_{\nu_{k+1}}\right)\right]$$

where

$$\begin{aligned}\delta_{\nu_{k+1}} = &-\left(\nu_k^{-2} \ln \frac{\sum_{j=1}^N e^{J(\xi_{j,k})\nu_k}}{N(1-\alpha)} - \nu_k^{-1} \frac{\sum_{j=1}^N J(\xi_{j,k}) e^{J(\xi_{j,k})\nu_k}}{\sum_{j=1}^N e^{J(\xi_{j,k})\nu_k}}\right) \\ &+ \left(\nu_k^{-2} \ln \frac{\sum_{\xi} \mathbb{P}_{\theta}(\xi) e^{J(\xi)\nu_k}}{1-\alpha} - \nu_k^{-1} \frac{\sum_{\xi} \mathbb{P}_{\theta}(\xi) J(\xi) e^{J(\xi)\nu_k}}{\sum_{\xi} \mathbb{P}_{\theta}(\xi) e^{J(\xi)\nu_k}}\right).\end{aligned}$$

In order to show that the update rule converges to the solution of (A.6), we need to verify several conditions to satisfy the assumptions of Theorem 2 in Chapter 2 of Borkar's book[19]:

(1) $\nabla_\nu L(\nu,\theta)$ is Lipschitz in ν.

Proof. Recall that

$$\nabla_\nu L(\nu,\theta) = -\nu^{-2}\ln\frac{\sum_\xi \mathbb{P}_\theta(\xi)e^{J(\xi)\nu}}{1-\alpha} + \nu^{-1}\frac{\sum_\xi \mathbb{P}_\theta(\xi)J(\xi)e^{J(\xi)\nu}}{\sum_\xi \mathbb{P}_\theta(\xi)e^{J(\xi)\nu}}$$

and denote this function as $f(\nu)$. Since $\nu \in [V_{\min}, V_{\max}]$ and $\nabla_\nu L(\nu,\theta)$ is continuous and differentiable, we have

$$\begin{aligned} f'(\nu) = {} & 2\nu^{-3}\ln\frac{\sum_\xi \mathbb{P}_\theta(\xi)e^{J(\xi)\nu}}{1-\alpha} - 2\nu^{-2}\frac{\sum_\xi \mathbb{P}_\theta(\xi)J(\xi)e^{J(\xi)\nu}}{\sum_\xi \mathbb{P}_\theta(\xi)e^{J(\xi)\nu}} \\ & + \nu^{-1}\frac{\sum_\xi \mathbb{P}_\theta(\xi)J^2(\xi)e^{J(\xi)\nu}}{\sum_\xi \mathbb{P}_\theta(\xi)e^{J(\xi)\nu}} - \left(\frac{\sum_\xi \mathbb{P}_\theta(\xi)J(\xi)e^{J(\xi)\nu}}{\sum_\xi \mathbb{P}_\theta(\xi)e^{J(\xi)\nu}}\right)^2. \end{aligned}$$

By applying the subadditivity property of absolute value, we have

$$\begin{aligned} |f'(\nu)| \le {} & \left|2\nu^{-3}\ln\frac{\sum_\xi \mathbb{P}_\theta(\xi)e^{J(\xi)\nu}}{1-\alpha}\right| + \left|2\nu^{-2}\frac{\sum_\xi \mathbb{P}_\theta(\xi)J(\xi)e^{J(\xi)\nu}}{\sum_\xi \mathbb{P}_\theta(\xi)e^{J(\xi)\nu}}\right| \\ & + \left|\nu^{-1}\frac{\sum_\xi \mathbb{P}_\theta(\xi)J^2(\xi)e^{J(\xi)\nu}}{\sum_\xi \mathbb{P}_\theta(\xi)e^{J(\xi)\nu}}\right| + \left|\nu^{-1}(\frac{\sum_\xi \mathbb{P}_\theta(\xi)J(\xi)e^{J(\xi)\nu}}{\sum_\xi \mathbb{P}_\theta(\xi)e^{J(\xi)\nu}})^2\right| \\ \le {} & 2\nu^{-3}\left(\frac{D_{\max}}{1-\gamma}\nu - \ln(1-\alpha)\right) + 2\nu^{-2}\frac{D_{\max}}{1-\gamma} + \nu^{-1}\left(\frac{D_{\max}}{1-\gamma}\right)^2 + \nu^{-1}\left(\frac{D_{\max}}{1-\gamma}\right)^2 \\ \le {} & 4V_{\min}^{-2}\frac{D_{\max}}{1-\gamma} + 2V_{\min}^{-1}\left(\frac{D_{\max}}{1-\gamma}\right)^2 - 2V_{\min}^{-3}\ln(1-\alpha). \end{aligned}$$

By the mean value theorem, $\nabla_\nu L(\nu,\theta)$ is Lipschitz in ν. □

(2). Stepsize $\zeta_2(k)$ satisfies $\sum_k \zeta_2(k) = \infty$ and $\sum_k \zeta_2^2(k) < \infty$.

Proof. Refer to Assumption 5. □

(3). $\{\delta_{\nu_{k+1}}\}$ is a Martingale difference sequence (MDS), i.e., $\mathbb{E}[\delta_{\nu_{k+1}}|\mathcal{F}_{\nu,k}] = 0$ and $\mathbb{E}[||\delta_{\nu_{k+1}}||^2|\mathcal{F}_{\nu,k}] \le K(1+||\nu_k||^2)$ where $\mathcal{F}_{\nu,k} = \sigma(\nu_m, \delta_m, m \le k)$ is the filtration of ν_k generated

by different independent trajectories.

Proof. (i). $\mathbb{E}[\delta_{\nu_{k+1}}|\mathcal{F}_{\nu,k}] = 0$ since the trajectories are generated based on the sampling probability mass function and all these trajectories are independent.

(ii). Recall that

$$\begin{aligned}
|\delta_{\nu_{k+1}}| &= \left| -\left(\nu_k^{-2}\ln\frac{\sum_{j=1}^N e^{J(\xi_{j,k})\nu_k}}{N(1-\alpha)} - \nu_k^{-1}\frac{\sum_{j=1}^N J(\xi_{j,k})e^{J(\xi_{j,k})\nu_k}}{\sum_{j=1}^N e^{J(\xi_{j,k})\nu_k}}\right)\right. \\
&\quad \left. + \left(\nu_k^{-2}\ln\frac{\sum_{\xi}\mathbb{P}_\theta(\xi)e^{J(\xi)\nu_k}}{1-\alpha} - \nu_k^{-1}\frac{\sum_{\xi}\mathbb{P}_\theta(\xi)J(\xi)e^{J(\xi)\nu_k}}{\sum_{\xi}\mathbb{P}_\theta(\xi)e^{J(\xi)\nu_k}}\right)\right| \\
&= \left|\nu_k^{-2}\left(\ln\sum_{\xi}\mathbb{P}_\theta(\xi)e^{J(\xi)\nu_k} - \ln\frac{\sum_{\xi}e^{J(\xi)\nu_k}}{N}\right)\right. \\
&\quad \left. + \nu_k^{-1}\left(\frac{\sum_{j=1}^N J(\xi_{j,k})e^{J(\xi_{j,k})\nu_k}}{\sum_{j=1}^N e^{J(\xi_{j,k})\nu_k}} - \frac{\sum_{\xi}\mathbb{P}_\theta(\xi)J(\xi)e^{J(\xi)\nu_k}}{\sum_{\xi}\mathbb{P}_\theta(\xi)e^{J(\xi)\nu_k}}\right)\right| \\
&\le \left|\nu_k^{-2}\cdot 2\frac{D_{\max}}{1-\gamma}\nu_k + \nu_k^{-1}\cdot 2\frac{D_{\max}}{1-\gamma}\right| \\
&\le 4V_{\min}^{-1}\frac{D_{\max}}{1-\gamma}.
\end{aligned}$$

Thus, $\mathbb{E}[||\delta_{\nu_{k+1}}||^2|\mathcal{F}_{\nu,k}] \le (4V_{\min}^{-1}\frac{D_{\max}}{1-\gamma})^2$. Combining (i) and (ii), we show that $\delta_{\nu_{k+1}}$ is a MDS. □

(4). $\sup_k |||\nu_k||^2 < \infty$.

Proof. By Assumption 2 and note that the projection ensures ν_k is in $[V_{\min}, V_{\max}]$. □

Based on these conditions, we can invoke Theorem 2 in Ch2 of Borkar's book to show that the sequence of $\{\nu_k\}$ converges almost surely to the solution of the o.d.e for ν. To show the convergence of ν-update, it remains to show that the solution of the o.d.e converges to a fixed point.

For any given θ, define the following Lyapunov function

$$\mathcal{L}_\theta(\nu) = L(\nu,\theta) - L(\nu^*,\theta)$$

where ν^* is a minimum point since $L(\nu,\theta)$ is convex in ν. Then $\mathcal{L}_\theta(\nu)$ is a positive definite

function, i.e., $\mathcal{L}_\theta(\nu) \geq 0$. In order to use the Lyapunov theory for asymptotically stable system from Chapter 4 of Khalil and Grizzle [56], we need to show $\frac{d}{dt}\mathcal{L}_\theta(\nu) \leq 0$ and it's non-zero if $||\Upsilon_\nu[-\nabla_\nu L(\nu,\theta)]|| \neq 0$.

Now define

$$\frac{d}{dt}L(\nu,\theta) = \nabla_\nu L(\nu,\theta)\Upsilon_\nu[-\nabla_\nu L(\nu,\theta)],$$

then

$$\frac{d}{dt}\mathcal{L}_\theta(\nu) = \frac{d}{dt}L(\nu,\theta) - \frac{d}{dt}L(\nu,\theta)|_{\nu=\nu^*} = \frac{d}{dt}L(\nu,\theta) = \nabla_\nu L(\nu,\theta)\Upsilon_\nu[-\nabla_\nu L(\nu,\theta)].$$

It remains to show $\frac{d}{dt}L(\nu,\theta) \leq 0$ and it's non-zero whenever $||\Upsilon_\nu[-\nabla_\nu L(\nu,\theta)]|| \neq 0$ by considering three cases.

Case 1: When $\nu \in (V_{\min}, V_{\max})$.

$\exists$ a sufficiently small $\eta_0 > 0$ such that $\nu - \eta_0\nabla_\nu L(\nu,\theta) \in [V_{\min}, V_{\max}]$ and

$$\Gamma_{\mathcal{N}}(\nu - \eta_0\nabla_\nu L(\nu,\theta)) - \nu = -\eta_0\nabla_\nu L(\nu,\theta).$$

Recall that $\Upsilon_\nu[-\nabla_\nu L(\nu,\theta)] = \lim_{0<\eta\to 0}\frac{\Gamma_{\mathcal{N}}(\nu-\eta\nabla_\nu L(\nu,\theta))-\Gamma_{\mathcal{N}}(\nu)}{\eta}$, we have

$$\frac{d}{dt}L(\nu,\theta) = -||\nabla_\nu L(\nu,\theta)||^2 \leq 0$$

and $\frac{d}{dt}L(\nu,\theta)$ is non-zero if $\nabla_\nu L(\nu,\theta) \neq 0$, i.e., $||\Upsilon_\nu[-\nabla_\nu L(\nu,\theta)]|| \neq 0$.

Case 2: When $\nu \in \{V_{\min}, V_{\max}\}$, for any $\eta \in (0,\eta_0]$ and some $\eta_0 > 0$, $(\nu - \eta_0\nabla_\nu L(\nu,\theta)) \in [V_{\min}, V_{\max}]$.

Since $(\nu - \eta\nabla_\nu L(\nu,\theta)) \in [V_{\min}, V_{\max}]$, we have

$$\Upsilon_\nu[-\nabla_\nu L(\nu,\theta)] = -\nabla_\nu L(\nu,\theta)$$

and this further implies

$$\frac{d}{dt}L(\nu,\theta) = -||\nabla_\nu L(\nu,\theta)||^2 \leq 0$$

and $\frac{d}{dt}L(\nu,\theta)$ is non-zero if $\nabla_\nu L(\nu,\theta) \neq 0$, i.e., $||\Upsilon_\nu[-\nabla_\nu L(\nu,\theta)]|| \neq 0$.

Case 3: When $\nu \in \{V_{\min}, V_{\max}\}$, there exists some $\eta \in (0,\eta_0]$ for any $\eta_0 > 0$, $(\nu - \eta\nabla_\nu L(\nu,\theta)) \notin [V_{\min}, V_{\max}]$.

Define $\nu_\eta := \nu - \eta\nabla_\nu L(\nu,\theta)$ for any $\eta > 0$. In this condition, considering the strong convex function $f(\nu) = \frac{1}{2}(\nu - \nu_\eta)^2$, invoking the first order optimality condition, one obtains for all $\nu \in [V_{\min}, V_{\max}]$,

$$\nabla f(\nu_\eta^*)(\nu - \nu^*) = (\nu^* - \nu_\eta)(\nu - \nu_\eta^*) \geq 0.$$

Note that ν_η^* is the projection of ν_η and it's unique due to the strong convexity of $f(\nu)$. Then, we can have

$$\begin{aligned}
\frac{d}{dt}L(\nu,\theta) &= \nabla_\nu L(\nu,\theta)\Upsilon_\nu[-\nabla_\nu L(\nu,\theta)] \\
&= \nabla_\nu L(\nu,\theta)\left(\lim_{0<\eta\to 0}\frac{\nu_\eta^* - \nu}{\eta}\right) \\
&= \left(\lim_{0<\eta\to 0}\frac{-(\nu_\eta - \nu)}{\eta}\right)\left(\lim_{0<\eta\to 0}\frac{\nu_\eta^* - \nu}{\eta}\right) \\
&= \lim_{0<\eta\to 0}\frac{-||\nu_\eta^* - \nu||^2}{\eta^2} + \lim_{0<\eta\to 0}\frac{(\nu_\eta^* - \nu_\eta)(\nu_\eta^* - \nu)}{\eta^2} \leq 0.
\end{aligned}$$

Based on these arguments, we have shown that $\frac{d}{dt}\mathcal{L}_\theta(\nu) \leq 0$ and it's non-zero whenever $||\Upsilon_\nu[-\nabla_\nu L(\nu,\theta)]|| \neq 0$. Then by applying the Lyapunov theory in the reference, we know that for any initial condition $\nu(0)$, $\nu(t)$ converges to ν^*, i.e., $L(\nu^*,\theta) \leq L(\nu(t),\theta) \leq L(\nu(0),\theta)$ for any $t \geq 0$. Furthermore, v^* is a stationary point since $\Upsilon_\nu[-\nabla_\nu L(\nu,\theta)]|_{\nu=\nu^*} = 0$.

Therefore, the sequence $\{\nu_k\}$ generated by (2.21) converges to the solution of (A.6), which converges almost surely to ν^*.

Step 2. Show θ-update converges.

Since ν converges on a faster timescale than θ, the θ-update can be rewritten using the

converged $\nu^*(\theta)$, i.e.,

$$\theta_{k+1} = \Gamma_\Theta\left[\theta_k - \zeta_1(k)\nu^*(\theta_k)^{-1}\frac{\sum_{j=1}^N \nabla_\theta \log \mathbb{P}_\theta(\xi_{j,k})|_{\theta=\theta_k} e^{J(\xi_{j,k})\nu^*(\theta_k)}}{\sum_{j=1}^N e^{J(\xi_{j,k})\nu^*(\theta_k)}}\right].$$

Considering the following dynamic system

$$\dot{\theta} = \Upsilon_\theta[-\nabla_\theta L(\nu,\theta)|_{\nu=\nu^*(\theta)}] \tag{A.7}$$

where

$$\Upsilon_\theta[-K(\theta)] = \lim_{0<\eta\to 0}\frac{\Gamma_\Theta(\theta+\eta K(\theta)) - \Gamma_\Theta(\theta)}{\eta}.$$

Note that $||\nu_k - \nu^*(\theta_k|| \to 0$ almost surely and by the continuity of $\nabla_\theta L(\nu,\theta)$, we know that

$$||\nabla_\theta L(\nu,\theta)|_{\theta=\theta_k,\nu=\nu_k} - \nabla_\theta L(\nu,\theta)|_{\theta=\theta_k,\nu=\nu^*(\theta_k)}|| \to 0$$

a.s.. Then, we can rewrite the θ-update as

$$\theta_{k+1} = \Gamma_\Theta\left[\theta_k + \zeta_1(k)\left(-\nu^*(\theta_k)^{-1}\frac{\sum_\xi \mathbb{P}_\theta(\xi)\nabla_\theta \log \mathbb{P}_\theta(\xi)|_{\theta=\theta_k} e^{J(\xi)\nu^*(\theta_k)}}{\sum_\xi \mathbb{P}_\theta(\xi)|_{\theta=\theta_k} e^{J(\xi)\nu^*(\theta_k)}} + \delta_{\theta_{k+1}}\right)\right]$$

where

$$\begin{aligned}
\delta_{\theta_{k+1}} &= \nu^*(\theta_k)^{-1}\frac{\sum_\xi \mathbb{P}_\theta(\xi)\nabla_\theta \log \mathbb{P}_\theta(\xi)|_{\theta=\theta_k} e^{J(\xi)\nu^*(\theta_k)}}{\sum_\xi \mathbb{P}_\theta(\xi)|_{\theta=\theta_k} e^{J(\xi)\nu^*(\theta_k)}} \\
&\quad - \nu^*(\theta_k)^{-1}\frac{\sum_{j=1}^N \nabla_\theta \log \mathbb{P}_\theta(\xi_{j,k})|_{\theta=\theta_k} e^{J(\xi_{j,k})\nu^*(\theta_k)}}{\sum_{j=1}^N e^{J(\xi_{j,k})\nu^*(\theta_k)}} \\
&\quad + \left(\nu^*(\theta_k)^{-1} - \nu_k^{-1}\right)\frac{\sum_{j=1}^N \nabla_\theta \log \mathbb{P}_\theta(\xi_{j,k})|_{\theta=\theta_k} e^{J(\xi_{j,k})\nu^*(\theta_k)}}{\sum_{j=1}^N e^{J(\xi_{j,k})\nu^*(\theta_k)}} \\
&\quad + \nu_k^{-1}\left(\frac{\sum_{j=1}^N \nabla_\theta \log \mathbb{P}_\theta(\xi_{j,k})|_{\theta=\theta_k} e^{J(\xi_{j,k})\nu^*(\theta_k)}}{\sum_{j=1}^N e^{J(\xi_{j,k})\nu^*(\theta_k)}} - \frac{\sum_{j=1}^N \nabla_\theta \log \mathbb{P}_\theta(\xi_{j,k})|_{\theta=\theta_k} e^{J(\xi_{j,k})\nu_k}}{\sum_{j=1}^N e^{J(\xi_{j,k})\nu_k}}\right)
\end{aligned}$$

Similarly, to prove (A.7) is well-posed, we need to check:

(1). $\nabla_\theta L(\nu,\theta)$ is Lipschitz in θ.

Proof. Recall that

$$\nabla_\theta L(\nu,\theta) = \nu^{-1}\frac{\sum_\xi \mathbb{P}_\theta(\xi)\nabla_\theta \log \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}}{\sum_\xi \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}}.$$

For convenience, Let $f(\theta)$ denote the above function, we can obtain

$$f'(\theta) = \nu^{-1}\frac{\nabla_\theta(\sum_\xi \mathbb{P}_\theta(\xi)\nabla_\theta \log \mathbb{P}_\theta(\xi) e^{J(\xi)\nu})}{\sum_\xi \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}} - \nu^{-1}\left(\frac{\sum_\xi \mathbb{P}_\theta(\xi)\nabla_\theta \log \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}}{\sum_\xi \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}}\right)^2$$

By applying the subadditivity of absolute value, we have

$$|f'(\theta)| \leq \left|\nu^{-1}\frac{\nabla_\theta(\sum_\xi \mathbb{P}_\theta(\xi)\nabla_\theta \log \mathbb{P}_\theta(\xi) e^{J(\xi)\nu})}{\sum_\xi \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}}\right| + \left|\nu^{-1}\left(\frac{\sum_\xi \mathbb{P}_\theta(\xi)\nabla_\theta \log \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}}{\sum_\xi \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}}\right)^2\right|$$

(i). For the first term, combining Proposition 5 and the fact that $\sum_\xi \mathbb{P}_\theta(\xi) e^{J(\xi)\nu} \geq 1$ (since $J(\xi) \in [0, \frac{D_{\max}}{1-\gamma}]$), we have

$$\left|\nu^{-1}\frac{\nabla_\theta(\sum_\xi \mathbb{P}_\theta(\xi)\nabla_\theta \log \mathbb{P}_\theta(\xi) e^{J(\xi)\nu})}{\sum_\xi \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}}\right| \leq V_{\min}^{-1} K_1(\xi).$$

(ii). For the second term, by Proposition 6, we have

$$\left|\nu^{-1}(\frac{\sum_\xi \mathbb{P}_\theta(\xi)\nabla_\theta \log \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}}{\sum_\xi \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}})^2\right| \leq V_{\min}^{-1} K_2^2(\xi)$$

Based on these arguments, we have

$$|f'(\theta)| \leq V_{\min}^{-1} K_1(\xi) + V_{\min}^{-1} K_2^2(\xi).$$

Therefore, by the mean value theorem, $\nabla_\theta L(\nu,\theta)$ is Lipschitz in θ. □

(2). Stepsize $\zeta_1(k)$ satisfies $\sum_k \zeta_1(k) = \infty$ and $\sum_k \zeta_1^2(k) < \infty$.

Proof. Refer to Assumption 5. □

(3). $\{\delta_{\theta_{k+1}}\}$ is a Martingale difference sequence (MDS), i.e., $\mathbb{E}[\delta_{\theta_{k+1}}|\mathcal{F}_{\theta,k}] = 0$ and

$\mathbb{E}[||\delta_{\theta_{k+1}}||^2|\mathcal{F}_{\theta,k}] \leq K(1+||\nu_k||^2)$ where $\mathcal{F}_{\theta,k} = \sigma(\theta_m, \delta_m, m \leq k)$ is the filtration of θ_k generated by different independent trajectories.

Proof. (i). $\mathbb{E}[\delta_{\theta_{k+1}}|\mathcal{F}_{\theta,k}] = 0$ since the trajectories are generated based on the sampling probability mass function and all these trajectories are independent.
(ii). First we consider the last two components of $\delta_{\theta_{k+1}}$. Since $||\nu^*(\theta_k) - \nu_k|| \to 0$ almost surely and $\nabla_\theta \log \mathbb{P}_\theta(\xi_{j,k})$ is Lipschitz in θ, θ lies in a compact set Θ, both $J(\xi_{j,k})$ and π_k are bounded, and $\nu, \nu^*(\theta_k)$ lie in a compact set $\mathcal{N}$, we can concludes that as $k \to \infty$, the last two components converges to 0 almost surely. Then,

$$\begin{aligned}
|\delta_{\theta_{k+1}}| &\leq \left| \nu^*(\theta_k)^{-1} \frac{\sum_\xi \mathbb{P}_\theta(\xi) \nabla_\theta \log \mathbb{P}_\theta(\xi)|_{\theta=\theta_k} e^{J(\xi)\nu^*(\theta_k)}}{\sum_\xi \mathbb{P}_\theta(\xi)|_{\theta=\theta_k} e^{J(\xi)\nu^*(\theta_k)}} \right| \\
&\quad + \left| \nu^*(\theta_k)^{-1} \frac{\sum_{j=1}^N \nabla_\theta \log \mathbb{P}_\theta(\xi_{j,k})|_{\theta=\theta_k} e^{J(\xi_{j,k})\nu^*(\theta_k)}}{\sum_{j=1}^N e^{J(\xi_{j,k})\nu^*(\theta_k)}} \right| \\
&\leq V_{\min}^{-1} K_2(\xi) + V_{\min}^{-1} K_2(\xi) = 2V_{\min}^{-1} K_2(\xi).
\end{aligned}$$

Therefore, we can show that

$$\mathbb{E}[||\delta_{\theta_{k+1}}||^2|\mathcal{F}_{\theta,k}] \leq \left(2V_{\min}^{-1} K_2(\xi)\right)^2$$

□

(4). $\sup_k |||\theta_k||^2 < \infty$.

Proof. Refer to the assumption that $\theta \in \Theta$ and the projection in the update rule ensures every $\theta_k \in \Theta$. □

Based on these conditions, we can invoke Theorem 2 in Ch2 of Borkar's book to show that the sequence of $\{\theta_k\}$ converges almost surely to the solution of the o.d.e for θ. In addition, we need to show that $\nu^*(\theta)$ is an asymptotically stable equilibrium point for the sequence $\{\nu_k\}$. Besides the convergence analysis of the ν-update, we need to check that $\nabla_\nu L(\nu, \theta)$ is Lipschitz in θ.

Proposition 4. $\nabla_\nu L(\nu, \theta)$ is Lipschitz in θ.

Proof. Recall that

$$\nabla_\nu L(\nu, \theta) = -\nu^{-2} \ln \frac{\sum_\xi \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}}{1-\alpha} + \nu^{-1} \frac{\sum_\xi \mathbb{P}_\theta(\xi) J(\xi) e^{J(\xi)\nu}}{\sum_\xi \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}},$$

let $f(\theta)$ denote this function and we have

$$\begin{aligned} f'(\theta) = &-\nu^{-2} \frac{\sum_\xi \mathbb{P}_\theta(\xi) \nabla_\theta \log \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}}{\sum_\xi \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}} + \nu^{-1} \frac{\sum_\xi \mathbb{P}_\theta(\xi) \nabla_\theta \log \mathbb{P}_\theta(\xi) J(\xi) e^{J(\xi)\nu}}{\sum_\xi \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}} \\ &- \nu^{-1} \frac{\sum_\xi \mathbb{P}_\theta(\xi) J(\xi) e^{J(\xi)\nu} \cdot \sum_\xi \mathbb{P}_\theta(\xi) \nabla_\theta \log \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}}{(\sum_\xi \mathbb{P}_\theta(\xi) e^{J(\xi)\nu})^2}. \end{aligned}$$

Then, we obtain

$$\begin{aligned} |f'(\theta)| &\leq \nu^{-2} \left| \frac{\sum_\xi \mathbb{P}_\theta(\xi) \nabla_\theta \log \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}}{\sum_\xi \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}} \right| + \nu^{-1} \left| \frac{\sum_\xi \mathbb{P}_\theta(\xi) \nabla_\theta \log \mathbb{P}_\theta(\xi) J(\xi) e^{J(\xi)\nu}}{\sum_\xi \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}} \right| \\ &\quad + \nu^{-1} \left| \frac{\sum_\xi \mathbb{P}_\theta(\xi) J(\xi) e^{J(\xi)\nu}}{\sum_\xi \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}} \right| \cdot \left| \frac{\sum_\xi \mathbb{P}_\theta(\xi) \nabla_\theta \log \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}}{\sum_\xi \mathbb{P}_\theta(\xi) e^{J(\xi)\nu}} \right| \\ &\leq V_{\min}^{-2} K_2(\xi) + 2 V_{\min}^{-1} \frac{D_{\max}}{1-\gamma} K_2(\xi) \end{aligned}$$

Therefore, by the mean value theorem, $\nabla_\nu L(\nu, \theta)$ is Lipschitz in θ. □

To show the convergence of θ-update, it remains to show that the solution of the o.d.e converges to a fixed point θ^*. Now we apply the Lyapunov analysis for θ-update. Define the following Lyapunov function

$$\mathcal{L}(\theta) = L(\nu^*(\theta), \theta) - L(\nu^*(\theta^*), \theta^*),$$

where θ^* is a local minimum point. Then $\mathcal{L}(\theta)$ is a local positive definite function, i.e., $\mathcal{L}(\theta) \geq 0$. In order to use the Lyapunov theory for asymptotically stable system from Chapter 4 of Khalil and Grizzle (2002), we need to verify to show $\frac{d}{dt}\mathcal{L}(\nu) \leq 0$ and it's non-zero if $||\Upsilon_\nu[-\nabla_\theta L(\nu, \theta)|_{\nu=\nu^*(\theta)}]|| \neq 0$.

Now define

$$\frac{d}{dt}L(\nu,\theta)|_{\nu=\nu^*(\theta)} = (\nabla_\theta L(\nu,\theta)|_{\nu=\nu^*(\theta)})^{\mathrm{T}}\Upsilon_\theta[-\nabla_\theta L(\nu,\theta)|_{\nu=\nu^*(\theta)}],$$

then

$$\frac{d}{dt}\mathcal{L}(\nu) = \frac{d}{dt}L(\nu^*(\theta),\theta) - \frac{d}{dt}L(\nu^*(\theta),\theta)|_{\theta=\theta^*} = \frac{d}{dt}L(\nu,\theta)|_{\nu=\nu^*(\theta)}.$$

It remains to show $\frac{d}{dt}L(\nu,\theta)|_{\nu=\nu^*(\theta)} \leq 0$ and it's non-zero whenever $||\Upsilon_\theta[-\nabla_\theta L(\nu^(\theta),\theta)]|_{\theta=\theta^*}|| \neq 0$ by considering three cases. For convenience, let $\Theta^o = \Theta\backslash\partial\Theta$ denote the interior of the set Θ.

Case 1: When $\theta \in \Theta^o$.

Recall that Θ is a convex compact set, then there exists a sufficiently small $\eta_0 > 0$ such that $\theta - \eta_0\nabla_\theta L(\nu,\theta)|_{\nu=\nu^*(\theta)} \in \Theta$ and

$$\Gamma_\Theta(\theta - \eta_0\nabla_\theta L(\nu,\theta)|_{\nu=\nu^*(\theta)}) - \nu = -\eta_0\nabla_\theta L(\nu,\theta)|_{\nu=\nu^*(\theta)}.$$

Recall that $\Upsilon_\theta[-K(\theta)] = \lim_{0<\eta\to 0}\frac{\Gamma_\Theta(\theta+\eta K(\theta))-\Gamma_\Theta(\theta)}{\eta}$, we have

$$\frac{d}{dt}L(\nu,\theta)|_{\nu=\nu^*(\theta)} = -||\nabla_\theta L(\nu,\theta)|_{\nu=\nu^*(\theta)}||^2 \leq 0$$

and $\frac{d}{dt}L(\nu,\theta)|_{\nu=\nu^*(\theta)}$ is non-zero if $||\nabla_\theta L(\nu,\theta)|_{\nu=\nu^*(\theta)}|| \neq 0$, i.e., $||\Upsilon_\theta[-\nabla_\theta L(\nu,\theta)|_{\nu=\nu^*(\theta)]}|| \neq 0$.

Case 2: When $\theta \in \partial\Theta$, for any $\eta \in (0,\eta_0]$ and some $\eta_0 > 0$, $(\theta - \eta\nabla_\theta L(\nu,\theta)|_{\nu=\nu^*(\theta)}) \in \Theta$.

Since $(\theta - \eta\nabla_\theta L(\nu,\theta)|_{\nu=\nu^*(\theta)}) \in \Theta$, we have

$$\Upsilon_\theta[-\nabla_\theta L(\nu,\theta)|_{\nu=\nu^*(\theta)}] = -\nabla_\theta L(\nu,\theta)|_{\nu=\nu^*(\theta)}$$

and this further implies

$$\frac{d}{dt}L(\nu,\theta)|_{\nu=\nu^*(\theta)} = -||\nabla_\theta L(\nu,\theta)|_{\nu=\nu^*(\theta)}||^2 \leq 0$$

and $\frac{d}{dt}L(\nu,\theta)|_{\nu=\nu^*(\theta)}$ is non-zero if $||\nabla_\theta L(\nu,\theta)|_{\nu=\nu^*(\theta)}|| \neq 0$, i.e., $||\Upsilon_\theta[-\nabla_\theta L(\nu,\theta)|_{\nu=\nu^*(\theta)}]|| \neq 0$.

Case 3: When $\theta \in \partial\Theta$, for some $\eta \in (0, \eta_0]$ and any $\eta_0 > 0$, $(\theta - \eta\nabla_\theta L(\nu,\theta)|_{\nu=\nu^*(\theta)}) \notin \Theta$.

Define $\theta_\eta := \theta - \eta\nabla_\theta L(\nu,\theta)|_{\nu=\nu^*(\theta)}$ for any $\eta > 0$. In this condition, considering the strong convex function $f(\theta) = \frac{1}{2}(\theta - \theta_\eta)^2$, invoking the first order optimality condition, one obtains for all $\theta \in \Theta$,

$$\nabla f(\theta_\eta^*)^{\mathrm{T}}(\theta - \theta^*) = (\theta^* - \theta_\eta)(\theta - \theta_\eta^*) \geq 0.$$

Note that θ_η^* is the projection of ν_η and it's unique due to the strong convexity of $f(\theta)$. Then, we can have

$$\begin{aligned}
\frac{d}{dt}L(\nu,\theta)|_{\nu=\nu^*(\theta)} &= \left(\nabla_\theta L(\nu,\theta)|_{\nu=\nu^*(\theta)}\right)^{\mathrm{T}}\Upsilon_\theta[-\nabla_\theta L(\nu,\theta)|_{\nu=\nu^*(\theta)}] \\
&= \left(\nabla_\theta L(\nu,\theta)|_{\nu=\nu^*(\theta)}\right)^{\mathrm{T}}\left(\lim_{0<\eta\to 0}\frac{\theta_\eta^* - \theta}{\eta}\right) \\
&= \left(\lim_{0<\eta\to 0}\frac{-(\theta_\eta - \theta)}{\eta}\right)\left(\lim_{0<\eta\to 0}\frac{\theta_\eta^* - \theta}{\eta}\right) \\
&= \lim_{0<\eta\to 0}\frac{-||\theta_\eta^* - \theta||^2}{\eta^2} + \lim_{0<\eta\to 0}\frac{(\theta_\eta^* - \theta_\eta)(\theta_\eta^* - \theta)}{\eta^2} \leq 0.
\end{aligned}$$

Based on these arguments, we know that $\frac{d}{dt}\mathcal{L}(\nu) \leq 0$ and it's non-zero whenever $||\Upsilon_\theta[-\nabla_\theta L(\nu,\theta)|_{\nu=\nu^*(\theta)}]|| \neq 0$. Then by applying the Lyapunov theory in the reference, we know that for any initial condition $\theta(0)$, $\theta(t)$ converges to θ^*, i.e., $L(\nu^*(\theta^*),\theta^*) \leq L(\nu^*(\theta(t)),\theta(t)) \leq L(\nu^*(\theta(0)),\theta(0))$ for any $t \geq 0$. Therefore, the sequence $\{\theta_k\}$ generated by following the θ-update converges almost surely to θ^*.

Step 3. Local Minimum

We have shown that $\{\nu_k, \theta_k\}$ converges to $(\nu^*,\theta^*) = (\nu^*(\theta^*),\theta^*)$. Moreover, by the Lyapunov analysis, we know with any initial condition $\nu(0),\theta(0)$, the state trajectories $\nu(t)$ and $\theta(t)$ of (A.6) and (A.7) converges to (ν^*,θ^*) and $L(\nu^*,\theta^*) \leq L(\nu^*(\theta(t)),\theta) \leq L(\nu^*(\theta(0)),\theta(0)) \leq L(\nu(t),\theta(0)) \leq L(\nu(0),\theta(0))$ for any $t \geq 0$.

By contradiction, suppose that (ν^*,θ^*) is not a local minimum. Then there exists $(\bar{\nu},\bar{\theta}) \in$

$[V_{\min}, V_{\max}] \times \Theta \cap \mathcal{B}_{\nu^*,\theta^*}(r)$ such that

$$L(\bar{\nu}, \bar{\theta}) = \min_{(\nu,\theta)\in[V_{\min},V_{\max}]\times\Theta\cap\mathcal{B}_{\nu^*,\theta^*}(r)} L(\nu, \theta).$$

By setting $\theta(0) = \bar{\theta}$, the above equation implies that

$$L(\bar{\nu}, \bar{\theta}) = \min_{(\nu,\theta)\in[V_{\min},V_{\max}]\times\Theta\cap\mathcal{B}_{\nu^*,\theta^*}(r)} L(\nu, \theta) < L(\nu^*, \theta^*) \leq L(\bar{\nu}, \theta(0)) = L(\bar{\nu}, \bar{\theta})$$

which is a contradiction. Therefore, (ν^*, θ^*) is a local minimum.

Appendix B

Technical Results in Chapter 3

B.1 Computation of Gradient Estimates

In this section, we provide the details of the gradient estimate computations. From the definition of $L(\nu, \omega, \theta)$, we obtain the following:

$$\nabla_\nu L(\nu,\omega,\theta) = \omega + \mathbb{E}_P\left[\phi^*\left(\frac{J^\theta(x_0)}{\nu} - \omega + \beta\right)\right] + \nu\nabla_\nu \mathbb{E}_P\left[\phi^*\left(\frac{J^\theta(x_0)}{\nu} - \omega + \beta\right)\right], \tag{B.1}$$

$$\nabla_\omega L(\nu,\omega,\theta) = \nu + \nu\nabla_\omega \mathbb{E}_P\left[\phi^*\left(\frac{J^\theta(x_0)}{\nu} - \omega + \beta\right)\right], \tag{B.2}$$

$$\nabla_\theta L(\nu,\omega,\theta) = \nu\nabla_\theta \mathbb{E}_P\left[\phi^*\left(\frac{J^\theta(x_0)}{\nu} - \omega + \beta\right)\right]. \tag{B.3}$$

1. $\nabla_\nu L(\nu, \omega, \theta)$: by expanding the expectation, we have

$$L(\nu,\omega,\theta) = \nu\left[\omega + \sum_\xi \mathbb{P}_\theta(\xi)\left(\phi^*\left(\frac{J(\xi)}{\nu} - \omega + \beta\right)\right)\right].$$

By taking the gradient w.r.t. ν, we have

$$\begin{aligned}\widehat{\nabla_\nu} L(\nu,\omega,\theta) &= \omega + \sum_\xi \mathbb{P}_\theta(\xi)\phi^*\left(\frac{J(\xi)}{\nu} - \omega + \beta\right) + \nu \sum_\xi \mathbb{P}_\theta(\xi)\nabla_\nu\left[\phi^*\left(\frac{J(\xi)}{\nu} - \omega + \beta\right)\right] \\ &= \omega + \sum_\xi \mathbb{P}_\theta(\xi)\phi^*\left(\frac{J(\xi)}{\nu} - \omega + \beta\right) - \nu \sum_\xi \mathbb{P}_\theta(\xi)\frac{J(\xi)}{\nu^2}\frac{\partial \phi^*}{\partial u}\bigg|_{u=\frac{J(\xi)}{\nu}-\omega+\beta} \\ &= \omega + \sum_\xi \mathbb{P}_\theta(\xi)\phi^*\left(\frac{J(\xi)}{\nu} - \omega + \beta\right) - \sum_\xi \mathbb{P}_\theta(\xi)\frac{J(\xi)}{\nu}\frac{\partial \phi^*}{\partial u}\bigg|_{u=\frac{J(\xi)}{\nu}-\omega+\beta}.\end{aligned}$$

2. $\nabla_\omega L(\nu,\omega,\theta)$: by taking the gradient w.r.t. ω, we have

$$\begin{aligned}\widehat{\nabla_\omega} L(\nu,\omega,\theta) &= \nu + \nu\nabla_\omega\left[\sum_\xi \mathbb{P}_\theta(\xi)\phi^*\left(\frac{J(\xi)}{\nu} - \omega + \beta\right)\right] \\ &= \nu + \nu \sum_\xi \mathbb{P}_\theta(\xi)\nabla_\omega\left[\phi^*\left(\frac{J(\xi)}{\nu} - \omega + \beta\right)\right] \\ &= \nu - \nu \sum_\xi \mathbb{P}_\theta(\xi)\frac{\partial \phi^*}{\partial u}\bigg|_{u=\frac{J(\xi)}{\nu}-\omega+\beta}.\end{aligned}$$

3. $\nabla_\theta L(\nu,\omega,\theta)$: by taking the gradient w.r.t. θ, we have

$$\begin{aligned}\widehat{\nabla_\theta} L(\nu,\omega,\theta) &= \nu\nabla_\theta\left[\sum_\xi \mathbb{P}_\theta(\xi)\phi^*\left(\frac{J(\xi)}{\nu} - \omega + \beta\right)\right] \\ &= \nu \sum_\xi \nabla_\theta \mathbb{P}_\theta(\xi)\phi^*\left(\frac{J(\xi)}{\nu} - \omega + \beta\right) \\ &= \nu \sum_\xi \mathbb{P}_\theta(\xi)\nabla_\theta \log \mathbb{P}_\theta(\xi)\phi^*\left(\frac{J(\xi)}{\nu} - \omega + \beta\right),\end{aligned}$$

where the last equality is due to $\nabla_\theta \mathbb{P}_\theta(\xi) = \mathbb{P}_\theta(\xi)\nabla_\theta \log \mathbb{P}_\theta(\xi)$.

B.2 Proof of Theorem 7

In this section, we provide the detailed proof that was outlined in the proof sketch in the main content. We will begin by analyzing the multi-time scale discrete stochastic approximation and

proceed through the convergence of the sequences $(\nu_k, \omega_k, \theta_k)$ to the local optimal solutions.

B.2.1 Convergence of ν-update

Since ν converges a faster time scale than ω and θ, we can regard ω and θ as fixed in the ν-update, i.e.,

$$\nu_{k+1} = \Gamma_{\mathcal{N}}\left[\nu_k - \zeta_1(k)\left(\omega + \sum_{j=1}^{N}\frac{1}{N}\phi^*\left(\frac{J(\xi_{j,k})}{\nu_k} - \omega + \beta\right) - \sum_{j=1}^{N}\frac{1}{N}\frac{J(\xi_{j,k})}{\nu_k}\frac{\partial\phi^*}{\partial u}\bigg|_{u=\frac{J(\xi_{j,k})}{\nu_k}-\omega+\beta}\right)\right].$$

Consider the continuous time dynamics of ν defined using differential inclusion

$$\dot{\nu} \in \Upsilon_\nu\left[-\widehat{\nabla_\nu} L(\nu, \omega, \theta)\right], \tag{B.4}$$

where

$$\Upsilon_\nu[G(\nu)] := \lim_{0<\eta\to 0}\frac{\Gamma_N(\nu + \eta G(\nu)) - \Gamma_N(\nu)}{\eta}.$$

Here $\Upsilon_\nu[G(\nu)]$ is the left directional derivative of the function $\Gamma_N(\nu)$ in the direction of $G(\nu)$. Using the left directional derivative $\Upsilon_\nu[G(\nu)]$ in the sub-gradient descent algorithm for ν ensures that the gradient points in the descent direction along the boundary of ν whenever the ν-update hits its boundary.

Now consider the following equation,

$$\nu_{k+1} = \Gamma_{\mathcal{N}}\left[\nu_k - \zeta_1(k)\left(\omega + \sum_{\xi}\mathbb{P}_\theta(\xi)\phi^*\left(\frac{J(\xi)}{\nu_k} - \omega + \beta\right) - \sum_{\xi}\mathbb{P}_\theta(\xi)\frac{J(\xi)}{\nu_k}\frac{\partial\phi^*}{\partial u}\bigg|_{u=\frac{J(\xi)}{\nu_k}-\omega+\beta} + \delta_{\nu_{k+1}}\right)\right],$$

where

$$\begin{aligned}\delta_{\nu_{k+1}} = &\left(\omega + \sum_{j=1}^{N}\frac{1}{N}\phi^*\left(\frac{J(\xi_{j,k})}{\nu_k} - \omega + \beta\right) - \sum_{j=1}^{N}\frac{1}{N}\frac{J(\xi_{j,k})}{\nu_k}\frac{\partial\phi^*}{\partial u}\bigg|_{u=\frac{J(\xi_{j,k})}{\nu_k}-\omega+\beta}\right)\\ &- \left(\omega + \sum_{\xi}\mathbb{P}_\theta(\xi)\phi^*\left(\frac{J(\xi)}{\nu_k} - \omega + \beta\right) - \sum_{\xi}\mathbb{P}_\theta(\xi)\frac{J(\xi)}{\nu_k}\frac{\partial\phi^*}{\partial u}\bigg|_{u=\frac{J(\xi)}{\nu_k}-\omega+\beta}\right).\end{aligned}$$

In order to show that the update rule converges to the solution of the o.d.e, we need to verify several conditions. Before going through this process, we firstly make the following assumptions, which will be used to guarantee the convergence of our algorithm.

Assumption 9. The parameters ν and ω are bounded, i.e., $\nu \in [V_{\min}, V_{\max}]$ and $\omega \in [W_{\min}, W_{\max}]$.

Assumption 10. Let $U_{\min}$ and $U_{\max}$ denote the bound for $u = \frac{J(\xi)}{\nu} - \omega + \beta$. The function ϕ satisfies:
1. The first derivative of the conjugate function ϕ^* is bounded in $[U_{\min}, U_{\max}]$.
2. The second derivative of the conjugate function ϕ^* is bounded in $[U_{\min}, U_{\max}]$.

In Lemma 6 in Appendix B.3, we show that $\widehat{\nabla_\nu} L(\nu, \omega, \theta)$ is Lipschitz continuous in ν. Given that the step size ζ_1 satisfies Assumption 5, we have $\sum_k \zeta_1(k) = \infty$ and $\sum_k \zeta_1^2(k) < \infty$. Furthermore, in Lemma 7 in Appendix B.3, we show that the sequence $\{\delta_{\nu_{k+1}}\}$ forms a martingale difference sequence. In addition, under Assumption 9, we have $\sup_k ||\nu_k|| < \infty$. With these conditions, we can invoke Corollary 4 in Chapter 5 of [19] to show that the update rule in our algorithm converges almost surely to the set $[V_{\min}, V_{\max}]$.

To complete the proof of convergence for the ν-update, we must show that the sequence converges to a fixed point of the o.d.e. (B.4). To establish this, we apply a Lyapunov stability analysis.

For any given ω and θ, define the following Lyapunov function

$$\mathcal{L}_{\omega,\theta}(\nu) = L(\nu, \omega, \theta) - L(\nu^*, \omega, \theta),$$

where ν^* is a minimum point.

To utilize the Lyapunov theory for asymptotically stable differential inclusions (Theorem 3.10 and Corollary 3.11 in [14]), we need to verify that the Lyapunov function defined above satisfies both Hypothesis 3.1 and Hypothesis 3.9 from [14].

We begin by verifying that the Lyapunov function satisfies Hypothesis 3.9, which requires showing that $\frac{d}{dt}\mathcal{L}_{\omega,\theta}(\nu) \leq 0$ and $\nabla_t \mathcal{L}_{\omega,\theta}(\nu)$ is non-zero if $\left\| \Gamma_N[-\widehat{\nabla_\nu} L(\nu, \omega, \theta)] \right\| \neq 0$. Considering

the continuous-time dynamics for ν, we have

$$\frac{d}{dt}L(\nu,\omega,\theta) = \widehat{\nabla_\nu} L(\nu,\omega,\theta)\Upsilon_N\left[-\widehat{\nabla_\nu} L(\nu,\omega,\theta)\right].$$

Therefore, we obtain

$$\begin{aligned}\frac{d}{dt}\mathcal{L}_{\theta,\lambda}(\nu) &= \widehat{\nabla_\nu} L(\nu,\omega,\theta)\Upsilon_\nu\left[-\widehat{\nabla_\nu} L(\nu,\omega,\theta)\right] - \widehat{\nabla_\nu} L(\nu,\omega,\theta)\Upsilon_\nu\left[-\widehat{\nabla_\nu} L(\nu,\omega,\theta)\right]\Big|_{\nu=\nu^*} \\ &= \widehat{\nabla_\nu} L(\nu,\omega,\theta)\Upsilon_\nu\left[-\widehat{\nabla_\nu} L(\nu,\omega,\theta)\right] \\ &= \frac{d}{dt}\widehat{\nabla_\nu} L(\nu,\omega,\theta).\end{aligned}$$

We need to demonstrate that $\frac{d}{dt}\mathcal{L}_{\omega,\theta}(\nu) \leq 0$ and that this quantity is non-zero whenever

$$\left\|\Gamma_N[-\widehat{\nabla_\nu} L(\nu,\omega,\theta)]\right\| \neq 0.$$

Case 1: $\nu \in (V_{\min}, V_{\max})$.

There exists a sufficiently small $\eta_0 > 0$ such that $\nu - \eta_0\widehat{\nabla_\nu} L(\nu,\omega,\theta) \in [V_{\min}, V_{\max}]$ and

$$\Gamma_N\left[\nu - \eta_0\widehat{\nabla_\nu} L(\nu,\omega,\theta)\right] = -\eta_0\widehat{\nabla_\nu} L(\nu,\omega,\theta).$$

Recalling the definition of $\Upsilon\left[-\widehat{\nabla_\nu} L(\nu,\omega,\theta)\right]$, we obtain

$$\frac{d}{dt}L(\nu,\theta,\lambda) = -\left\|\widehat{\nabla_\nu} L(\nu,\omega,\theta)\right\| \leq 0$$

and $\frac{d}{dt}L(\nu,\theta,\lambda) < 0$ if $\widehat{\nabla_\nu} L(\nu,\omega,\theta) \neq 0$.

Case 2: $\nu \in \{V_{\min}, V_{\max}\}$.

Notice that there are two cases, which depend on whether the set

$$F(\nu) := \left\{\widehat{\nabla_\nu} L(\nu,\omega,\theta)\middle|\forall\eta_0 > 0, \exists\eta \in [0,\eta_0] \text{ such that } \nu - \eta\widehat{\nabla_\nu} L(\nu,\omega,\theta) \notin [V_{\min}, V_{\max}]\right\}$$

is empty or not.

Case 2-1: $F(\nu)$ is empty.

Since $\nu \in \{V_{\min}, V_{\max}\}$ and $\nu - \eta\widehat{\nabla}_\nu L(\nu,\omega,\theta) \in [V_{\min}, V_{\max}]$, we know

$$\Upsilon_\nu\left[-\widehat{\nabla}_\nu L(\nu,\omega,\theta)\right] = -\widehat{\nabla}_\nu L(\nu,\omega,\theta),$$

which implies that

$$\frac{d}{dt}L(\nu,\theta,\lambda) = -\left\|\widehat{\nabla}_\nu L(\nu,\omega,\theta)\right\| \leq 0$$

and $\frac{d}{dt}L(\nu,\theta,\lambda) < 0$ if $\widehat{\nabla}_\nu L(\nu,\omega,\theta) \neq 0$.

Case 2-2: $F(\nu)$ is not empty.

For any $\eta > 0$, define $\nu_\eta := \nu - \eta\widehat{\nabla}_\nu L(\nu,\omega,\theta)$. When $0 < \eta \to 0$, $\Gamma_N[\nu_\eta]$ is the projection of ν_η to the tangent space of $[V_{\min}, V_{\max}]$. For any $\hat{\nu} \in [V_{\min}, V_{\max}]$, since the set $\{\nu \in [V_{\min}, V_{\max}] : ||\nu - \nu_\eta||_2 \leq ||\hat{\nu} - \nu_\eta||_2\}$ is compact, then the project of ν_η on $[V_{\min}, V_{\max}]$ exists. Furthermore, since $g(\nu) = \frac{1}{2}(\nu - \nu_\eta)^2$ is a strongly convex function and $\nabla_\nu g(\nu) = \nu - \nu_\eta$. By the first order optimal condition, we obtains $\forall \nu \in [V_{\min}, V_{\max}]$,

$$\nabla g(\nu_\eta^*)(\nu - \nu_\eta^*) = (\nu_\eta^* - \nu_\eta)(\nu - \nu_\eta^*) \geq 0,$$

where ν_η^* is the unique projection of ν_η. Due to the uniqueness, we know only if $\nu = \nu_\eta^*$, the above equality holds. Therefore, for any $\nu \in [V_{\min}, V_{\max}]$ and $\eta > 0$,

$$\begin{aligned}
\widehat{\nabla}_\nu L(\nu,\omega,\theta)\Upsilon_\nu\left[-\widehat{\nabla}_\nu L(\nu,\omega,\theta)\right] &= \widehat{\nabla}_\nu L(\nu,\omega,\theta)\lim_{0<\eta\to 0}\frac{\nu_\eta^* - \nu}{\eta} \\
&= \lim_{0<\eta\to 0}\frac{\nu - \nu_\eta}{\eta}\lim_{0<\eta\to 0}\frac{\nu_\eta^* - \nu}{\eta} \\
&= \lim_{0<\eta\to 0}\frac{-||\nu_\eta^* - \nu||^2}{\eta^2} + \lim_{0<\eta\to 0}(\nu^* - \nu_\eta)\frac{\nu^* - \nu}{\eta^2} \leq 0.
\end{aligned}$$

Note that for any $\widehat{\nabla}_\nu L(\nu,\omega,\theta) \cap F(\nu)^c$, $\nu - \eta\widehat{\nabla}_\nu L(\nu,\omega,\theta) \in [V_{\min}, V_{\max}]$ for any $\eta \in [0, \eta_0]$ and some $\eta_0 > 0$. Thus, this follows the statement in the empty case.

Combining all these arguments, we conclude that $\frac{d}{dt}L(\nu,\omega,\theta) \leq 0$, and this inequality holds strictly whenever $\Upsilon_\nu\left[-\widehat{\nabla_\nu}L(\nu,\omega,\theta)\right] \neq 0$. As a result, $\frac{d}{dt}\mathcal{L}_{\omega,\theta}(\nu) \leq 0$ and remains non-zero under the same condition.

Having shown that $\mathcal{L}_{\omega,\theta}(\nu)$ satisfies Hypotheses 3.1 and 3.9, we can now apply the results from [14]. This ensures that the ν-update converges almost surely to the solution of the o.d.e. (B.4), which, in turn, converges to $\nu^* \in [V_{\min}, V_{\max}]$.

B.2.2 Convergence of ω-update

After establishing the convergence of the ν-update, we proceed to demonstrate the convergence of the ω-update. Given that ν converges on a faster timescale than ω, and θ operates on the slowest timescale, the ω-update can be expressed using the converged value $\nu^*(\omega)$ while treating θ as an invariant quantity, i.e.,

$$\omega_{k+1} = \Gamma_{\mathcal{R}}\left[\omega_k - \zeta_2(k)\left(\nu^*(\omega_k) - \nu^*(\omega_k)\sum_{j=1}^{N}\frac{1}{N}\frac{\partial\phi^*}{\partial u}\bigg|_{u=\frac{J(\xi_{j,k})}{\nu^*(\omega_k)}-\omega_k+\beta}\right)\right].$$

Considering the continuous time dynamic of ω,

$$\dot{\omega} \in \Upsilon_\omega\left[-\widehat{\nabla_\omega}L(\nu,\omega,\theta)\right], \tag{B.5}$$

where

$$\Upsilon_\omega[G(\omega)] := \lim_{0<\eta\to 0}\frac{\Gamma_{\mathcal{R}}(\omega+\eta G(\omega)) - \Gamma_{\mathcal{R}}(\omega)}{\eta}.$$

The ω-update can be rewritten as a stochastic approximation, i.e.,

$$\omega_{k+1} = \Gamma_{\mathcal{R}}\left[\omega_k - \zeta_2(k)\left(\nu^*(\omega_k) - \nu^*(\omega_k)\cdot\sum_{\xi}\mathbb{P}_\theta(\xi)\frac{\partial\phi^*}{\partial u}\bigg|_{u=\frac{J(\xi)}{\nu^*(\omega_k)}-\omega_k+\beta}\right) + \delta_{\omega_{k+1}}\right],$$

where

$$\delta_{\omega_{k+1}} = -\left(\nu^*(\omega_k) - \nu^*(\omega_k)\sum_{\xi}\mathbb{P}_\theta(\xi)\frac{\partial\phi^*}{\partial u}\bigg|_{u=\frac{J(\xi)}{\nu^*(\omega_k)}-\omega_k+\beta}\right) + \left(\nu^*(\omega_k) - \nu^*(\omega_k)\sum_{j=1}^{N}\frac{1}{N}\frac{\partial\phi^*}{\partial u}\bigg|_{u=\frac{J(\xi_{j,k})}{\nu^*(\omega_k)}-\omega_k+\beta}\right).$$

To demonstrate that the update rule converges to the solution of the o.d.e., we need to verify conditions similar to those established previously. In particular, in Lemma 8 of Appendix B.3, we show that $\widehat{\nabla_\omega} L(\nu, \omega, \theta)$ is Lipschitz continuous in ω. The step size $\zeta_2(k)$ satisfies $\sum_k \zeta_2(k) = \infty$ and $\sum_k \zeta_2^2(k) < \infty$, as stated in Assumption 5. Moreover, Assumption 9 ensures that $\sup_k ||\omega_k|| < \infty$.

Next, we focus on the Lyapunov analysis for the ω-update. For any fixed θ, we define the Lyapunov function as:

$$\mathcal{L}_\theta(\omega) = L(\nu^*(\omega), \omega, \theta) - L(\nu^*(\omega), \omega^*, \theta),$$

where ω^* is a local minimum point. Analogous to the approach used for the ν-update, we can express:

$$\frac{d}{dt}\mathcal{L}_\theta(\omega) = \frac{d}{dt}\widehat{\nabla_\omega} L(\nu^*(\omega), \omega, \theta).$$

Following a method similar to the Lyapunov analysis for the ν-update, we can show that $\frac{d}{dt}\mathcal{L}_\theta(\omega) \leq 0$ and that this quantity is strictly non-zero whenever $\left\|\Gamma_{\mathcal{R}}[-\widehat{\nabla_\omega} L(\nu, \omega, \theta|_{\nu=\nu^*(\omega)})]\right\| \neq 0$. Consequently, we demonstrate that the ω-update converges almost surely to the solution of the o.d.e. (B.4), which, in turn, converges to $\omega^* \in [W_{\min}, W_{\max}]$.

B.2.3 Local minimum

In this section, we aim to establish the convergence of the sequence $\{\nu_k, \omega_k\}$ towards a local minimum of the objective function $L(\nu, \omega, \theta)$, while keeping θ fixed. Building upon the arguments

presented in the previous sections, we show that, for any given initial states $\nu(0)$ and $\omega(0)$, the sequences $\nu(t)$ and $\omega(t)$ converge to their respective optimal stationary points, ν^* and ω^*. This further implies

$$\begin{aligned} L(\nu^*, \omega^*, \theta) &\leq L(\nu(\omega^*(t)), \omega(t), \theta) \\ &\leq L(\nu(\omega(0)), \omega(0), \theta) \\ &\leq L(\nu(t), \omega(0), \theta) \\ &\leq L(\nu(0), \omega(0), \theta). \end{aligned}$$

We demonstrate the existence of a local minimum through contraction.

Suppose that (ν^*, ω^*) is not a local minimum, then there exits a point $(\bar{\nu}, \bar{\omega}) \in [V_{\min}, V_{\max}] \times [W_{\min}, W_{\max}] \cap \mathcal{B}_{(\nu^*, \omega^*)}(r)$ such that

$$L(\bar{\nu}, \bar{\omega}, \theta) = \min_{(\nu,\omega) \in [V_{\min}, V_{\max}] \times [W_{\min}, W_{\max}] \cap \mathcal{B}_{(\nu^*, \omega^*)}(r)} L(\nu, \omega, \theta).$$

The minimum is attained by the Weierstrass extreme value theorem. By setting $\omega(0) = \bar{\omega}$, we have

$$\begin{aligned} L(\bar{\nu}, \bar{\omega}, \theta) &= \min_{(\nu,\omega) \in [V_{\min}, V_{\max}] \times [W_{\min}, W_{\max}] \cap \mathcal{B}_{(\nu^*, \omega^*)}(r)} L(\nu, \omega, \theta) \\ &\leq L(\nu^*, \omega^*, \theta) \\ &\leq L(\bar{\nu}, \bar{\omega}, \theta), \end{aligned}$$

which is a contraction.

Therefore, (ν^*, ω^*) is a local minimum for $L(\nu, \omega, \theta)$ for any fixed θ.

B.2.4 Convergence of θ-update

Given that θ converges on the slowest timescale, we can express the θ-update as:

$$\theta_{k+1} = \Gamma_\Theta\left[\theta_k - \zeta_3(k)\left(\nu^*(\theta)\sum_{j=1}^{N}\frac{1}{N}\nabla_\theta \log \mathbb{P}_\theta(\xi_{j,k})\cdot\phi^*\left(\frac{J(\xi_{j,k})}{\nu^*(\theta)} - \omega^*(\theta) + \beta\right)\right)\right].$$

We now consider the following o.d.e. for θ:

$$\dot{\theta} \in \Upsilon_\theta\left[-\widehat{\nabla}_\theta L(\nu,\omega,\theta)\right], \tag{B.6}$$

where

$$\Upsilon_\theta[G(\theta)] := \lim_{0<\eta\to 0}\frac{\Gamma_\Theta(\theta+\eta G(\theta)) - \Gamma_\Theta(\theta)}{\eta}.$$

The θ-update can be rewritten as a stochastic approximation, i.e.,

$$\theta_{k+1} = \Gamma_\Theta\left[\theta_k - \zeta_3(k)\cdot\left(\widehat{\nabla}_\theta L(\nu,\omega,\theta)\big|_{\nu=\nu^*(\theta_i),\omega=\omega_k,\theta=\theta_k} + \delta_{\theta_{k+1}}\right)\right],$$

where

$$\begin{aligned}\delta_{\theta_{k+1}} = &-\nu^*(\theta)\sum_{\xi}\mathbb{P}_\theta(\xi)\nabla_\theta\log\mathbb{P}_\theta(\xi)\cdot\phi^*\left(\frac{J(\xi)}{\nu^*(\theta)} - \omega^*(\theta_k) + \beta\right)\\ &+\nu^*(\theta)\sum_{\xi}\frac{1}{N}\nabla_\theta\log\mathbb{P}_\theta(\xi)\cdot\phi^*\left(\frac{J(\xi)}{\nu^*(\theta)} - \omega^*(\theta_k) + \beta\right).\end{aligned}$$

To demonstrate that the update rule converges to the solution of the o.d.e., we need to verify several conditions. First, in Lemma 10 in Appendix B.3, we show that $\widehat{\nabla}_\theta L(\nu,\omega,\theta)$ is Lipschitz continuous in θ. Second, the step size $\zeta_3(k)$ satisfies $\sum_k \zeta_3(k) = \infty$ and $\sum_k \zeta_3^2(k) < \infty$, which follows from Assumption 5. Additionally, in Lemma 11 of Appendix B.3, we show that $\{\delta_{\omega_{k+1}}\}$ forms a martingale difference sequence. Finally, θ is in a compact and closed set Θ, which ensures that $\sup_k ||\theta_k|| < \infty$.

It remains to check the Lyapunov analysis for θ-update. The general idea here is same with the

Lyapunov analysis above, but the difference here is that θ is vector other than a scalar. We first define the Lyapunov function

$$\mathcal{L}(\theta) = L(\nu^*(\theta), \omega^*(\theta), \theta) - L(\nu^*(\theta^*), \omega^*(\theta^*), \theta^*),$$

where θ^* is a local minimum point. Consider the continuous time dynamics for θ, we have

$$\frac{d}{dt}\mathcal{L}(\theta) = \frac{d}{dt}\widehat{\nabla}_\theta L(\nu^*(\theta), \omega^*(\theta), \theta).$$

It remains to show that $\frac{d}{dt}\widehat{\nabla}_\theta L(\nu^*(\theta), \omega^*(\theta), \theta) \leq 0$ and the equality holds if and only if

$$\Upsilon_\theta \left[-\widehat{\nabla}_\theta L(\nu, \omega, \theta)\big|_{\nu=\nu^*(\theta), \omega=\omega^*(\theta)} \right] = 0.$$

There are three cases we have to consider.

Case 1: θ is in the interior of Θ (not on the boundary).

Since Θ is a compact closed set, there exists a sufficient small $\eta_0 > 0$ such that

$$\theta - \eta_0 \widehat{\nabla}_\theta L(\nu, \omega, \theta)\big|_{\nu=\nu^*(\theta), \omega=\omega^*(\theta)} \in \Theta$$

and

$$\Gamma_\Theta \left(\theta - \eta_0 \widehat{\nabla}_\theta L(\nu, \omega, \theta)\big|_{\nu=\nu^*(\theta), \omega=\omega^*(\theta)} \right) - \theta = -\eta_0 \widehat{\nabla}_\theta L(\nu, \omega, \theta)\big|_{\nu=\nu^*(\theta), \omega=\omega^*(\theta)}.$$

Recall the definition of Υ_θ, we have

$$\frac{d}{dt} L(\nu, \omega, \theta)\bigg|_{\nu=\nu^*(\theta), \omega=\omega^*(\theta)} = -\left\| \widehat{\nabla}_\theta L(\nu, \omega, \theta)|_{\nu=\nu^*(\theta), \omega=\omega^*(\theta)} \right\|^2 \leq 0.$$

Furthermore, the equality only holds when $\frac{d}{dt} L(\nu, \omega, \theta)|_{\nu=\nu^*(\theta), \omega=\omega^*(\theta)} = 0$.

Case 2: θ is on the boundary of Θ and $\theta - \eta \widehat{\nabla}_\theta L(\nu, \omega, \theta)|_{\nu=\nu^*(\theta), \omega=\omega^*(\theta)} \in \Theta$ for any $\eta \in (0, \eta_0]$

and some $\eta_0 > 0$.

In this case, we have

$$\Upsilon_\theta \left[-\widehat{\nabla}_\theta L(\nu, \omega, \theta)\big|_{\nu=\nu^*(\theta),\omega=\omega^*(\theta)} \right] = -\widehat{\nabla}_\theta L(\nu, \omega, \theta)\big|_{\nu=\nu^*(\theta),\omega=\omega^*(\theta)}.$$

Therefore, we obtain

$$\frac{d}{dt} L(\nu, \omega, \theta)\bigg|_{\nu=\nu^*(\theta),\omega=\omega^*(\theta)} = -\left\| \widehat{\nabla}_\theta L(\nu, \omega, \theta)\big|_{\nu=\nu^*(\theta),\omega=\omega^*(\theta)} \right\|^2 \leq 0.$$

Moreover, the equality only holds when $\widehat{\nabla}_\theta L(\nu, \omega, \theta)|_{\nu=\nu^*(\theta),\omega=\omega^*(\theta)} = 0$.

Case 3: θ is on the boundary of Θ but $\theta - \eta\widehat{\nabla}_\theta L(\nu, \omega, \theta)|_{\nu=\nu^*(\theta),\omega=\omega^*(\theta)} \notin \Theta$ for some $\eta \in (0, \eta_0]$ and any $\eta_0 > 0$.

For any $\eta > 0$, we define

$$\theta_\eta = \theta - \eta\widehat{\nabla}_\theta L(\nu, \omega, \theta)|_{\nu=\nu^*(\theta),\omega=\omega^*(\theta)}.$$

In this case, when $0 < \eta \to 0$, $\Gamma_\Theta[\theta_\eta]$ is the projection of θ_η to the tangent space of Θ. For any $\hat{\theta} \in \Theta$, since $\{\theta \in \Theta : ||\theta - \theta_\eta||_2 \leq \left\|\hat{\theta} - \theta_\eta\right\|_2\}$ is a compact set, the project of θ_η exists. Define $g(\theta) = \frac{1}{2}||\theta - \theta_\eta||_2^2$, since $g(\theta)$ is a strong convex function and $\nabla_\theta g(\theta) = \theta - \theta_\eta$, we obtain

$$\nabla g(\theta_\eta^*)^\top (\theta - \theta_\eta^*) = (\theta_\eta^* - \theta_\eta)^\top (\theta - \theta_\eta^*) \geq 0,$$

for any $\theta \in \Theta$, where θ_η^* is the projection of θ_η. Due to the uniqueness of this projection, the

equality holds if and only if $\theta = \theta^*_\eta$. Therefore, for any $\theta \in \Theta$ and $\eta > 0$,

$$\begin{aligned}
&\left(\widehat{\nabla}_\theta L(\nu,\omega,\theta)\big|_{\nu=\nu^*(\theta),\omega=\omega^*(\theta)}\right)^\top \cdot \Upsilon_\nu\left[-\widehat{\nabla}_\nu L(\nu,\omega,\theta)\big|_{\nu=\nu^*(\theta),\omega=\omega^*(\theta)}\right] \\
&= \left(\widehat{\nabla}_\theta L(\nu,\omega,\theta)\big|_{\nu=\nu^*(\theta),\omega=\omega^*(\theta)}\right)^\top \lim_{0<\eta\to 0} \frac{\theta^*_\eta - \theta}{\eta} \\
&= \left(\lim_{0<\eta\to 0} \frac{\theta - \theta_\eta}{\eta}\right)^\top \lim_{0<\eta\to 0} \frac{\theta^*_\eta - \theta}{\eta} \\
&= \lim_{0<\eta\to 0} \frac{-||\theta^*_\eta - \theta||^2}{\eta^2} + \lim_{0<\eta\to 0} (\theta^* - \theta_\eta)^\top \frac{\theta^* - \theta}{\eta^2} \leq 0.
\end{aligned}$$

Combining all these arguments, we have $\frac{d}{dt}L(\nu^*(\theta),\omega^*(\theta),\theta) \leq 0$ and it is non-zero whenever

$$\Upsilon_\theta\left[-\widehat{\nabla}_\theta L(\nu,\omega,\theta)\big|_{\nu=\nu^*(\theta),\omega=\omega^*(\theta)}\right] \neq 0.$$

Therefore, we know that $\frac{d}{dt}\mathcal{L}(\theta) \leq 0$ and it is non-zero whenever $\Upsilon_\theta\left[-\widehat{\nabla}_\theta L(\nu,\omega,\theta)\big|_{\nu=\nu^*(\theta),\omega=\omega^*(\theta)}\right] \neq 0$. Now, we can establish the almost sure convergence of the θ-update to the solution of the o.d.e given by equation (B.6), which in turn converges to $\theta^* \in \Theta$.

Combining with the fact that (ν^*,ω^*) are local minimum for $L(\nu,\omega,\theta)$, we further conclude that θ^* is a local optimal policy for the ϕ-divergence optimization problem.

B.3 Technical Lemmas

In this section, we present the technical lemmas that are used in the convergence analysis in the proof of Theorem 7. We begin by introducing the following propositions, derived from the definition of $\mathbb{P}_\theta$, which are crucial for demonstrating that the gradient estimates in Algorithm 3.1 are Lipschitz continuous. These results further aid in establishing the technical lemmas that will be discussed later.

Proposition 5. By the definition of $\mathbb{P}_\theta(\xi)$ and $\nabla_\theta \log \mathbb{P}_\theta(\xi)$, we have

$$\begin{aligned}
&\mathbb{P}_\theta(\xi)\nabla_\theta \log \mathbb{P}_\theta(\xi) \\
&= P_0(x_0)\prod_{k=0}^{T-1}\pi(a_k|x_k,\theta)P(x_{k+1}|x_k,a_k)\sum_{k=0}^{T-1}\frac{\nabla_\theta\pi(a_k|x_k,\theta)}{\pi(a_k|x_k,\theta)} \\
&= P_0(x_0)\sum_{k=0}^{T-1}\prod_{i\neq k}^{T-1}\nabla_\theta\pi(a_k|x_k,\theta)\pi(a_k|x_k,\theta)P(x_{k+1}|x_k,a_k).
\end{aligned}$$

Combining Assumption 3 and the fact that the sum of products of Lipschitz function is Lipschitz, $\mathbb{P}_\theta(\xi)\nabla_\theta \log \mathbb{P}_\theta(\xi)$ and $\sum_\xi \mathbb{P}_\theta(\xi)\nabla_\theta \log \mathbb{P}_\theta(\xi)$ are Lipschitz in θ. Furthermore, since the gradient of Lipschitz function is bounded, we have

$$\left|\nabla_\theta(\sum_\xi \mathbb{P}_\theta(\xi)\nabla_\theta \log \mathbb{P}_\theta(\xi))\right| \leq K_1(\xi).$$

Also,

$$\mathbb{E}[\nabla_\theta \log \mathbb{P}_\theta(\xi)] = \sum_\xi \mathbb{P}_\theta(\xi)\nabla_\theta \log \mathbb{P}_\theta(\xi) = 0.$$

Proposition 6. By Assumption 3, $\nabla_\theta \log \mathbb{P}_\theta(\xi)$ is bounded, i.e., $|\nabla_\theta \log \mathbb{P}_\theta(\xi)| \leq K_2(\xi)$.

Lemma 6. $\widehat{\nabla_\nu} L(\nu,\omega,\theta)$ is Lipschitz in ν.

Proof. Recall that

$$\widehat{\nabla_\nu} L(\nu,\omega,\theta) = \omega + \sum_\xi \mathbb{P}_\theta(\xi)\phi^*\left(\frac{J(\xi)}{\nu} - \omega + \beta\right) - \sum_\xi \mathbb{P}_\theta(\xi)\frac{J(\xi)}{\nu}\frac{\partial\phi^*}{\partial u}\bigg|_{u=\frac{J(\xi)}{\nu}-\omega+\beta}.$$

Let $f(\nu)$ denote $\widehat{\nabla_\nu} L(\nu, \omega, \theta)$, we have

$$
\begin{aligned}
f'(\nu) &= \sum_{\xi} \mathbb{P}_\theta(\xi) \nabla_\nu \left(\phi^* \left(\frac{J(\xi)}{\nu} - \omega + \beta \right) \right) - \sum_{\xi} \mathbb{P}_\theta(\xi) \nabla_\nu \left(\frac{J(\xi)}{\nu} \frac{\partial \phi^*}{\partial u} \Big|_{u=\frac{J(\xi)}{\nu}-\omega+\beta} \right) \\
&= -\sum_{\xi} \mathbb{P}_\theta(\xi) \frac{J(\xi)}{\nu^2} \frac{\partial \phi^*}{\partial u} \Big|_{u=\frac{J(\xi)}{\nu}-\omega+\beta} \\
&\quad - \sum_{\xi} \mathbb{P}_\theta(\xi) \left(- \frac{J(\xi)}{\nu^2} \frac{\partial \phi^*}{\partial u} \Big|_{u=\frac{J(\xi)}{\nu}-\omega+\beta} - \frac{J^2(\xi)}{\nu^3} \frac{\partial \phi^*}{\partial u^2} \Big|_{u=\frac{J(\xi)}{\nu}-\omega+\beta} \right) \\
&= -\sum_{\xi} \mathbb{P}_\theta(\xi) \frac{J(\xi)}{\nu^2} \frac{\partial \phi^*}{\partial u} \Big|_{u=\frac{J(\xi)}{\nu}-\omega+\beta} + \sum_{\xi} \mathbb{P}_\theta(\xi) \left(\frac{J(\xi)}{\nu^2} + \frac{J^2(\xi)}{\nu^3} \right) \frac{\partial \phi^*}{\partial u} \Big|_{u=\frac{J(\xi)}{\nu}-\omega+\beta} \\
&= \sum_{\xi} \mathbb{P}_\theta(\xi) \frac{J^2(\xi)}{\nu^3} \frac{\partial \phi^*}{\partial u^2} \Big|_{u=\frac{J(\xi)}{\nu}-\omega+\beta}.
\end{aligned}
$$

Notice that $J(\xi)$ is bounded by $[-\frac{C_{\max}}{1-\gamma}, \frac{C_{\max}}{1-\gamma}]$, ν is bounded by $[V_{\min}, V_{\max}]$ and ω is bounded by $[W_{\min}, W_{\max}]$. By Assumption 10, we know that $f'(\nu)$ is bounded. Thus, $\widehat{\nabla_\nu} L(\nu, \omega, \theta)$ is Lipschitz in ν. □

Lemma 7. $\{\delta_{\nu_{k+1}}\}$ is a martingale difference sequence.

Proof. Due to the fact that the trajectories are generated based on the sampling p.m.f and all these trajectories are independent, we have $\mathbb{E}[\delta_{\nu_{k+1}} | \mathcal{F}_{\nu,k}] = 0$ where $\mathcal{F}_{\nu,k} = \sigma(\nu_m, \delta_{\nu_m}, m \leq k)$ is the filtration of ν_k generated by different independent trajectories.

We need to prove that $\mathbb{E}[||\delta_{\nu_{k+1}}||^2 | \mathcal{F}_{\nu,k}]$ is bounded. Consider

$$
\begin{aligned}
\delta_{\nu_{k+1}} &= \left(\omega + \sum_{j=1}^{N} \frac{1}{N} \phi^* \left(\frac{J(\xi_{j,k})}{\nu_k} - \omega + \beta \right) - \sum_{j=1}^{N} \frac{1}{N} \frac{J(\xi_{j,k})}{\nu_k} \frac{\partial \phi^*}{\partial u} \Big|_{u=\frac{J(\xi_{j,k})}{\nu_k}-\omega+\beta} \right) \\
&\quad - \left(\omega + \sum_{\xi} \mathbb{P}_\theta(\xi) \phi^* \left(\frac{J(\xi)}{\nu_k} - \omega + \beta \right) - \sum_{\xi} \mathbb{P}_\theta(\xi) \frac{J(\xi)}{\nu_k} \frac{\partial \phi^*}{\partial u} \Big|_{u=\frac{J(\xi)}{\nu_k}-\omega+\beta} \right) \\
&= -\sum_{\xi} \mathbb{P}_\theta(\xi) \phi^* \left(\frac{J(\xi)}{\nu_k} - \omega + \beta \right) + \sum_{j=1}^{N} \frac{1}{N} \phi^* \left(\frac{J(\xi_{j,k})}{\nu_k} - \omega + \beta \right) \\
&\quad - \sum_{j=1}^{N} \frac{1}{N} \frac{J(\xi_{j,k})}{\nu_k} \frac{\partial \phi^*}{\partial u} \Big|_{u=\frac{J(\xi_{j,k})}{\nu_k}-\omega+\beta} + \sum_{\xi} \mathbb{P}_\theta(\xi) \frac{J(\xi)}{\nu_k} \frac{\partial \phi^*}{\partial u} \Big|_{u=\frac{J(\xi)}{\nu_k}-\omega+\beta}.
\end{aligned}
$$

Notice that ϕ^* is a convex function and $\frac{J(\xi_{j,k})}{\nu_k} - \omega + \beta$ is bounded. Then, $\phi^*(\frac{J(\xi_{j,k})}{\nu_k} - \omega + \beta)$ is bounded. For convenience, we denote it as $\phi^*(\frac{J(\xi_{j,k})}{\nu_k} - \omega + \beta) \in [c_1, c_2]$, where $c_1, c_2 \in \mathbb{R}$. By Assumption 10, we have $\frac{\partial \phi^*}{\partial u}\Big|_{u=\frac{J(\xi)}{\nu_k}-\omega+\beta} \in [c_3, c_4]$, where $c_3, c_4 \in \mathbb{R}$. Then, we have

$$\delta_{\nu_{k+1}} \leq c_2 - c_1 + \frac{C_{\max}}{(1-\gamma)V_{\min}}(c_4 - c_3).$$

Let $c_5 \in \mathbb{R}$ denote the real value on the right side, we further have, $||\delta_{\nu_{k+1}}||^2 \leq (c_5)^2$, which implies $\{\delta_{\nu_{k+1}}\}$ is a martingale difference sequence. □

Lemma 8. $\widehat{\nabla_\omega} L(\nu, \omega, \theta)$ is Lipschitz in ω.

Proof. Recall that

$$\widehat{\nabla_\omega} L(\nu, \omega, \theta) = \nu - \nu \sum_{\xi} \mathbb{P}_\theta(\xi) \frac{\partial \phi^*}{\partial u}\Big|_{u=\frac{J(\xi)}{\nu}-\omega+\beta}.$$

For convenience, denote $\widehat{\nabla_\omega} L(\nu, \omega, \theta)$ as $f(\omega)$. We have

$$f'(\omega) = \nu \sum_{\xi} \mathbb{P}_\theta(\xi) \frac{\partial \phi^*}{\partial u^2}\Big|_{u=\frac{J(\xi)}{\nu}-\omega+\beta}.$$

Recall that the second derivative of ϕ^* is bounded in a closed set and ν is also bounded. We know $f'(\omega)$ is bounded, thus, $\widehat{\nabla_\omega} L(\nu, \omega, \theta)$ is Lipschitz in ω. □

Lemma 9. $\{\delta_{\omega_{k+1}}\}$ is a martingale difference sequence.

Proof. Note that the trajectories are generated based on the sampling p.m.f and all these trajectories are independent, we have $\mathbb{E}[\delta_{\omega_{k+1}} | \mathcal{F}_{\nu,k}] = 0$ where $\mathcal{F}_{\omega,k} = \sigma(\omega_m, \delta_{\omega_m}, m \leq k)$ is the filtration of ω_k generated by different independent trajectories.

We now demonstrate that $\mathbb{E}[||\delta_{\omega_{k+1}}||^2|\mathcal{F}_{\nu,k}]$ is bounded. Consider

$$
\begin{aligned}
&\delta_{\omega_{k+1}} \\
&= -\left(\nu^*(\omega_k) - \nu^*(\omega_k)\sum_{\xi}\mathbb{P}_\theta(\xi)\frac{\partial\phi^*}{\partial u}\bigg|_{u=\frac{J(\xi)}{\nu^*(\omega_k)}-\omega_k+\beta}\right) + \left(\nu^*(\omega_k) - \nu^*(\omega_k)\sum_{j=1}^{N}\frac{1}{N}\frac{\partial\phi^*}{\partial u}\bigg|_{u=\frac{J(\xi_{j,k})}{\nu^*(\omega_k)}-\omega_k+\beta}\right) \\
&= \nu^*(\omega_k)\sum_{\xi}\mathbb{P}_\theta(\xi)\frac{\partial\phi^*}{\partial u}\bigg|_{u=\frac{J(\xi)}{\nu^*(\omega_k)}-\omega_k+\beta} - \nu^*(\omega_k)\sum_{j=1}^{N}\frac{1}{N}\frac{\partial\phi^*}{\partial u}\bigg|_{u=\frac{J(\xi_{j,k})}{\nu^*(\omega_k)}-\omega_k+\beta}.
\end{aligned}
$$

Since the first derivative of ϕ^* is bounded in a closed set, for convenience, denote its bound as $[c_6, c_7]$, where $c_6, c_7 \in \mathbb{R}$, we have

$$\delta_{\omega_{k+1}} \leq V_{\max}|c_7 - c_6|.$$

Thus, $||\delta_{\omega_{k+1}}||^2$ is bounded, which further implies $\{\delta_{\omega_{k+1}}\}$ is a martingale difference sequence. □

Lemma 10. $\widehat{\nabla}_\theta L(\nu, \omega, \theta)$ is Lipschitz in θ.

Proof. Recall that

$$\widehat{\nabla}_\theta L(\nu, \omega, \theta) = \sum_{\xi}\mathbb{P}_\theta(\xi)\nabla_\theta \log \mathbb{P}_\theta(\xi)\phi^*\left(\frac{J(\xi)}{\nu} - \omega + \beta\right).$$

By Assumption 3 and 10, we know that $\nabla_\theta \mathbb{P}_\theta(\xi)$ and $\nabla_\theta \log \mathbb{P}_\theta(\xi)$ are Lipschitz in θ. By the fact that the sum of Lipschitz functions is Lipschitz, we know that $\widehat{\nabla}_\theta L(\nu, \omega, \theta)$ is Lipschitz in θ. □

Lemma 11. $\{\delta_{\omega_{k+1}}\}$ is a martingale difference sequence.

Proof. Since the trajectories are generated based on the sampling p.m.f and all these trajectories are independent, we have $\mathbb{E}[\delta_{\theta_{k+1}}|\mathcal{F}_{\nu,k}] = 0$ where $\mathcal{F}_{\theta,k} = \sigma(\theta_m, \delta_{\theta_m}, m \leq k)$ is the filtration of θ_k generated by different independent trajectories.

It remains to show $\mathbb{E}[||\delta_{\theta_{k+1}}||^2|\mathcal{F}_{\nu,k}]$ is bounded. Consider

$$\begin{aligned}\delta_{\theta_{k+1}} &= -\nu^*(\theta)\sum_{\xi}\mathbb{P}_\theta(\xi)\nabla_\theta\log\mathbb{P}_\theta(\xi)\cdot\phi^*\left(\frac{J(\xi)}{\nu^*(\theta)}-\omega^*(\theta_k)+\beta\right)\\ &\quad+\nu^*(\theta)\sum_{\xi}\frac{1}{N}\nabla_\theta\log\mathbb{P}_\theta(\xi)\cdot\phi^*\left(\frac{J(\xi)}{\nu^*(\theta)}-\omega^*(\theta_k)+\beta\right)\\ &\leq V_{\max}(K_1(\xi)+K_2(\xi))\max\left\{|U_{\min}|,|U_{\max}|\right\}.\end{aligned}$$

Thus, $||\delta_{\theta_{k+1}}||^2 \leq (c_8)^2$, where $c_8 = (V_{\max}K_1(\xi)+K_2(\xi))\max\left\{|U_{\min}|,|U_{\max}|\right\} \in \mathbb{R}$, which further implies that $\{\delta_{\theta_{k+1}}\}$ is a martingale difference sequence. □

Appendix C

Technical Results in Chapter 4

C.1 Proof of Lemma 2

Note that $\sum_{x' \in \mathcal{X}} \xi(x') P(x'|x,a) = 1$ and $\xi(x') \geq 0$, then by Definition 8, we have P1 and P2. For P3, applying the definition of sup norm, we have

$$-||V_1 - V_2||_\infty \leq V_1(x,y) - V_2(x,y) \leq ||V_1 - V_2||_\infty.$$

Applying P1 and P2, we have

$$-\gamma||V_1 - V_2||_\infty \leq \mathbf{T}[V_1](x,y) - \mathbf{T}[V_2](x,y) \leq \gamma||V_1 - V_2||_\infty,$$

which implies

$$|\mathbf{T}[V_1](x,y) - \mathbf{T}[V_2](x,y)| \leq \gamma||V_1 - V_2||_\infty.$$

Now, we prove P4. Suppose $yV(x,y)$ is concave in $y \in \mathcal{Y}$. Let y_1, y_2 be two elements in $\mathcal{Y}$ and define $y_\lambda = (1-\lambda)y_1 + \lambda y_2$, where $\lambda \in [0,1]$. By Definition 8, for every $x \in \mathcal{X}$, we have

$$
\begin{aligned}
&(1-\lambda)y_1\mathbf{T}[V](x,y_1) + \lambda y_2 \mathbf{T}[V](x,y_2) \\
&= (1-\lambda)y_1 \min_{a_1\in\mathcal{A}} \left[C(x,a_1) + \gamma \max_{\xi_1 \in \mathcal{U}_{\text{NCVaR}}(y_1,\vec{\kappa}(x,a_1),P(\cdot|x,a_1))} \sum_{x'\in\mathcal{X}} \xi_1(x') V(x', y_1\xi(x')) P(x'|x,a_1) \right] \\
&\quad + \lambda y_2 \min_{a_2\in\mathcal{A}} \left[C(x,a_2) + \gamma \max_{\xi_2 \in \mathcal{U}_{\text{NCVaR}}(y_2,\vec{\kappa}(x,a_2),P(\cdot|x,a_2))} \sum_{x'\in\mathcal{X}} \xi_2(x') V(x', y_2\xi(x')) P(x'|x,a_2) \right] \\
&\le \min_{a\in\mathcal{A}} \Bigg[y_\lambda C(x,a) + \gamma \max_{\substack{\xi_1 \in \mathcal{U}_{\text{NCVaR}}(y_1,\vec{\kappa}(x,a),P(\cdot|x,a)) \\ \xi_2 \in \mathcal{U}_{\text{NCVaR}}(y_2,\vec{\kappa}(x,a),P(\cdot|x,a))}} \sum_{x'\in\mathcal{X}} P(x'|x,a) \big((1-\lambda)y_1\xi_1(x')V(x', y_1\xi_1(x')) \\
&\qquad\qquad + \lambda y_2 \xi_2(x') V(x', y_2\xi_2(x')) \big) \Bigg] \\
&\le \min_{a\in\mathcal{A}} \Bigg[y_\lambda C(x,a) + \gamma \max_{\substack{\xi_1 \in \mathcal{U}_{\text{NCVaR}}(y_1,\vec{\kappa}(x,a),P(\cdot|x,a)) \\ \xi_2 \in \mathcal{U}_{\text{NCVaR}}(y_2,\vec{\kappa}(x,a),P(\cdot|x,a))}} \sum_{x'\in\mathcal{X}} P(x'|x,a) \big((1-\lambda)y_1\xi_1(x') + \lambda y_2\xi_2(x') \big) \\
&\qquad\qquad \cdot V\big(x, (1-\lambda)y_1\xi_1(x') + \lambda y_2\xi_2(x')\big) \Bigg].
\end{aligned}
\tag{C.1}
$$

The first inequality is by the concavity of min and the second inequality is due to concavity of $yV(x,y)$. Now, we need to show

$$
\xi_\lambda = \frac{(1-\lambda)y_1\xi_1 + \lambda y_2\xi_2}{y_\lambda} \in \mathcal{U}_{\text{NCVaR}}(y_\lambda, \vec{\kappa}(x,a), P(\cdot|x,a)),
$$

where $\xi_1 \in \mathcal{U}_{\text{NCVaR}}(y_1, \vec{\kappa}(x,a), P(\cdot|x,a))$ and $\xi_2 \in \mathcal{U}_{\text{NCVaR}}(y_2, \vec{\kappa}(x,a), P(\cdot|x,a))$. By the definition of ξ_1, ξ_2 and ξ_λ , we obtain

$$
\textstyle\sum_{x'\in\mathcal{X}} \xi_\lambda(x') P(x'|x,a) = 1,
$$

and

$$
\xi_\lambda = \frac{(1-\lambda)y_1\xi_1 + \lambda y_2\xi_2}{y_\lambda} \le \frac{\vec{\kappa}(x,a)}{y_\lambda}.
$$

Thus, ξ_λ is in the set $\mathcal{U}_{\text{NCVaR}}(y_\lambda, \vec{\kappa}(x,a), P(\cdot|x,a))$. Applying this result to (C.1), we obtain

$$\begin{aligned}
&(1-\lambda)y_1\mathbf{T}[V](x,y_1) + \lambda y_2 \mathbf{T}[V](x,y_2) \\
&\leq \min_{a\in\mathcal{A}} \left[y_\lambda C(x,a) + \gamma \max_{\xi \in \mathcal{U}_{\text{NCVaR}}(y_\lambda, \vec{\kappa}(x,a), P(\cdot|x,a))} \sum_{x'\in\mathcal{X}} P(x'|x,a) y_\lambda \xi(x') V(x, y_\lambda \xi(x')) \right] \\
&= y_\lambda \mathbf{T}[V](x, y_\lambda).
\end{aligned}$$

Thus, we show $y\mathbf{T}[V](x,y)$ is concave in y given $yV(x,y)$ is concave.

It remains to show that the maximization problem in (4.4) is concave. Here, we consider the function

$$G_{x,y,a}(z) := \begin{cases} zV(x',z)P(x'|x,a)/y & \text{if} \quad y \neq 0 \\ 0 & \text{otherwise.} \end{cases}$$

Suppose $zV(x,z)$ is concave in z. When $y=0$, the function is concave in z. Otherwise, $G_{x,y,a}(z)$ is concave in z due to the fact that $P(x'|x,a) \geq 0$. This result further implies

$$\sum_{x'\in\mathcal{X}} \xi(x') V(x', y\xi(x')) P(x'|x,a) = \sum_{x'\in\mathcal{X}} G_{x,y,a}(y\xi(x'))$$

is concave in ξ. Combining all these results, P4 holds.

C.2 Proof of Theorem 9

The proof of Theorem 9 follows the idea in the proof of Theorem 4 in [24].

For convenience, denote the total discounted cost from time 0 up to time T as $\mathcal{C}_{0,T} = \sum_{t=0}^{T} \gamma^t C(x_t, a_t)$ and the initial value as $V_0(x,y)$. In the first place, we want to show the following equation holds for any $(x,y) \in \mathcal{X}\times\mathcal{Y}$ by induction hypothesis.

$$V_n(x,y) := \mathbf{T}^n[V_0](x,y) = \min_{\pi\in\Pi_M} \text{NCVaR}_{y,\vec{\kappa}} \left(\mathcal{C}_{0,n} + \gamma^n V_0(x_n, y_n) | x_0 = x, \pi \right), \tag{C.2}$$

where $x_0 = x$, $y_0 = y$ and $a_t = \pi(x_t, y_t)$.

Here, we first verify equation (C.2) hold when $n = 1$. Let (x_1, y_1) denote $(x', y\xi(x'))$, then by definition we have

$$V_1(x, y) = \mathbf{T}[V_0](x, y) = \min_{\pi \in \Pi_H} \left[C(x_0, a_0) + \gamma \text{NCVaR}_{y,\vec{\kappa}} \left(C(x_1, a_1) + V_0(x_1, y_1) | x_0 = x, \pi \right) \right].$$

When $n = 1$, a_1 only depends on x_1 and y_1, therefore, π is a Markovian policy, i.e., $\pi \in \Pi_M$. Hence, by moving constant terms inside, we have

$$V_1(x, y) = \min_{\pi \in \Pi_M} \text{NCVaR}_{y,\vec{\kappa}} \left(\mathcal{C}_{0,1} + \gamma V_0(x_1, y_1) | x_0 = x, \pi \right).$$

Secondly, assume equation (C.2) hold when $n = k$, then for $n = k+1$ with $x_0 = x$ and $y_0 = y$, we obtain

$$\begin{aligned}
&V_{k+1}(x, y) := \mathbf{T}^{k+1}[V_0](x, y) = \mathbf{T}[V_k](x, y) \\
&= \min_{a \in \mathcal{A}} \left[C(x, a) + \gamma \max_{\xi \in \mathcal{U}_{\text{NCVaR}}(y, \vec{\kappa}(x,a), P(\cdot|x,a))} \sum_{x' \in \mathcal{X}} \xi(x') V_k(x', y\xi(x')) P(x'|x, a) \right] \\
&= \min_{a \in \mathcal{A}} \left[C(x, a) + \gamma \max_{\xi \in \mathcal{U}_{\text{NCVaR}}(y, \vec{\kappa}(x,a), P(\cdot|x,a))} \sum_{x' \in \mathcal{X}} \xi(x') P(x'|x, a) \min_{\pi \in \Pi_M} \text{NCVaR}_{y_1,\vec{\kappa}}(\mathcal{C}_{0,k} + \gamma^k V_0 | x_0 = x_1, \pi) \right] \\
&= \min_{a \in \mathcal{A}} \left[C(x, a) + \max_{\xi \in \mathcal{U}_{\text{NCVaR}}(y, \vec{\kappa}(x,a), P(\cdot|x,a))} \mathbb{E}_{\xi P} \left[\min_{\pi \in \Pi_M} \text{NCVaR}_{y_1,\vec{\kappa}}(\mathcal{C}_{1,k+1} + \gamma^{k+1} V_0 | x_1, \pi) \right] \right] \\
&= \min_{a \in \mathcal{A}} \left[\min_{\pi \in \Pi_M} \text{NCVaR}_{y,\vec{\kappa}} \left(\mathcal{C}_{0,k+1} + \gamma^{k+1} V_0 | x_0 = x, \pi \right) \right] \\
&= \min_{\pi \in \Pi_M} \text{NCVaR}_{y,\vec{\kappa}} \left(\mathcal{C}_{0,k+1} + \gamma^{k+1} V_0 | x_0 = x, \pi \right).
\end{aligned}$$

Thus, (C.2) is proved by induction.

In the second place, we need to show

$$V^*(x_0, y_0) = \min_{\pi \in \Pi_M} \text{NCVaR}_{y_0,\vec{\kappa}} \left(\lim_{n \to \infty} \mathcal{C}_{0,n} | x_0, \pi \right).$$

By the definition of V^*, Proposition 2.2 in [15] and definition of V_n, for any $(x, y) \in \mathcal{X} \times \mathcal{Y}$, we

have

$$V^*(x,y) = \mathbf{T}[V^*](x,y) = \lim_{n\to\infty} \mathbf{T}^n[V_0](x,y) = \lim_{n\to\infty} V_n(x,y).$$

Combining the above results, we have

$$V^*(x_0,y_0) = \lim_{n\to\infty} V_n(x_0,y_0) = \min_{\pi\in\Pi_M} \text{NCVaR}_{y_0,\vec{\kappa}} \left(\lim_{n\to\infty} (\mathcal{C}_{0,n} + \gamma^n V_0(x_n,y_n)) | x_0, \pi \right)$$

since the state-wise cost is bounded and V_0 is also bounded. By applying the coherent property of NCVaR, we obtain

$$\begin{aligned} V^*(x_0,y_0) &\leq \min_{\pi\in\Pi_M} \left[\text{NCVaR}_{y_0,\vec{\kappa}} \left(\lim_{n\to\infty} \mathcal{C}_{0,n} | x_0, \pi \right) + \lim_{n\to\infty} \gamma^n \parallel V_0 \parallel_\infty \right] \\ &\leq \min_{\pi\in\Pi_M} \text{NCVaR}_{y_0,\vec{\kappa}} \left(\lim_{n\to\infty} \mathcal{C}_{0,n} | x_0, \pi \right) + \left| \lim_{n\to\infty} \gamma^n \parallel V_0 \parallel_\infty \right| \end{aligned}$$

which implies

$$\begin{aligned} -\lim_{n\to\infty} \gamma^n \parallel V_0 \parallel_\infty \leq V^*(x_0,y_0) - \min_{\pi\in\Pi_M} \text{NCVaR}_{y_0,\vec{\kappa}} \left(\lim_{n\to\infty} \mathcal{C}_{0,n} | x_0, \pi \right) \\ \leq \lim_{n\to\infty} \gamma^n \parallel V_0 \parallel_\infty . \end{aligned}$$

Since $\gamma \in (0,1]$, the term $\lim_{n\to\infty} \gamma^n \parallel V_0 \parallel_\infty \to 0$ as $n \to \infty$. Thus,

$$V^*(x_0,y_0) = \min_{\pi\in\Pi_M} \text{NCVaR}_{y_0,\vec{\kappa}} \left(\lim_{n\to\infty} \mathcal{C}_{0,n} | x_0, \pi \right)$$

holds for any $(x_0,y_0) \in \mathcal{X} \times \mathcal{Y}$.

Lastly, we want to get the optimal value over all history-dependent policies, which means given (x_0,y_0), we have

$$V^*(x_0,y_0) = \min_{\pi\in\Pi_H} \text{NCVaR}_{y_0,\vec{\kappa}} \left(\lim_{n\to\infty} \mathcal{C}_{0,n} | x_0, \pi \right).$$

For each $(x_t, y_t) \in \mathcal{X} \times \mathcal{Y}$ with $t \geq 0$, the t^{th} tail-subproblem is defined as follow:

$$\mathbb{V}(x_t, y_t) = \min_{\pi \in \Pi_H} \text{NCVaR}_{y_t, \vec{\kappa}} \left(\lim_{n \to \infty} \mathcal{C}_{t,n} | x_t, \pi \right)$$

where the tail policy sequence is $\pi = \{\pi_t, \pi_{t+1}, \dots\}$.

For any history depend policy $\tilde{\pi} \in \Pi_H$, we also define the $\tilde{\pi}$-induced value function as $\text{NCVaR}_{y_t, \vec{\kappa}}(\lim_{n \to \infty} \mathcal{C}_{t,n} | x_t, \tilde{\pi})$ where $\tilde{\pi} = \{\tilde{\pi}_t, \tilde{\pi}_{t+1}, \dots\}$ and $a_j = \tilde{\pi}_j(h_j)$ for $j \geq t$. Let π^* denote the optimal policy of the t^{th}-subproblem, then for any state x_{t+1} and confidence level y_{t+1}, the policy $\tilde{\pi} = \{\pi^*_{t+1}, \pi^*_{t+2}, \dots\}$ is a feasible policy for the $(t+1)^{th}$-subproblem:

$$\min_{\pi \in \Pi_H} \text{NCVaR}_{y_{t+1}, \vec{\kappa}} \left(\lim_{n \to \infty} \mathcal{C}_{t+1,n} | x_{t+1}, \pi \right).$$

Based on all these results, for any $(x_t, y_t) \in \mathcal{X} \times \mathcal{Y}$ with $a_t = \pi^*_t(x_t)$, we have

$$\begin{aligned}
\mathbb{V}(x_t, y_t) &= \text{NCVaR}_{y_t, \vec{\kappa}} \left(\lim_{n \to \infty} \mathcal{C}_{t,n} | x_t, \pi^* \right) \\
&= C(x_t, a_t) + \gamma \text{NCVaR}_{y_t, \vec{\kappa}} \left(\lim_{n \to \infty} \mathcal{C}_{t+1,n} | x_{t+1}, \tilde{\pi} \right) \\
&= C(x_t, a_t) + \gamma \max_{\xi \in \mathcal{U}_{\text{NCVaR}}(y_t, \vec{\kappa}(x,a), P(\cdot|x_t, a_t))} \mathbb{E} \left[\xi(x_{t+1}) \cdot \text{NCVaR}_{y_{t+1}, \vec{\kappa}} \left(\lim_{n \to \infty} \mathcal{C}_{t+1,n} | x_{t+1}, \tilde{\pi} \right) \right] \\
&= C(x_t, a_t) + \gamma \max_{\xi \in \mathcal{U}_{\text{NCVaR}}(y_t, \vec{\kappa}, P(\cdot|x_t, a_t))} \mathbb{E}_\xi \left[\mathbb{V}^{\tilde{\pi}}(x_{t+1}, y_t \xi(x_t + 1)) | x_t, y_t, a_t \right] \\
&\geq C(x_t, a_t) + \gamma \max_{\xi \in \mathcal{U}_{\text{NCVaR}}(y_t, \vec{\kappa}, P(\cdot|x_t, a_t))} \mathbb{E}_\xi \left[\mathbb{V}(x_{t+1}, y_t \xi(x_t + 1)) | x_t, y_t, a_t \right] \\
&\geq \mathbf{T}[\mathbb{V}](x_t, y_t).
\end{aligned}$$

For the equality, the third one is by the decomposition theorem and the forth one is by defining $\mathbb{V}^{\tilde{\pi}}(x_t, y_t) = \text{NCVaR}_{y_t, \vec{\kappa}}(\lim_{n \to \infty} \mathcal{C}_{t,n} | x_t, \tilde{\pi})$. Moreover, the first inequality is by $\mathbb{V}^{\tilde{\pi}}(x, y) \geq \mathbb{V}(x, y)$ for any $(x, y) \in \mathcal{X} \times \mathcal{Y}$ and the last inequality is by the definition of $\mathbf{T}$.

For any state x_{t+1} and confidence level y_{t+1}, let $\pi^* = \{\pi^*_{t+1}, \pi^*_{t+2}, \dots\} \in \Pi_H$ be an optimal policy for the $(t+1)^{th}$ tail subproblem. Then we can construct policy $\tilde{\pi} = \{\tilde{\pi}_t, \tilde{\pi}_{t+1}, \dots\} \in \Pi_H$ for the t^{th} subproblem from π^* when given $(x_t, y_t) \in \mathcal{X} \times \mathcal{Y}$ is given. For the policy $\tilde{\pi}$, we choose

$\tilde{\pi}_t(x_t) = u^*(x_t, y_t)$ and $\tilde{\pi}_j(h_j) = \pi_j^*(h_j)$, where

$$u^*(x_t, y_t) \in \arg\min_{a \in \mathcal{A}} \left[C(x_t, a) + \gamma \max_{\xi \in \mathcal{U}_{\text{NCVaR}}(y_t, \vec{\kappa}, P(\cdot|x_t, a))} \mathbb{E}_\xi \left[\mathbb{V}(x_{t+1}, y_t \xi x_{t+1}) | x_t, y_t, a \right] \right],$$

with the given confidence level y_t of the t^{th} tail-subproblem and the connection between y_t and y_{t+1} is $y_{t+1} = y_t \xi^*(x_{t+1})$ with

$$\xi^* \in \arg \max_{\xi \in \mathcal{U}_{\text{NCVaR}}(y_t, \vec{\kappa}(x,a), P(\cdot|x_t, a^*))} \mathbb{E} \left[\xi(x_{t+1}) \text{NCVaR}_{y_t \xi(x_{t+1}), \vec{\kappa}} \left(\lim_{n \to \infty} \mathcal{C}_{t+1,n} | x_{t+1,n}, \tilde{\pi} \right) \right].$$

Since π^* is an optimal policy, it is a feasible policy for the tail subproblem from time $t+1$. Then the policy $\tilde{\pi} \in \Pi_H$ is a feasible policy for the tail subproblem from time t: $\min_{\pi \in \Pi_H} \text{NCVaR}_{y_t}(\lim_{n \to \infty} \mathcal{C}_{t+1,n} | x_t, \pi)$. Hence,

$$\mathbb{V}(x_t, y_t) \leq C\left(x_t, \tilde{\pi}_t(x_t)\right) + \gamma \text{NCVaR}_{y_t, \vec{\kappa}} \left(\lim_{n \to \infty} \mathcal{C}_{t+1,n} | x_t, \tilde{\pi} \right).$$

By the definition of π^*, we obtain

$$\begin{aligned}
\mathbb{V}(x_t, y_t) &\leq C(x_t, u^*(x_t, y_t)) + \gamma \max_{\xi \in \mathcal{U}_{\text{EVaR}}(y_t, \vec{\kappa}(x,a), P(\cdot|x_t, u^*(x_t, y_t)))} \\
&\quad \mathbb{E} \left[\xi(x_{t+1}) \cdot \text{NCVaR}_{y_t \xi(x_{t+1}), \vec{\kappa}} \left(\lim_{n \to \infty} \mathcal{C}_{t+1,n} | x_{t+1}, \tilde{\pi} \right) \Big| x_t, y_t, u^*(x_t, y_t) \right] \\
&\leq C(x_t, u^*(x_t, y_t)) + \gamma \max_{\xi \in \mathcal{U}_{\text{NCVaR}}(y_t, \vec{\kappa}, P(\cdot|x_t, u^*(x_t, y_t)))} \mathbb{E}_\xi \left[\mathbb{V}(x_{t+1}, y_t \xi(x_{t+1})) | x_t, y_t, u^*(x_t, y_t) \right] \\
&= \mathbf{T}[\mathbb{V}](x_t, y_t).
\end{aligned}$$

Recall the result $\mathbb{V}(x_t, y_t) \geq \mathbf{T}[\mathbb{V}](x_t, y_t)$ and $\mathbb{V}(x_t, y_t) \leq \mathbf{T}[\mathbb{V}](x_t, y_t)$ hold for all $t \geq 0$, we can show that $\mathbb{V}$ is a fixed-point solution of $\mathbb{V}(x_t, y_t) = \mathbf{T}[\mathbb{V}](x_t, y_t)$ for any $(x, y) \in \mathcal{X} \times \mathcal{Y}$. Due to the fact that the fixed-point solution is unique, we have $V^*(x, y) = \mathbb{V}(x, y)$ for any $(x, y) \in \mathcal{X} \times \mathcal{Y}$, i.e.,

$$V^*(x, y) = \mathbb{V}(x, y) = \min_{\pi \in \Pi_H} \text{NCVaR}_{y, \vec{\kappa}} \left(\lim_{n \to \infty} \mathcal{C}_{0,n} | x_0 = x, \pi \right).$$

The proof is complete by combining all those results.

Appendix D

Technical Results in Chapter 5

D.1 Proof of Exploration Phase

D.1.1 Proof of Lemma 3

Recall the definition of value function V for various policy types:

$$\pi \in \Pi_H : V_h^{\pi}(s_h, b_h; H_h) = \mathbb{E}_{\pi}\left[\left(b_h - \sum_{t=h}^{H} r_t\right)^{+} \middle| s_h, b_h, H_h\right],$$

$$\rho \in \Pi^{\text{Aug}} : V_h^{\rho}(s_h, b_h) = \mathbb{E}^{\rho}\left[\left(b_h - \sum_{t=h}^{H} r_t\right)^{+} \middle| s_h, b_h\right].$$

Notice that executing ρ, b in the augmented MDP is equivalent to executing policy $\pi^{\rho,b}$ in the original MDP, where $\pi_h^{\rho,b}(s_h, H_h) = \rho_h(s_h, b - r_1 - \ldots - r_{h-1})$. Consequently, their V functions should be equivalent.

Therefore, by Lemma 15, we have

$$
\begin{aligned}
&\mathrm{CVaR}_\alpha^\star(s_1;r) - \mathrm{CVaR}_\alpha^{\hat{\rho}_r}(s_1;r) \\
&= \mathrm{CVaR}_\alpha^{\pi^{\rho^\star,b_1^\star}}(s_1;r) - \mathrm{CVaR}_\alpha^{\pi^{\hat{\rho},\hat{b}_1}}(s_1;r) \\
&= \underbrace{\mathrm{CVaR}_\alpha^{\pi^{\rho^\star,b_1^\star}}(s_1;r) - \widehat{\mathrm{CVaR}}_\alpha^{\pi^{\rho^\star,b_1^\star}}(s_1;r)}_{\text{Evaluation error I}} + \underbrace{\widehat{\mathrm{CVaR}}_\alpha^{\pi^{\rho^\star,b_1^\star}}(s_1;r) - \widehat{\mathrm{CVaR}}_\alpha^{\pi^{\hat{\rho}^\star,\hat{b}_1^\star}}(s_1;r)}_{\leq 0 \text{ by definition}} \\
&\quad + \underbrace{\widehat{\mathrm{CVaR}}_\alpha^{\pi^{\hat{\rho}^\star,\hat{b}_1^\star}}(s_1;r) - \widehat{\mathrm{CVaR}}_\alpha^{\pi^{\hat{\rho},\hat{b}_1}}(s_1;r)}_{\text{optimization error } \leq \epsilon/3 \text{ by Assumption 8}} + \underbrace{\widehat{\mathrm{CVaR}}_\alpha^{\pi^{\hat{\rho},\hat{b}_1}}(s_1;r) - \mathrm{CVaR}_\alpha^{\pi^{\hat{\rho},\hat{b}_1}}(s_1;r)}_{\text{Evaluation error II}}.
\end{aligned}
$$

By the triangle inequality, we have

$$
\begin{aligned}
&\left|\mathrm{CVaR}_\alpha^\star(s_1;r) - \mathrm{CVaR}_\alpha^{\hat{\rho}_r^\star}(s_1;r)\right| \\
&\leq \left|\mathrm{CVaR}_\alpha^{\pi^{\rho^\star,b^\star}}(s_1;r) - \widehat{\mathrm{CVaR}}_\alpha^{\pi^{\rho^\star,b^\star}}(s_1;r)\right| + \left|\widehat{\mathrm{CVaR}}_\alpha^{\pi^{\hat{\rho}^\star,\hat{b}^\star}}(s_1;r) - \mathrm{CVaR}_\alpha^{\pi^{\hat{\rho}^\star,\hat{b}^\star}}(s_1;r)\right|.
\end{aligned}
$$

For the evaluation errors, by the definition of CVaR, we have

$$
\begin{aligned}
&\left|\mathrm{CVaR}_\alpha^{\pi^{\rho^\star,b_1^\star}}(s_1;r) - \widehat{\mathrm{CVaR}}_\alpha^{\pi^{\rho^\star,b_1^\star}}(s_1;r)\right| \\
&= \left|b_1^\star - \alpha^{-1} V_1^{\pi^{\rho^\star,b_1^\star}}(s_1,b_1^\star;r) - \max_{b_1\in[0,H]}\left\{b_1 - \alpha^{-1}\hat{V}_1^{\pi^{\rho^\star,b_1^\star}}(s_1,b_1;r)\right\}\right| \\
&\leq \left|b_1^\star - \alpha^{-1} V_1^{\pi^{\rho^\star,b_1^\star}}(s_1,b_1^\star;r) - \left(b_1^\star - \alpha^{-1}\hat{V}_1^{\pi^{\rho^\star,b_1^\star}}(s_1,b_1^\star;r)\right)\right| \\
&\leq \alpha^{-1}\left|V_1^{\pi^{\rho^\star,b_1^\star}}(s_1,b_1^\star;r) - \hat{V}_1^{\pi^{\rho^\star,b_1^\star}}(s_1,b_1^\star;r)\right|,
\end{aligned}
$$

and similarly,

$$
\left|\widehat{\mathrm{CVaR}}_\alpha^{\pi^{\hat{\rho},\hat{b}_1}}(s_1;r) - \mathrm{CVaR}_\alpha^{\pi^{\hat{\rho},\hat{b}_1}}(s_1;r)\right| \leq \alpha^{-1}\left|V_1^{\pi^{\hat{\rho},\hat{b}_1}}(s_1,\hat{b}_1;r) - \hat{V}_1^{\pi^{\hat{\rho},\hat{b}_1}}(s_1,\hat{b}_1;r)\right|.
$$

Therefore, if an exploration algorithm that satisfies

$$
\left|V_1^\rho(s_1,b_1;r) - \hat{V}_1^\rho(s_1,b_1;r)\right| \leq \epsilon\alpha/3, \forall \rho \in \Pi^{\mathrm{Aug}}, \forall b_1 \in [0,H],
$$

or equivalently,

$$\left|Q_1^{\rho}(s_1, b_1, \rho(s_1, b_1); r) - \hat{Q}_1^{\rho}(s_1, b_1, \rho(s_1, b_1); r)\right| \leq \epsilon\alpha/3, \forall \rho \in \Pi^{\text{Aug}}, \forall b_1 \in [0, H],$$

it further ensures $\left|\text{CVaR}_{\alpha}^{\star}(s_1; r) - \text{CVaR}_{\alpha}^{\hat{\rho}_r^{\star}}(s_1; r)\right| \leq \epsilon$, which completes the proof.

D.1.2 Proof of Lemma 4

We first consider the case where the initial budget b_1 is fixed and for convenience, we omit the index $h+1$ by using (s', b'). Referring to the Bellman equations in both the empirical augmented MDP and the true augmented MDP,

$$\hat{Q}_h^{t,\rho}(s_h, b_h, a_h; r) = \sum_{s'} \hat{P}_h^t(s'|s, a)\hat{Q}_{h+1}^{t,\rho}(s', b', \rho(s', b'); r),$$

$$\text{and } Q_h^{\rho}(s_h, b_h, a_h; r) = \sum_{s'} P_h(s'|s, a)Q_{h+1}^{\rho}(s', b', \rho(s', b'); r),$$

we have

$$\begin{aligned}
&\hat{Q}_h^{t,\rho}(s_h, b_h, a_h; r) - Q_h^{\rho}(s_h, b_h, a_h; r) \\
&= \sum_{s'} \hat{P}_h^t(s'|s, a)\hat{Q}_{h+1}^{t,\rho}(s', b', \rho(s', b'); r) - \sum_{s'} P_h(s'|s, a)Q_{h+1}^{\rho}(s', b', \rho(s', b'); r) \\
&= \sum_{s'} \left(\hat{P}_h^t(s'|s, a) - P_h(s'|s, a)\right) Q_{h+1}^{\rho}(s', b', \rho(s', b'); r) \\
&\quad + \sum_{s'} \hat{P}_h^t(s'|s, a) \left(\hat{Q}_{h+1}^{t,\rho}(s', b', \rho(s', b'); r) - Q_{h+1}^{\rho}(s', b', \rho(s', b'); r)\right).
\end{aligned}$$

Thus, for $n_h^t(s,a) \geq 0$, we obtain

$$\begin{aligned}
\hat{e}_h^{t,\rho}(s_h, b_h, a_h; r) &= |\hat{Q}_h^{t,\rho}(s_h, b_h, a_h; r) - Q_h^{\rho}(s_h, b_h, a_h; r)| \\
&\overset{(1)}{\leq} \sum_{s'} \left| \hat{P}_h^t(s'|s,a) - P_h(s'|s,a) \right| Q_{h+1}^{\rho}(s', b', \rho(s', b'); r) \\
&\quad + \sum_{s'} \hat{P}_h^t(s'|s,a) \left| \hat{Q}_{h+1}^{t,\rho}(s', b', \rho(s', b'); r) - Q_{h+1}^{\rho}(s', b', \rho(s', b'); r) \right| \\
&\overset{(2)}{\leq} b_1 \|\hat{P}_h^t(\cdot|s,a) - P_h(\cdot|s,a)\|_1 + \sum_{s'} \hat{P}_h^t(s'|s,a) \hat{e}_{h+1}^{t,\rho}(s', b', a'; r) \\
&\overset{(3)}{\leq} b_1 \sqrt{\frac{2\beta(n_h^t(s,a), \delta)}{n_h^t(s,a)}} + \sum_{s'} \hat{P}_h^t(s'|s,a) \hat{e}_{h+1}^{t,\rho}(s', b', a'; r),
\end{aligned}$$

where (1) is due to the Pinsker's inequality; (2) is due to the fact that $Q_h^{\rho}(s_h, b_h, a_h; r) \leq b_1$ ($Q_h^{\rho}(s_h, b_h, a_h; r) \leq (b_h)^+ \leq b_1$ as $b_{h+1} = b_h - r_h$) for all s, a, b, r and the definition of L_1 norm; (3) is due to the fact that $\text{TV}(P, Q) = \frac{1}{2}\|P(\cdot) - Q(\cdot)\|_1 \leq \sqrt{\frac{1}{2}\text{KL}(P, Q)}$ and the definition of $\mathcal{E}$.

Notice that $\hat{e}_h^{t,\rho}(s_h, b_h, a_h; r) \leq b_1$, then for all $n_h^t(s,a) \geq 0$, we have

$$\hat{e}_h^{t,\rho}(s_h, a_h, b_h; r) \leq \min \left\{ b_1, b_1 \sqrt{\frac{2\beta(n_h^t(s,a), \delta)}{n_h^t(s,a)}} + \sum_{s'} \hat{P}_h^t(s'|s,a) \hat{e}_{h+1}^{t,\rho}(s', a', b'; r) \right\}.$$

Notice that $b_1 \in [0, H]$, in order to find the upper bound of the estimation error over all the initial budgets, we extend the inequality to

$$\begin{aligned}
\hat{e}_h^{t,\rho}(s_h, a_h, b_h; r) &\leq \max_{b_1 \in [0,H]} \left\{ \min \left\{ b_1, b_1 \sqrt{\frac{2\beta(n_h^t(s,a), \delta)}{n_h^t(s,a)}} + \sum_{s'} \hat{P}_h^t(s'|s,a) \hat{e}_{h+1}^{t,\rho}(s', a', b'; r) \right\} \right\} \\
&\leq \min \left\{ H, H \sqrt{\frac{2\beta(n_h^t(s,a), \delta)}{n_h^t(s,a)}} + \sum_{s'} \hat{P}_h^t(s'|s,a) \hat{e}_{h+1}^{t,\rho}(s', a', b'; r) \right\}.
\end{aligned}$$

Now we prove Lemma 4 by induction. For $H+1$, since

$$\hat{e}_{H+1}^{t,\rho}(s, a, b; r) = |\hat{Q}_{H+1}^{t,\rho}(s, a, b; r) - Q_{H+1}^{\rho}(s, a, b; r)| = \max\{0, b_1\} - \max\{0, b_1\} = 0$$

and $E^t_{H+1}(s,a) = 0$ for all (s,a), the result is true. Assume the result holds for $h+1$, i.e., $\hat{e}^{t,\rho}_{h+1}(s,a,b;r) \leq E^t_{h+1}(s,a;b_1)$ for all (s,a), we have

$$\begin{aligned}\hat{e}^{t,\rho}_h(s,a,b;r) &\leq \min\left\{H, H\sqrt{\frac{2\beta(n^t_h(s,a),\delta)}{n^t_h(s,a)}} + \sum_{s'} \hat{P}^t_h(s'|s,a)\hat{e}^{t,\rho}_{h+1}(s',a',b';r)\right\} \\ &\leq \min\left\{H, H\sqrt{\frac{2\beta(n^t_h(s,a),\delta)}{n^t_h(s,a)}} + \sum_{s'} \hat{P}^t_h(s'|s,a)\max_{a\in\mathcal{A}} E_{h+1}(s',a)\right\} = E^t_h(s,a)\end{aligned}$$

holds for h, which complete the proof.

D.1.3 Proof of Theorem 11

Notice that in the exploration phase, we follow the exploration policy π rather than ρ. We begin by introducing some notations. Let $P^\pi_h(s,a)$ represent the probability that the state-action pair (s,a) is reached at the h-th step of a trajectory under the exploraion policy π. We use the shorthand $p^h_t(s,a) = p^h_{\pi_t}(s,a)$ for simplicity. The pseudo-counts $\bar{n}^t_h(s,a)$ are defined as $\sum_{i=1}^t P^i_h(s,a)$, and we define the event

$$\mathcal{E}^{\text{cnt}} = \left\{\forall t \in \mathbb{N}^\star, \forall h \in [H], \forall (s,a) \in \mathcal{S}\times\mathcal{A} : n^t_h(s,a) \geq \frac{1}{2}\bar{n}^t_h(s,a) - \beta^{\text{cnt}}(\delta)\right\},$$

where $\beta_{\text{cnt}}(\delta) = \log(2SAH/\delta)$. Recalling the event $\mathcal{E}$ defined in Lemma 4, we let $\mathcal{F} = \mathcal{E}\cap\mathcal{E}^{\text{cnt}}$ and introduce the following lemma.

By Lemma 16 and the principle of inclusion-exclusion, we have $P(\mathcal{F}) = P(\mathcal{E}\cap\mathcal{E}^{\text{cnt}}) = P(\mathcal{E}) + P(\mathcal{E}^{\text{cnt}}) - P(\mathcal{E}\cup\mathcal{E}^{cnt}) \geq P(\mathcal{E}) + P(\mathcal{E}^{\text{cnt}}) - 1 = 1-\delta$. From Lemma 5, on the event $\mathcal{F}$, it is shown that $\text{CVaR}^\star_\alpha(s_1;r) - \text{CVaR}^{\hat{\rho}^*}_\alpha(s_1;r) \leq \epsilon$ for all reward functions r, thereby proving that CVaR-RF-UCRL is (ε,δ)-PAC.

We now proceed to upper bound the sample complexity of CVaR-RF-UCRL on the event $\mathcal{F}$. The first step involves introducing an average upper bound on the error at step h under policy π^{t+1},

defined as

$$\mathbb{Q}_h^t = \sum_{(s,a)} P_h^{t+1}(s,a) E_h^t(s,a).$$

By Lemma 13, the average errors can be related as follows:

$$\begin{aligned}
\mathbb{Q}_t^h &\leq 3H \sum_{(s,a)} P_h^{t+1}(s,a) \left[\sqrt{\frac{\beta(n_h^t(s,a),\delta)}{n_h^t(s,a)}} \wedge 1 \right] \\
&\quad + \sum_{(s,a)} \sum_{(s',a')} P_h^{t+1}(s,a) P_h(s'|s,a) \mathbb{I}(a' = \pi^{t+1}(s')) E_{h+1}^t(s',a') \\
&\leq 3H \sum_{(s,a)} P_h^{t+1}(s,a) \left[\sqrt{\frac{\beta(n_h^t(s,a),\delta)}{n_h^t(s,a)}} \wedge 1 \right] + \mathbb{Q}_{h+1}^t.
\end{aligned}$$

For $h = 1$, observe that $P_1^{t+1}(s_1,a) E_1^t(s_1,a) = E_1^t(s_1, \pi_1^{t+1}(s_1)) \mathbb{I}(\pi_1^{t+1}(s_1) = a)$, as the policy is deterministic. Now, if $t < t_{\text{stop}}$, $E_1^t(s_1, \pi_1^{t+1}(s_1)) \geq \epsilon/3$ by definition of the stopping rule, hence $\mathbb{Q}_t^1 = \sum_a P_1^{t+1}(s_1,a) E_1^t(s_1,a) \geq (\epsilon\alpha/3) \sum_{a \in \mathcal{A}} \mathbb{I}(\pi_1^{t+1}(s_1) = a) = \epsilon\alpha/3$. Thus, we have

$$\frac{\epsilon\alpha}{3} \leq 3 \sum_{h=1}^{H} \sum_{(s,a)} H P_h^{t+1}(s,a) \left[\sqrt{\frac{\beta(n_h^t(s,a),\delta)}{n_h^t(s,a)}} \wedge 1 \right]$$

for $t < t_{\text{stop}}$. Summing these inequalities for $t \in \{0, \ldots, T\}$ where $T < t_{\text{stop}}$ gives:

$$(T+1)\epsilon\alpha \leq 9 \sum_{h=1}^{H} H \sum_{(s,a)} \sum_{t=0}^{T} P_h^{t+1}(s,a) \left[\sqrt{\frac{\beta(n_h^t(s,a),\delta)}{n_h^t(s,a)}} \wedge 1 \right].$$

The next step involves relating the counts to the pseudo-counts, taking into account that the event $\mathcal{E}^{\text{cnt}}$ holds.

Using Lemma 17, it can be stated that, on the event F, for $T < t_{\text{stop}}$, the inequality

$$\begin{aligned}
(T+1)\epsilon\alpha &\leq 18 \sum_{h=1}^{H} H \sum_{(s,a)} \sum_{t=0}^{T} P_h^{t+1}(s,a) \sqrt{\frac{\beta(n_h^t(s,a),\delta)}{n_h^t(s,a) \vee 1}} \\
&\leq 18 \sqrt{\beta(T+1,\delta)} \sum_{h=1}^{H} H \sum_{(s,a)} \sum_{t=0}^{T} \frac{\bar{n}_h^{t+1}(s,a) - n_h^t(s,a)}{\sqrt{\bar{n}_h^t(s,a) \vee 1}},
\end{aligned}$$

is derived, where the relation $P_h^{t+1}(s,a) = \bar{n}_h^{t+1}(s,a) - \bar{n}_h^t(s,a)$, as per the definition of pseudo-counts, is used.

Applying Lemma 18 to bound the sum over t, we get:

$$\begin{aligned}(T+1)\epsilon\alpha &\leq 18(1+\sqrt{2})\sqrt{\beta(T+1,\delta)}\sum_{h=1}^{H} H \sum_{(s,a)} \sqrt{n_h^{T+1}(s,a)} \\ &\leq 18(1+\sqrt{2})\sqrt{\beta(T+1,\delta)}\sum_{h=1}^{H} H\sqrt{SA}\sqrt{\sum_{s,a} n_h^{T+1}(s,a)}.\end{aligned}$$

Given that $\sum_{s,a} n_h^{T+1}(s,a) = T+1$, the inequality simplifies to:

$$\sqrt{T+1}\epsilon\alpha \leq 18(1+\sqrt{2})\sqrt{SA}H^2\sqrt{\beta(T+1,\delta)}.$$

For sufficiently large T, this inequality cannot hold, as the left-hand side grows with $\sqrt{T}$, while the right-hand side is logarithmic. Therefore, t_{stop} is finite and satisfies (applying the inequality to $T = t_{\text{stop}} - 1$):

$$t_{\text{stop}} \leq \tilde{\mathcal{O}}\left(\frac{H^4S^2A}{\epsilon^2\alpha^2}\right)$$

The conclusion follows from Lemma 19.

D.2 Proof of Planning Phase

D.2.1 Proof of Theorem 12

The utilization of discretization in the algorithm significantly impacts its computational tractability, and it is applied in two main areas:

1. In the dynamic programming step at each timestep h, the algorithm exclusively computes $Q_h(s_h, b_h, a_h)$ for all s_h, a_h and b_h within the grid. This leads to a total runtime of $\mathcal{O}(SAH\eta^{-1}T_{\text{step}})$, where T_{step} represents the time required for each step. The time complexity here arises from discretization and is a function of the state space size, action space size, and the horizon length.

2. When computing $\hat{b}$, the algorithm searches over the grid to find the solution. Since the returns distribution is supported on the grid, the α-quantile of the return distribution (the optimal solution) exists on the grid. This computation has a time complexity of $\mathcal{O}(\eta^{-1})$, which is considered a lower-order term compared to the first part.

It's important to note that the most time-consuming part of the algorithm is the computation of expectations, specifically the term:

$$[P_h V_{h+1}]\left(s_h, b_h, a_h\right) = \mathbb{E}_{s_{h+1}\sim P(\cdot|s_h,a_h)}[V^{\star}_{h+1}(s_{h+1}, b_{h+1})].$$

In the discretized MDP, this expectation can be computed using only grid elements, implying $T_{\text{step}} = \mathcal{O}(S\eta^{-1})$. As a result, the overall time complexity of this algorithm is approximately $\mathcal{O}(SAH\eta^{-1}T_{\text{step}}) = \mathcal{O}(S^2AH\eta^{-2})$.

D.2.2 Proof of Theorem 13

The proof draws inspiration from [7, 99]. To facilitate the discussion, we introduce the following notation. Let $Z_{\rho,\mathcal{M}}$ represent the returns from executing ρ in the MDP $\mathcal{M}$. For random variables X, Y, we say Y stochastically dominates X, which is denoted $X \preceq Y$. This dominance implies that for any real value t, the probability that Y is less than or equal to t is greater than or equal to the probability of X being less than or equal to t, i.e., $\forall t \in \mathbb{R} : Pr(Y \leq t) \leq Pr(X \leq t)$.

1) From disc($\mathcal{M}$) to $\mathcal{M}$:

Consider any policy $\rho \in \Pi^{\text{Aug}}$ and $b \in [0, 1]$ (which we use in disc($\mathcal{M}$)). Define an adapted policy for use in $\mathcal{M}$ as follows:

$$\text{adapted}(\rho, b_1)_h(s_h, r_{1:h-1}) = \rho_h(s_h, b_1 - \phi(r_1) - \cdots - \phi(r_h - 1)).$$

The adapted policy simulates the evolution of b in disc($\mathcal{M}$) by using the history. Let $Z_{\rho,b,\text{disc}(\mathcal{M})}$ be the returns from running ρ, b in disc($\mathcal{M}$). Let $Z_{\text{adopted}(\rho,b),\mathcal{M}}$ be the returns from running

adopted(ρ, b) in $\mathcal{M}$. According to Lemma H.1 in [99], we almost surely have

$$Z_{\rho,b,\text{disc}(\mathcal{M})} - H\eta \preceq Z_{\text{adopted}(\rho,b),\mathcal{M}} \preceq Z_{\rho,b,\text{disc}(\mathcal{M})}.$$

Thus, for any $x \in \mathbb{R}$, it follows that

$$F_{\rho,b,\text{disc}(\mathcal{M})}(x) \leq F_{\text{adapted}(\rho,b),\mathcal{M}}(x) \leq F_{\rho,b,\text{disc}(\mathcal{M})}(x + H\eta)$$

where $F_{\rho,b,\text{disc}(\mathcal{M})}$ is the CDF of $Z_{\rho,b,\text{disc}(\mathcal{M})}$ and $F_{\text{adapted}(\rho,b),\text{disc}(\mathcal{M})}$ is the CDF of $Z_{\text{adapted}(\rho,b),\mathcal{M}}$.

Based on these arguments and Theorem H.3 in [99], we conclude:

$$\text{CVaR}_\alpha(\text{adapted}(\rho, b); \mathcal{M}) \geq \text{CVaR}_\alpha(\rho, b; \text{disc}(\mathcal{M})) - \alpha^{-1} H\eta. \tag{D.1}$$

2) From $\mathcal{M}$ to disc$(\mathcal{M})$: Let's introduce the memory-MDP model as defined in [99] first. The memory-MDP mode augments a standard MDP with a memory generator M_h, which produces memory items $m_h \sim M_h(s_h, a_h, r_h, H_h)$ at each timestep. These memories are stored into the history $H_h = (s_t, a_t, r_t, m_t)_{t\in[h-1]}$. The process of executing π in this memory-MDP is as follows: for any $h \in [H]$, $a_h \sim \pi_h(s_h, H_h)$, $s_{h+1} \sim P(\cdot|s_h, a_h)$, $r_h = r(s_h, a_h)$ and $m_h \sim M_h(s_h, a_h, r_h, H_h)$. As a result of this process, the augmented MDP with memory has a history $H_h^{\text{Aug}} = (s_t, b_t, a_t, r_t, m_t)_{t\in[h-1]}$. This memory-MDP model allows us to capture and model dependencies on past experiences through the memory items.

Building on the framework presented in [99], consider a scenario where we have a policy $\rho \in \Pi^{\text{Aug}}$ and an initial budget $b \in [0, 1]$, which we intend to use in the original MDP $\mathcal{M}$. To adapt this policy to run in disc$(\mathcal{M})$, we introduce a discretized policy, which is history-dependent and incorporates memory. This policy operates in the discretized MDP disc$(\mathcal{M})$ and is defined as follows:

$$\text{disc}(\rho, b)_h(s_h, m_{1:h-1}) = \rho_h(s_h, b - m_1 - \cdots - m_{h-1}).$$

Indeed, this definition of the discretized policy disc(ρ, b) is designed to ensure that, despite

receiving discrete rewards $\hat{r_h}$ in the discretized MDP disc($\mathcal{M}$), the memory element m_h is carefully generated to imitate the reward that would have been received in the true MDP $\mathcal{M}$.

By applying Lemma H.2 in [99], we almost surely have

$$Z_{\rho,b,\mathcal{M}} \preceq Z_{\text{disc}(\rho,b),\text{disc}(\mathcal{M})}.$$

Consequently, if we define $F_{\rho,b,\mathcal{M}}$ as the CDF of $Z_{\rho,b,\mathcal{M}}$ and $F_{\text{disc}(\rho,b),\text{disc}(\mathcal{M})}$ as the CDF of $Z_{\text{disc}(\rho,b),\text{disc}(\mathcal{M})}$, we can establish that,

$$\forall x \in \mathbb{R}: \quad F_{\text{disc}(\rho,b),\text{disc}(\mathcal{M})} \leq F_{\rho,b,\mathcal{M}}.$$

Based on these observations and utilizing Theorem H.4 in [99], we obtain

$$\text{CVaR}^{\star}_{\alpha}(\text{disc}(\mathcal{M})) \geq \text{CVaR}^{\star}_{\alpha}(\mathcal{M}). \tag{D.2}$$

Combining Eq. (D.1) and Eq. (D.2), we have

$$|\text{CVaR}^{\rho^\star}_{\alpha}(s_1; r) - \text{CVaR}^{\hat{\rho}}_{\alpha}(s_1; r)| \leq \alpha^{-1} H\eta. \tag{D.3}$$

We can satisfy the assumption about the optimization error by selecting $\eta \leq \epsilon\alpha/3H$ to ensure

$$|\text{CVaR}^{\rho^\star}_{\alpha}(s_1; r) - \text{CVaR}^{\hat{\rho}}_{\alpha}(s_1; r)| \leq \epsilon/3.$$

D.3 Proof of Lower Bound

In this section, we prove our lower bound presented in Theorem 14. First, we develop the connection between the reward-free problem and the CVaR-reward-free RL problem.

Lemma 12. For any MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, H, P, r)$ with initial state s_1 and any policy π, there exists

another MDP $\mathcal{M}' = (\mathcal{S}', \mathcal{A}, H+1, P', r')$ with initial state s_0, we have

$$\text{CVaR}_\alpha^{\pi,\mathcal{M}'}(s_0) = \mathbb{E}_\pi \left[\sum_{h'=1}^{H} r_{h'}(s_{h'}, a_{h'}) \middle| s_1, \mathcal{M} \right]. \tag{D.4}$$

Proof. We set horizon h starting at 0 in $\mathcal{M}'$. We can build such a $\mathcal{M}' = (\mathcal{S}', \mathcal{A}, H+1, P', r')$, where $\mathcal{S}' = \mathcal{S} \cup s_0, s'$, $P'(\cdot|s,a) = P(\cdot|s,a)$ for any $s \in \mathcal{S}$ and $a \in \mathcal{A}$, $P'(s_1|s_0,a) = \alpha$ for any $a \in \mathcal{A}$, $P'(s'|s_0,a) = 1-\alpha$ for any $a \in \mathcal{A}$, $P'(s'|s',a) = 1$ for any $a \in \mathcal{A}$, $r'(s,a) = r(s,a)$ for any $s \in \mathcal{S}$ and $a \in \mathcal{A}$, $r(s_0,a) = 0$ for any $a \in \mathcal{A}$, and $r(s_1,a) = 1$ for any $a \in \mathcal{A}$.

For any policy π, $\sum_{h'=1}^{H} r_{h'}(s_{h'}, a_{h'})$ equals to H with probability at least $1-\alpha$. Thus, the α-VaR following by any policy π in the transferred MDP $\mathcal{M}'$ is H. We have

$$\begin{aligned}
\text{CVaR}_\alpha^{\pi,\mathcal{M}'}(s_0) =& \max_{b_0 \in [0,H]} \{b_0 - \alpha^{-1} V_0^{\pi,\mathcal{M}'}(s_0, b_0)\} \\
=& H - \alpha^{-1} \mathbb{E}_\pi \left[\left(H - \sum_{h'=0}^{H} r'_{h'}(s_{h'}, a_{h'}) \right) \middle| s_0, \mathcal{M}' \right] \\
=& H - \alpha^{-1}\alpha \mathbb{E}_\pi \left[\left(H - \sum_{h'=1}^{H} r'_{h'}(s_{h'}, a_{h'}) \right) \middle| s_1, \mathcal{M}' \right] \\
& \underbrace{- \alpha^{-1}(1-\alpha) \mathbb{E}_\pi \left[\left(H - \sum_{h'=1}^{H} r'_{h'}(s_{h'}, a_{h'}) \right) \middle| s', \mathcal{M}' \right]}_{=0} \\
=& \mathbb{E}_\pi \left[\sum_{h'=1}^{H} r'_{h'}(s_{h'}, a_{h'}) \middle| s_1, \mathcal{M}' \right] \\
=& \mathbb{E}_\pi \left[\sum_{h'=1}^{H} r_{h'}(s_{h'}, a_{h'}) | s_1, \mathcal{M} \right].
\end{aligned} \tag{D.5}$$

□

Now we can prove our lower bound, Theorem 14. Here, we restated Theorem 4.1 in [51], which shows that any reward-free exploration algorithm that output ϵ-optimal policy must collect at least $\Omega(S^2AH^2/\alpha\epsilon^2)$ trajectories in expectation.

Theorem 15. (Theorem 4.1 in [51]) Consider a universal constant $C > 0$. For a given risk tolerance

$\alpha \in (0, 1]$, if the number of actions $A \geq 2$, the number of states $S \geq C \log_2 A$, the horizon $H \geq C \log_2 S$, and the accuracy parameter $\epsilon \leq \min\{1/4\alpha, H/48\alpha\}$, then any reward-free exploration algorithm that can output ϵ-optimal policies for an arbitrary number of adaptively chosen reward functions with a success probability $\delta = 1/2$ must collect at least $\Omega(S^2AH^2/\alpha\epsilon^2)$ trajectories in expectation.

Thus, any CVaR-RF exploration algorithm must collect at least $\Omega(S^2AH^2/\epsilon^2)$ trajectories from the state s_1, in expectation, and then collect at least $\Omega(S^2AH^2/\alpha\epsilon^2)$ trajectories from the initial state s_0.

D.4 Technical Lemmas

D.4.1 An Essential Lemma for Upper Bound

The following crucial lemma establishes a relationship between the errors at step h and those at step $h + 1$.

Lemma 13. On the event $\mathcal{E}$, for all $h \in [H]$ and $(s, a) \in \mathcal{S} \times \mathcal{A}$,

$$E_h^t(s, a) \leq 3H \left[\sqrt{\frac{\beta(n_h^t(s, a), \delta)}{n_h^t(s, a)}} \wedge 1 \right] + \sum_{s' \in \mathcal{S}} P_h(s'|s, a) E_{h+1}(s', \rho^{t+1}(s')).$$

Proof. By the definition of $E_h^t(s, a)$ and the greedy policy ρ^{t+1}, if $n_h^t(s, a) > 0$,

$$E_h^t(s, a) \leq H \sqrt{\frac{2\beta(n_h^t(s, a), \delta)}{n_h^t(s, a)}} + \sum_{s' \in \mathcal{S}} \hat{P}_h(s'|s, a) E_{h+1}(s', \rho^{t+1}(s')).$$

By the definition of $\mathcal{E}$ and Pinsker's inequality, we further have

$$\begin{aligned}
&\sum_{s'\in\mathcal{S}} \hat{P}_h(s'|s,a)E_{h+1}^t(s',\rho^{t+1}(s')) \\
&\leq \sum_{s'\in\mathcal{S}} P_h(s'|s,a)E_{h+1}^t(s',\rho^{t+1}(s')) + \sum_{s'\in\mathcal{S}} \left(\hat{P}_h(s'|s,a) - P_h^t(s'|s,a)\right) E_{h+1}^t(s',\rho^{t+1}(s')) \\
&\leq \sum_{s'\in\mathcal{S}} P_h(s'|s,a)E_{h+1}^t(s',\rho^{t+1}(s')) + \|(\hat{P}_h(\cdot|s,a) - P_h^t(\cdot|s,a)\| \cdot H \\
&\leq \sum_{s'\in\mathcal{S}} P_h(s'|s,a)E_{h+1}^t(s',\rho^{t+1}(s')) + H\sqrt{\frac{2\beta(n_h^t(s,a),\delta)}{n_h^t(s,a)}},
\end{aligned}$$

where we use the fact that $E_{h+1}^t(s',\rho^{t+1}(s') \leq H$. Therefore, plugging in this inequality and using $2\sqrt{2} \leq 3$, we have

$$E_h^t(s,a) \leq \sum_{s'\in\mathcal{S}} P_h(s'|s,a)E_{h+1}^t(s',\rho^{t+1}(s')) + 3H\sqrt{\frac{\beta(n_h^t(s,a),\delta)}{n_h^t(s,a)}}.$$

Notice that

$$E_h^t(s,a) \leq H \leq 3H \leq 3H + \sum_{s'\in\mathcal{S}} P_h(s'|s,a)E_{h+1}^t(s',\rho^{t+1}(s')),$$

and this is also true for $n_h^t(s,a) = 0$ with $1/0 = +\infty$, which leads to the conclusion. □

D.4.2 Auxiliary Lemmas

Lemma 14. $\text{VaR}_\alpha = b^\star := \arg\max_{b\in\mathbb{R}}(b - \alpha^{-1}\mathbb{E}[(b-X)^+])$.

Proof. Recall the definitions of CVaR and VaR, we have $\text{CVaR}_\alpha(X) = \sup_b \left\{b - \frac{1}{\alpha}\mathbb{E}[(b-X)^+]\right\}$, $\text{VaR}_\alpha(X) = \inf\{x \in \mathbb{R} : P(X \leq x) \geq \alpha\}$. By Theorem 6.2 in [1], we have

$$\text{CVaR}_\alpha(X) = \mathbb{E}[X|X \geq \text{VaR}_\alpha(X)].$$

Firstly, we define $f(b) = b - \frac{1}{\alpha}\mathbb{E}[(b - X)^+]$, thus the derivative of $f(b)$ with respect to b is:

$$f'(b) = 1 - \frac{1}{\alpha}P(X \geq b).$$

By setting the derivative equal to zero, we have $P(X \leq b) = 1 - \alpha$. According to the definition of VaR, b is the α-th quantile of the distribution of X, which means $b = \text{VaR}_\alpha(X)$. Therefore, the critical point b^* that maximizes $f(b)$ is equal to $\text{VaR}_\alpha(X)$. Now we prove $f(b^*) = \text{CVaR}_\alpha(X)$.

$$\begin{aligned} f(b^*) &= \text{VaR}_\alpha(X) - \frac{1}{\alpha}\mathbb{E}[(\text{VaR}_\alpha(X) - X)^+] \\ &= \text{VaR}_\alpha(X) - \frac{1}{\alpha}\int_{-\infty}^{\text{VaR}_\alpha(X)} (\text{VaR}_\alpha(X) - x)dF(x) \\ &= \frac{1}{\alpha}\int_{\text{VaR}_\alpha(X)}^{\infty} xdF(x) = \mathbb{E}[X|X \geq \text{VaR}_\alpha(X)] = \text{CVaR}_\alpha(X). \end{aligned}$$

□

Lemma 15. (Lemma F.1 in [99]) Given any $\rho \in \Pi^{\text{Aug}}$, $h \in [H]$, augmented state (s_h, b_h), and history H_h, we have $V_h^\rho(s_h, b_h) = V_h^{\pi^{\rho,b}}(s_h, b_h; H_h)$ for $b = b_h + r_1 + \ldots + r_{h-1}$. Particularly, $V_1^\rho(s_1, \cdot) = V_1^{\pi^{\rho,b}}(s_1, \cdot)$.

Lemma 16. (Lemma 10 in [53]) Given $\beta(n, \delta) = \log(2SAH/\delta) + (S-1)\log\left(e\left(1 + \frac{n}{S-1}\right)\right)$, it holds that $P(\mathcal{E}) \geq 1 - \frac{\delta}{2}$. Furthermore, $P(\mathcal{E}^{\text{cnt}}) \geq 1 - \frac{\delta}{2}$.

Lemma 17. (Lemma 7 in [53]) On the event $\mathcal{E}^{\text{cnt}}$, for all $h \in [H]$ and $(s, a) \in \mathcal{S} \times \mathcal{A}$,

$$\forall t \in \mathbb{N}^*, \frac{\beta(n_h^t(s,a), \delta)}{n_h^t(s,a)} \wedge 1 \leq 4\frac{\beta(\bar{n}_h^t(s,a), \delta)}{\bar{n}_h^t(s,a) \vee 1}.$$

Lemma 18. (Lemma 19 in [5]) For any sequence of numbers $z_1, \ldots, z_n$ with $0 \leq z_k \leq Z_{k-1} = \max\left\{1, \sum_{i=1}^{k-1} z_i\right\}$,

$$\sum_{k=1}^{n} \frac{z_k}{\sqrt{Z_{k-1}}} \leq (1 + \sqrt{2})\sqrt{Z_n}.$$

Lemma 19. (Lemma 15 in [53].) Let $n \geq 1$ and $a, b, c, d > 0$. If $n\Delta^2 \leq a + b\log(c + dn)$ then

$$n \leq \frac{1}{\Delta^2}\left[a + b\log\left(c + \frac{d}{\Delta^4}(a + b(\sqrt{c} + \sqrt{d}))^2\right)\right].$$

Bibliography

[1] C. Acerbi and D. Tasche. On the Coherence of Expected Shortfall. *Journal of Banking & Finance*, 26(7):1487–1503, 2002.

[2] A. Ahmadi-Javid. Entropic Value-at-Risk: A New Coherent Risk Measure. *Journal of Optimization Theory and Applications*, 155(3):1105–1123, 2012.

[3] M. Ang, J. Sun, and Q. Yao. On the Dual Representation of Coherent Risk Measures. *Annals of Operations Research*, 262(1):29–46, Mar. 2018.

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

[5] P. Auer, T. Jaksch, and R. Ortner. Near-Optimal Regret Bounds for Reinforcement Learning. *Advances in Neural Information Processing Systems*, 21, 2008.

[6] M. G. Azar, I. Osband, and R. Munos. Minimax Regret Bounds for Reinforcement Learning. In *Proc. International Conference on Machine Learning*, pages 263–272, Sydney, Australia, 2017.

[7] O. Bastani, J. Y. Ma, E. Shen, and W. Xu. Regret Bounds for Risk-Sensitive Reinforcement Learning. *Advances in Neural Information Processing Systems*, 35:36259–36269, 2022.

[8] N. Bäuerle and J. Ott. Markov Decision Processes with Average-Value-at-Risk Criteria. *Mathematical Methods of Operations Research*, 74:361–379, 2011.

[9] J. Baxter and P. Bartlett. Infinite-Horizon Policy-Gradient Estimation. *Journal of Artificial Intelligence Research*, 15:319–350, 2001.

[10] M. G. Bellemare, W. Dabney, and M. Rowland. *Distributional Reinforcement Learning*. MIT Press, 2023.

[11] B. Belousov and J. Peters. f-Divergence Constrained policy Improvement. *arXiv preprint arXiv:1801.00056*, 2017.

[12] A. Ben-Tal, D. Den Hertog, A. De Waegenaere, B. Melenberg, and G. Rennen. Robust Solutions of Optimization Problems Affected by Uncertain Probabilities. *Management Science*, 59(2):341–357, 2013.

[13] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski. *Robust Optimization*. Princeton University Press, Princeton, NJ, 2009.

[14] M. Benaïm. Dynamics of Stochastic Approximation Algorithms. In *Seminaire de Probabilites XXXIII*, pages 1–68. Springer, 2006.

[15] D. Bertsekas. *Dynamic Programming and Optimal Control: Volume I*, volume 1. Athena Scientific, Nashua, NA, 2012.

[16] M. Best and R. Grauer. On the Sensitivity of Mean-Variance-Efficient Portfolios to Changes in Asset Means: Some Analytical and Computational Results. *The Review of Financial Studies*, 4(2):315–342, 1991.

[17] B. Bixby. The Gurobi Optimizer. *Transp. Re-search Part B*, 41(2):159–178, 2007.

[18] K. Boda and J. A. Filar. Time Consistent Dynamic Risk Measures. *Mathematical Methods of Operations Research*, 63(1):169–186, 2006.

[19] V. S Borkar. *Stochastic Approximation: A Dynamical Systems Viewpoint*, volume 48. Springer, 2009.

[20] G. Brockman. OpenAI Gym. *arXiv preprint arXiv:1606.01540*, 2016.

[21] J. Chen, A. Modi, A. Krishnamurthy, N. Jiang, and A. Agarwal. On the Statistical Efficiency of Reward-Free Exploration in Non-Linear RL. *Advances in Neural Information Processing Systems*, 35:20960–20973, 2022.

[22] Y. Chow and M. Ghavamzadeh. Algorithms for CVaR Optimization in MDPs. *Advances in Neural Information Processing Systems*, 27, Dec, 2014.

[23] Y. Chow, M. Ghavamzadeh, L. Janson, and M. Pavone. Risk-Constrained Reinforcement Learning with Percentile Risk Criteria. *The Journal of Machine Learning Research*, 18(1):6070–6120, 2017.

[24] Y. Chow, A. Tamar, S. Mannor, and M. Pavone. Risk-Sensitive and Robust Decision-Making: A CVaR Optimization Approach. *Advances in Neural Information Processing Systems*, 28, Dec, 2015.

[25] E. J. Collins. Using Markov Decision Processes To Optimize A Nonlinear Functional of the Final Distribution, with Manufacturing Applications. In *Stochastic Modelling in Innovative Manufacturing*, pages 30–45. Springer, 1997.

[26] C. Dann and E. Brunskill. Sample Complexity of Episodic Fixed-Horizon Reinforcement Learning. *Advances in Neural Information Processing Systems*, 28, 2015.

[27] C. Dann, T. Lattimore, and E. Brunskill. Unifying PAC andRregret: Uniform PAC bounds for Episodic Reinforcement Learning. *Advances in Neural Information Processing Systems*, 30, 2017.

[28] E. Delage and S. Mannor. Percentile Optimization for Markov Decision Processes with Parameter Uncertainty. *Operations Research*, 58(1):203–213, 2010.

[29] D. Di Castro, A. Tamar, and S. Mannor. Policy Gradients with Variance Related Risk Criteria. *arXiv preprint arXiv:1206.6404*, 2012.

[30] Y. Ding, M. Jin, and J. Lavaei. Non-Stationary Risk-Sensitive Reinforcement Learning: Near-Optimal Dynamic Regret, Adaptive Detection, and Separation Design. In *Proc. the AAAI Conference on Artificial Intelligence*, volume 37, pages 7405–7413, Washington, DC, 2023.

[31] Y. Du, S. Wang, and L. Huang. Provably Efficient Risk-Sensitive Reinforcement Learning: Iterated CVaR and Worst Path. In *Proc. International Conference on Learning Representations*, 2022.

[32] Y. Fei, Z. Yang, Y. Chen, Z. Wang, and Q. Xie. Risk-Sensitive Reinforcement Learning: Near-Optimal Risk-Sample Tradeoff in Regret. *Advances in Neural Information Processing Systems*, 33:22384–22395, 2020.

[33] Y. Fei, Z. Yang, and Z. Wang. Entropic Risk-Sensitive Reinforcement Learning: A Meta Regret Framework with Function Approximation. 2020.

[34] Y. Fei, Z. Yang, and Z. Wang. Risk-Sensitive Reinforcement Learning with Function Approximation: A Debiasing Approach. In *Proc. International Conference on Machine Learning*, pages 3198–3207, 2021.

[35] C. N. Fiechter. Efficient Reinforcement Learning. In *Proc. Annual Conference on Computational Learning Theory*, pages 88–97, New Brunswick, NJ, 1994.

[36] K. Fisher and M. Statman. The Mean-Variance-Optimization Puzzle: Security Portfolios and Food Portfolios. *Financial Analysts Journal*, 53(4):41–50, 1997.

[37] H. Föllmer and T. Knispel. Entropic Risk Measures: Coherence vs. Convexity, Model Ambiguity and Robust Large Deviations. *Stochastics and Dynamics*, 11:333–351, 2011.

[38] H. Föllmer and A. Schied. Convex Measures of Risk and Trading Constraints. *Finance and Stochastics*, 6(4):429–447, Oct. 2002.

[39] M. Godbout, M. Heuillet, S. Chandra, R. Bhati, and A. Durand. ACRel: Adversarial Conditional Value-at-Risk Reinforcement Learning. *arXiv preprint arXiv:2109.09470*, 2021.

[40] M. Godbout, M. Heuillet, S. Chandra, R. Bhati, and A. Durand. CARL: Conditional-Value-at-Risk Adversarial Reinforcement Learning. *arXiv preprint arXiv:2109.09470*, Sep. 2021.

[41] C. Gong, Q. He, Y. Bai, Z. Yang, X. Chen, X. Hou, X. Zhang, Y. Liu, and G. Fan. The f-Divergence Reinforcement Learning Framework. *arXiv preprint arXiv:2109.11867*, 2021.

[42] R. Green and B. Hollifield. When Will Mean-Variance Efficient Portfolios Be Well Diversified? *The Journal of Finance*, 47(5):1785–1809, 1992.

[43] J. Hau, M. Petrik, and M. Ghavamzadeh. Entropic Risk Optimization in Discounted MDPs. In *Proc. International Conference on Artificial Intelligence and Statistics*, pages 47–76, Tamil Nadu, India, Feb, 2023.

[44] J. Hau, M. Petrik, M Ghavamzadeh, and R. Russel. RASR: Risk-Averse Soft-Robust MDPs with EVaR and Entropic Risk. *arXiv preprint arXiv:2209.04067*, 2022.

[45] C. Ho, M. Petrik, and W. Wiesemann. Fast Bellman Updates for Robust MDPs. In *Proc. International Conference on Machine Learning*, pages 1979–1988, Stockholm, Sweden, July. 2018.

[46] C. Pang Ho, M. Petrik, and W. Wiesemann. Robust Phi-Divergence MDPs. *arXiv preprint arXiv:2205.14202*, 2022.

[47] R. A. Howard and J. E. Matheson. Risk-Sensitive Markov Decision Processes. *Management Science*, 18(7):356–369, Mar. 1972.

[48] G. Iyengar. Robust Dynamic Programming. *Mathematics of Operations Research*, 30(2):257–280, 2005.

[49] A. Jain and A. Orlitsky. A General Method for Robust Learning from Batches. *Advances in Neural Information Processing Systems*, 33:21775–21785, Dec, 2020.

[50] C. Jin, Z. Allen-Zhu, S. Bubeck, and M. I Jordan. Is Q-Learning Provably Efficient? *Advances in Neural Information Processing Systems*, 31, 2018.

[51] C. Jin, A. Krishnamurthy, M. Simchowitz, and T. Yu. Reward-Free Exploration for Reinforcement Learning. In *Proc. International Conference on Machine Learning*, pages 4870–4879, Stockholm, Sweden, 2020.

[52] H. Kashima. Risk-Sensitive Learning via Minimization of Empirical Conditional Value-at-Risk. *IEICE Transactions on Information and Systems*, 90(12):2043–2052, Dec. 2007.

[53] E. Kaufmann, P. Ménard, O. D. Domingues, A. Jonsson, E. Leurent, and M. Valko. Adaptive Reward-Free Exploration. In *Proc. Algorithmic Learning Theory*, pages 865–891, Paris, France, 2021.

[54] L. Ke, S. Choudhury, M. Barnes, W. Sun, G. Lee, and S. Srinivasa. Lmitation Learning as f-Divergence Minimization. In *Algorithmic Foundations of Robotics XIV: Proceedings of Workshop on the Algorithmic Foundations of Robotics 14*, pages 313–329. Springer, 2021.

[55] R. Keramati, C. Dann, A. Tamkin, and E. Brunskill. Being Optimistic to Be Conservative: Quickly Learning a CVaR Policy. In *Proc. The AAAI conference on artificial intelligence*, volume 34, pages 4436–4443, New York, NY, Feb, 2020.

[56] H. K. Khalil and J. Grizzle. Nonlinear Systems. Vol. 3 Prentice Hall. *New Jersey*, 1996.

[57] D. Kim and S. Oh. TRC: Trust Region Conditional Value at Risk for Safe Reinforcement Learning. *IEEE Robotics and Automation Letters*, 7(2):2621–2628, 2022.

[58] P. La and M. Ghavamzadeh. Actor-Critic Algorithms for Risk-Sensitive MDPs. *Advances in Neural Information Processing Systems*, 26, Dec, 2013.

[59] Y. Le Tallec. Robust, Risk-Snsitive, and Data-Driven Control of Markov Decision Processes. 2007.

[60] L. Li. Sample Somplexity Bounds of Exploration. In *Reinforcement Learning: State-of-the-Art*, pages 175–204. Springer, 2012.

[61] S. H. Lim and I. Malik. Distributional Reinforcement Learning for Risk-Sensitive Policies. *Advances in Neural Information Processing Systems*, 35:30977–30989, 2022.

[62] F. C. Lunenburg. The Decision Making Process. In *National Forum of Educational Administration & Supervision Journal*, volume 27, 2010.

[63] F. Luo and S. Mehrotra. Distributionally Robust Optimization with Decision Dependent Ambiguity Sets. *Optimization Letters*, 14:2565–2594, 2020.

[64] X. Ma, L. Xia, Z. Zhou, J. Yang, and Q. Zhao. DSAC: Distributional Soft Actor Critic for Risk-Sensitive Reinforcement Learning. *arXiv preprint arXiv:2004.14547*, 2020.

[65] S. Mannor, D. Simester, P. Sun, and J. N. Tsitsiklis. Bias and Variance Approximation in Value Function Estimates. *Management Science*, 53(2):308–322, Feb. 2007.

[66] P. Marbach. Simulated-Based Methods for Markov Decision Processes. 1998.

[67] P. Ménard, O. D. Domingues, A. Jonsson, E. Kaufmann, E. Leurent, and M. Valko. Fast Active Learning for Pure Exploration in Reinforcement Learning. In *Proc. International Conference on Machine Learning*, pages 7599–7608, 2021.

[68] S. Miryoosefi and C. Jin. A Simple Reward-Free Approach to Constrained Reinforcement Learning. In *Proc. International Conference on Machine Learning*, pages 15666–15698, Baltimore, MD, 2022.

[69] T. Morimura, M. Sugiyama, H. Kashima, H. Hachiya, and T. Tanaka. Nonparametric Return Distribution Approximation for Reinforcement Learning. In *Proc. International Conference on Machine Learning*, 2010.

[70] X. Ni and L. Lai. Policy Gradient Based Entropic-VaR Optimization in Risk-Sensitive Reinforcement Learning. In *Proc. Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 1–6, Allerton, IL, 2022.

[71] X. Ni and L. Lai. Risk-Sensitive Reinforcement Learning via Entropic-VaR Optimization. In *Proc. Asilomar Conference on Signals, Systems, and Computers*, pages 953–959, Pacific Grove, CA, 2022.

[72] X. Ni and L. Lai. Risk-Sensitive Reinforcement Learning with ϕ-Divergence-Risk. *IEEE Transactions on Information Theory*, 2024. Submitted.

[73] X. Ni and L. Lai. Robust Risk-Sensitive Reinforcement Learning with Conditional Value-at-Risk. In *Proc. IEEE Information Theory Workshop*, Shenzhen, China, 2024.

[74] X. Ni, G. Liu, and L. Lai. Risk-Sensitive Reward-Free Reinforcement Learning with CVaR. In *Proc. International Conference on Machine Learning*, Vienna, Austria, Jul, 2024.

[75] A. Nilim and L. El Ghaoui. Robust Control of Markov Decision Processes with Uncertain Transition Matrices. *Operations Research*, 53(5):780–798, Oct. 2005.

[76] A. Nilim and L. Ghaoui. Robustness in Markov Decision Problems with Uncertain Transition Matrices. *Advances in Neural Information Processing Systems*, 16, 2003.

[77] O. Nohadani and K. Sharma. Optimization Under Decision-Dependent Uncertainty. *SIAM Journal on Optimization*, 28(2):1773–1795, 2018.

[78] I Gurobi Optimization et al. Gurobi Optimizer Reference Manual, 2018. *URL http://www. gurobi. com*, 2018.

[79] T. Osogami. Robustness and Risk-Sensitivity in Markov Decision Processes. In *Proc. Advances in Neural Information Processing Systems*, volume 25, pages 233–241, Lake Tahoe, NV, Dec. 2012.

[80] J. T. Ott. A Markov Decision Model For A Surveillance Application and Risk-Sensitive Markov Decision Processes. 2010.

[81] M. Petrik and D. Subramanian. An Approximate Solution Method For Large Risk-Averse Markov Decision Processes. *arXiv preprint arXiv:1210.4901*, 2012.

[82] G. C. Pflug and A. Pichler. Time-Consistent Decisions and Temporal Decomposition of Coherent Risk Functionals. *Mathematics of Operations Research*, 41(2):682–699, May. 2016.

[83] L. A. Prashanth. Policy Gradients for CVaR-Constrained MDPs. In *Proc. International Conference on Algorithmic Learning Theory*, pages 155–169, Bled, Slovenia, 2014.

[84] L. A. Prashanth, Michael C Fu, et al. Risk-Sensitive Reinforcement Learning via Policy Gradient Search. *Foundations and Trends® in Machine Learning*, 15(5):537–693, 2022.

[85] R. T. Rockafellar and S. Uryasev. Optimization of Conditional Value-at-Risk. *Journal of Risk*, 2:21–42, 2000.

[86] R. T. Rockafellar and S. Uryasev. Conditional Value-at-Risk for General Loss Distributions. *Journal of Banking & Finance*, 26(7):1443–1471, Jul. 2002.

[87] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz. Trust Region Policy Optimization. In *Proc. International Conference on Machine Learning*, pages 1889–1897, Lille, France, July, 2015.

[88] A. Shapiro, D. Dentcheva, and A. Ruszczynski. *Lectures On Stochastic Programming: Modeling and Theory*. SIAM, Philadelphia, PA, 2021.

[89] A. Shapiro, W. Tekaya, J. P. Da Costa, and M. P. Soares. Risk Neutral and Risk Averse Stochastic Dual Dynamic Programming Method. *European Journal of Operational Research*, 224(2):375–391, 2013.

[90] Y. Shen, M. J Tobia, T. Sommer, and K. Obermayer. Risk-Sensitive Reinforcement Learning. *Neural Computation*, 26(7):1298–1328, 2014.

[91] R. Singh, Q. Zhang, and Y. Chen. Improving Robustness via Risk Averse Distributional Reinforcement Learning. In *Proc. Learning for Dynamics and Control*, pages 958–968, Cambridge, MA, 2020.

[92] M. J. Sobel. The Variance of Discounted Markov Decision Processes. *Journal of Applied Probability*, 19(4):794–802, 1982.

[93] W. Sun, S. Rachev, F. Fabozzi, and P. Kalev. Long-Range Dependence and Heavy Tailedness in Modelling Trade Duration. *Working Paper, University of Karlsruhe*, 2005.

[94] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT press, Cambridge, MA, 2018.

[95] A. Tamar, Y. Chow, M. Ghavamzadeh, and S. Mannor. Policy Gradient for Coherent Risk Measures. *Advances in Neural Information Processing Systems*, 28, Dec, 2015.

[96] A. Tamar, Y. Glassner, and S. Mannor. Policy Gradients Beyond expectations: Conditional Value-at-Risk. *arXiv preprint arXiv:1404.3862*, 2014.

[97] A. Tamar, Y. Glassner, and S. Mannor. Optimizing the CVaR via Sampling. In *Proc. The AAAI Conference on Artificial Intelligence*, volume 29, Austin, TX, Feb, 2015.

[98] A. Tamar, S. Mannor, and H. Xu. Scaling Up Robust MDPs Using Function Approximation. In *Proc. International Conference on Machine Learning*, pages 181–189, Beijing, China, Jul. 2014.

[99] K. Wang, N. Kallus, and W. Sun. Near-Minimax-Optimal Risk-Sensitive Reinforcement Learning with CVaR. In *Proc. International Conference on Machine Learning*, pages 35864–35907, Honolulu, HI, Jul, 2023.

[100] R. Wang, S. S Du, L. Yang, and R. R Salakhutdinov. On Reward-Free Reinforcement Learning with Linear Function Approximation. *Advances in Neural Information Processing Systems*, 33:17816–17826, 2020.

[101] Y. Wang and S. Zou. Online Robust Reinforcement Learning with Model Uncertainty. *Advances in Neural Information Processing Systems*, 34:7193–7206, 2021.

[102] W. Wiesemann, D. Kuhn, and B. Rustem. Robust Markov Decision Processes. *Mathematics of Operations Research*, 38(1):153–183, 2013.

[103] R. J. Williams. Simple Statistical Gradient-Following Algorithms For Connectionist Reinforcement Learning. *Machine Learning*, 8(3):229–256, 1992.

[104] C. Ying, X. Zhou, H. Su, D. Yan, N. Chen, and J. Zhu. Towards Safe Reinforcement Learning via Constraining Conditional Value-at-Risk. *arXiv preprint arXiv:2206.04436*, 2022.

[105] P. Ying, R. He, J. Mao, Q. Zhang, H. Reith, J. Sui, Z. Ren, K. Nielsch, and G. Schierning. Towards Tellurium-Free Thermoelectric Modules for Power Generation from Low-Grade Heat. *Nature Communications*, 12(1):1–6, Feb. 2021.

[106] K. Zhang, T. Sun, Y. Tao, S. Genc, S. Mallya, and T. Basar. Robust Multi-Agent Reinforcement Learning with Model Uncertainty. *Advances in Neural Information Processing Systems*, 33:10571–10583, 2020.

[107] Z. Zhang, S. Du, and X. Ji. Near Optimal Reward-Free Reinforcement Learning. In *Proc. International Conference on Machine Learning*, pages 12402–12412, 2021.