EVaR Optimization for Risk-Sensitive Reinforcement Learning

Xinyi Ni and Lifeng Lai

Abstract

In the existing work on risk-sensitive reinforcement learning (RL) problems, in order to take uncertainty into consideration, risk measure such as conditional value-at-risk (CVaR) has been widely used to design robust RL algorithms. However, the uncertainty set in the dual representation of CVaR is defined by distributions whose Radon-Nikodym derivative is constrained to a certain range. This is a less common way to define distribution neighborhood in machine learning applications and hence its interpretation is less natural. This paper applies a recently developed risk measure named entropic value-at-risk (EVaR) to risk-sensitive RL problems. One appealing feature of EVaR is that the uncertainty set in its dual representation is defined by distributions whose Kullback-Leibler (KL) distance to the nominal distribution is less or equal to a certain level. Hence EVaR has a very natural interpretation for RL applications. In this paper, we address the EVaR optimization problem based on Markov decision process (MDP) by proposing a value iteration algorithm as well as its approximate version equipped with linear interpolation. Furthermore, for the case where the nominal transition kernel of the underlying MDP is unknown, we present a sample-based counterpart for the value iteration algorithm. Numerical examples are also provided to illustrate these proposed algorithms.

I. Introduction

Reinforcement learning (RL) [1] is an area of machine learning where agents learn from the environment to determine the actions. The environment is typically stated as a Markov decision process (MDP). 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 [2]. Despite the popularity of the *risk-neutral* approach, 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 [3] when there are uncertainties or modeling errors.

The uncertainty of the cost can be addressed in *risk-sensitive* MDPs [4] by utilizing risk measures. 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. Artzner et al. [5] propose 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. Follmer et al. [6] extend 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. The sensitive issue could be solved in *robust* MDPs by choosing some uncertainty sets to model the uncertainty and considering the worst case [7] over these uncertainty sets. Osogami [8] 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.

In risk-sensitive MDPs, one well-known risk measure is *value-at-risk* (VaR). However, VaR is not coherent due to the lack of subadditivity and convexity [5]. Furthermore, VaR is unstable and difficult to optimize when the costs are not normally distributed [9], [10]. To address these shortcomings, Rockafellar and Uryasev developed a new risk measure called *conditional value-at-risk* (CVaR) in [9] and [10]. CVaR

X. Ni and L. Lai are with the Department of Electrical and Computer Engineering, University of California, Davis, CA. Email: {xni,lflai}@ucdavis.edu. The work of X. Ni and L. Lai was supported by the National Science Foundation under Grants CCF-1717943, CNS-1824553, CCF-1908258 and ECCS-2000415.

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 [11]–[18]. However, as will be detailed in Section II, 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 [11], 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 *entropic value-at-risk* (EVaR) developed recently by Ahmadi et al. [19]. 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 [19]. One appealing feature of EVaR is the uncertainty set in its dual representation. In particular, [19] shows that the uncertainty set in the dual representation of EVaR is defined by distributions whose Kullback-Leibler (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. Due to the coherent property of EVaR, we can apply the alternative dual representation for EVaR in [19] 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 [20] 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 [11] 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 [21] and [22] 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.

The remainder of this paper is organized as follows. In Section III, we introduce the problem formulation. In Section IV, we describe our value iteration algorithms. We also introduce a more practical approximation version using linear interpolation. In Section V, we consider cases where the underlying MDP model is unknown and present a sample-base algorithm. In Section VI, we provide numerical examples to illustrate our approach. Finally, we offer concluding remarks in Section VII.

II. Preliminaries

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 [19].
(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 $\lambda \geq 0$
One very useful property of coherent risk measures is the dual representation theorem [21], which connects the risk-sensitiveness to robustness. Examples of coherent measures include the *conditional value-at-risk* and *entropic value-at-risk* [19] etc. Examples of non coherent measures include *variance*, *mean-standard-deviation* and *value-at-risk* etc [5].

In the following, we review risk measures that are directly related to our work. Let Z be a bounded random variable on the probability space $(\Omega, \mathcal{F}, P)$ with the cumulative distribution function (CDF) $F(z) = P(Z \leq z)$. The value-at-risk (VaR) [5] at confidence level $\alpha \in [0, 1]$ is the $1 - \alpha$ quantile of Z. Since we interpret Z as a cost in this paper, VaR is defined as:

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

However, VaR is not a coherent risk measure as it lacks the subadditivity and convexity properties [5]. In order to overcome the shortcomings of VaR, conditional value-at-risk (CVaR) is proposed in [9] and [10]. CVaR is defined as the mean of the worst $\alpha\%$ of values of Z, i.e.,

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

where $(z)^+ = \max(z, 0)$. CVaR is a coherent risk measure. 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 has been extensively applied to RL problems [11]–[18].

As mentioned above, for each coherent risk measure, there is a useful alternative dual representation [19]. 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 [23]:

$$\text{CVaR}_\alpha(Z) = \sup_{Q \in \mathcal{U}_{\text{CVaR}}} \mathbb{E}_Q(Z), \tag{1}$$

where

$$\mathcal{U}_{\text{CVaR}} = \left\{ Q \ll P : \frac{dQ}{dP} \in \left[0, \frac{1}{\alpha}\right] \right\}. \tag{2}$$

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 (2). While the neighborhood defined in $\mathcal{U}_{\text{CVaR}}$

is certainly relevant for some RL applications [11], it is a less common way to define distribution neighborhood in machine learning applications.

Recently, in [19], Ahmadi et al. propose a risk measure named entropic value-at-risk (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 $M_Z(t) = \mathbb{E}_P\left[e^{tZ}\right]$ exists for $t \geq 0$. The EVaR of a random variable $Z \in L_{M^+}$ with confidence level $1 - \alpha$ is defined as

$$\mathrm{EVaR}_\alpha(Z) = \inf_{t>0} \left\{ t^{-1} \ln(M_Z(t)) - t^{-1} \ln \alpha \right\}. \tag{3}$$

[19] shows that EVaR is the tightest upper bound for both VaR and CVaR. Similar to CVaR, EVaR is a coherent risk measure and EVaR_α is decreasing in α, i.e, EVaR_α tends to $\max(Z)$ as α decreasing to 0 and $\mathrm{EVaR}_1(Z)$ equals $\mathbb{E}(Z)$. One appealing feature of EVaR is its dual representation [19]:

$$\mathrm{EVaR}_\alpha(Z) = \sup_{Q \in \mathcal{U}_{\mathrm{EVaR}}} \mathbb{E}_Q(Z), \tag{4}$$

where

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

Here D_{KL} refers to 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 (4), we can see that $\mathrm{EVaR}_\alpha(Z)$ 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 distance) of P. Compared with the dual representation of CVaR, it's more common and natural to use KL distance 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.

III. EVaR Optimization

As mentioned in Section I, solving risk-sensitive decision making with a coherent risk measure ensures the robustness of the developed algorithms. In addition, as discussed in Section II, the dual representation of EVaR has a very natural way of defining the uncertainty set, which is widely used in machine learning problems. Motivated by these observations, in this paper, we apply EVaR to design robust algorithms for risk-sensitive RL.

In particular, we consider a Markov decision process represented by a tuple $(\mathcal{X}, \mathcal{A}, C, P, \gamma, x_0)$, where $\mathcal{X}$ is the state space, $\mathcal{A}$ is the action space, $C(x, a) \in [-C_{max}, C_{max}]$ is a bounded deterministic cost, $P(\cdot|x, a)$ is the transition probability distribution, $\gamma \in [0, 1]$ is the discounting factor, and x_0 is the initial state. For each state $x \in \mathcal{X}$, $\mathcal{A}(x)$ denotes the corresponding action set. For convenience, here we define some feasible set of policies μ. 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 μ_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., $\Pi_{H,t} := \{\mu_0 : H_0 \to \mathcal{A}, \mu_1 : H_1 \to \mathcal{A}, \ldots, \mu_t : H_t \to \mathcal{A} | \mu_j(\cdot|h_j) \in \mathcal{A}$ 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 $\Pi_{M,t} := \{\mu_0 : \mathcal{X} \to \mathcal{A}, \mu_1 : \mathcal{X} \to \mathcal{A}, \ldots, \mu_t : \mathcal{X} \to \mathcal{A} | \mu_j(\cdot|x_j) \in \mathcal{A}$ 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., $\mu_j = \mu$ for all $j \geq 0$.

The goal of this paper is to minimize the EVaR value of the total discounted cost and determine the corresponding optimal policies. As a result, the problem formulation of EVaR optimization in risk-sensitive reinforcement learning can be written as

$$\min_{\mu \in \Pi_H} \mathrm{EVaR}_\alpha \left(\lim_{T\to\infty} \sum_{t=0}^{T} \gamma^t C(x_t, a_t) \middle| x_0, \mu \right), \tag{5}$$

where $\mu = \{\mu_0, \mu_1, \dots\}$ is the policy sequence with action $a_t = \mu_t(h_t)$ for $t = \{0, 1, \dots\}$, $C(x_t, a_t)$ is the state-wise cost observed along a trajectory at time t and T is the length of time horizons.

Now, let P_S be the true probability measure of the total discounted cost under policy μ and Q_S denote another probability measure over this space. Using the dual representation (4) of EVaR, we can write the optimization problem (5) as

$$\min_{\mu \in \Pi_H} \sup_{Q_S \in \mathcal{U}_{\text{EVaR}}(\alpha, P_S)} \mathbb{E}_{Q_S} \left(\lim_{T \to \infty} \sum_{t=0}^{T} \gamma^t C(x_t, a_t) \middle| x_0, \mu \right),$$

where

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

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 IV, 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.

IV. Value Iteration for EVaR

In order to solve the primary optimization problem (5), we follow the idea of dynamic programming and apply the decomposition theorem of version independent risk measures used in [11] [20]. One important function in reinforcement learning 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 (5).

To begin with, we introduce the decomposition theorem for conditional EVaR. Firstly, equipped with the dual representation for EVaR in [19] and the definition of conditional risk measures in [20], 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*

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

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

Then, we introduce *version independent* risk measures mentioned in [20]. 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., $P(Z_1 \le z) = P(Z_2 \le z)$ for all $z \in \mathbb{R}$. By Corollary 3.1 in [19], we know EVaR is a version independent risk functional. Now, we can apply Theorem 21 in [20] to propose the EVaR decomposition theorem.

Theorem 1. *For any $\tau > t \ge 0$, 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*

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

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 paper, 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 μ, 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 μ at x_t. Recall the uncertainty set in EVaR dual representation,

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

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

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

Then the decomposition in Theorem 1 can be rewritten as

$$\textit{EVaR}_\alpha(Z|H_t, \mu) = \textit{esssup}\, \mathbb{E}_P[\xi(x_{t+1}) \cdot \textit{EVaR}_{\alpha\xi(x_{t+1})}(Z|H_{t+1}, \mu)|H_t, \mu], \tag{6}$$

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

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 (6), following the idea in [11], 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:*

$$V(x, y) = \min_{\mu \in \Pi_H} \textit{EVaR}_y \left(\lim_{T \to \infty} \sum_{t=0}^{T} \gamma^t C(x_t, a_t) | x_0 = x, \mu \right). \tag{7}$$

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:*

$$\mathbf{T}[V](x, y) = \min_{a \in \mathcal{A}} \left[C(x, a) + \gamma \max_{\xi \in \mathcal{U}_{\textit{EVaR}}(y, P(\cdot|x,a))} \sum_{x' \in \mathcal{X}} \xi(x') V(x', y\xi(x')) P(x'|x, a) \right]. \tag{8}$$

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* (8) *is concave. Furthermore,* $y\mathbf{T}[V](x, y)$ *is concave in y.*

Proof. Please refer to Appendix A 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 (5) 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* (5), *i.e.,*

$$V^*(x, y) = \min_{\mu \in \Pi_H} \textit{EVaR}_y \left(\lim_{T \to \infty} \sum_{t=0}^{T} \gamma^t C(x_t, a_t) | x_0 = x, \mu \right). \tag{9}$$

Proof. Please refer to Appendix B. □

We now discuss how to determine the optimal policy from V^*. Although the original optimization problem (5) 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:*

$$u^*(x_k, y_k) = a_k^*, \forall k \geq 0, \tag{10}$$

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

$$x_k \sim P\big(\cdot | x_{k-1}, a_{k-1}^* \big), y_k = y_{k-1} \xi_{x_{k-1}, y_{k-1}, a_{k-1}^*}(x_k), \tag{11}$$

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* (5) *with initial state* x_0 *and confidence level* α.

Proof. Please refer to Appendix C. □

Equipped with Theorem 2 and Theorem 3, we can now design a value iteration process to solve the EVaR optimization problem in (5).

However, Algorithm 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 [11]. 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 $\mathbf{T}$.

Algorithm 1: EVaR Value Iteration

```
For any $x \in \mathcal{X}$ and $y \in \mathcal{Y}$, arbitrarily choose $V_0(x, y)$.
For $t = 0, 1, 2, \dots$ and all $(x, y) \in \mathcal{X} \times \mathcal{Y}$, recursively applying the EVaR Bellman operator as
    $$V_{t+1}(x, y) = \mathbf{T}[V_t](x, y),$$
    and then get the optimal value function by $V^*(x, y) = \lim_{t \to \infty} V_t(x, y)$.
Selecting the specific initial state $x_0$ and confidence level $\alpha$, the solution of EVaR optimization
    problem can be immediately obtained as $V^*(x_0, \alpha)$.
Following Theorem 3, one can derive an optimal Markovian policy w.r.t $V^*(x, y)$.
```

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 $Y(x) = \{y_1, y_2, \dots, y_{N(x)}\} \in [0, 1]^{N(x)}$ 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:

$$\mathbf{T}_{\mathcal{I}}[V](x, y) = \min_{a \in \mathcal{A}} \left[C(x, a) + \gamma \max_{\xi \in \mathcal{U}_{\text{EVaR}}(y, P(\cdot|x,a))} \sum_{x' \in \mathcal{X}} \frac{\mathcal{I}_{x'}[V](y\xi(x'))}{y} P(x'|x, a) \right]. \tag{12}$$

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, $\mathbf{T}_I[V](x, 0) = \min_{a \in \mathcal{A}} \left[C(x, a) + \gamma \max_{x' \in \mathcal{X} : P(x'|x,a) > 0} V(x', 0) \right]$.

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 (12). 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 [11] can be used to bound the error between EVaR value iteration and approximate EVaR value iteration. In particular, suppose that Assumption 1 is satisfied and $\epsilon > 0$ is an error tolerance parameter. For 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 [11], one can show that Algorithm 2 has the following error bound:

$$\frac{-\gamma}{1-\gamma} O((\theta - 1) + \epsilon) \leq \hat{V}^*(x_0, a) - \min_{\mu \in \Pi_H} \text{EVaR}_\alpha (\lim_{T \to \infty} \sum_{t=0}^{T} \gamma^t C(x_t, a_t) | x_0, \mu) \leq 0$$

Algorithm 2: EVaR Value Iteration with Linear Interpolation

```
Choose the set of interpolation points $Y(x)$ and the initial value function $V_0(x, y)$ satisfying Assumption 1.
For $t = 1, 2, \dots$, for each $x \in \mathcal{X}$ and each $y_i \in Y(x)$, update the estimate of value function by
    $V_t(x, y_i) = \mathbf{T}_{\mathcal{I}}[V_{t-1}](x, y_i)$,
  and then get the near-optimal value function by $\hat{V}^*(x, y_i) = \lim_{t\to\infty} V_t(x, y_i)$.
Selecting the specific initial state $x_0$ and confidence level $\alpha$, the solution of EVaR optimization problem with linear interpolation can be immediately obtained as $\hat{V}^*(x_0, \alpha)$.
Following Theorem 3, one can derived an optimal policy w.r.t $\hat{V}^*(x, y)$.
```

and the following finite time convergence error bound:

$$\left| \mathbf{T}_I^n[V_0](x_0, \alpha) - \min_{\mu\in\Pi_H} \text{EVaR}(\lim_{T\to\infty} \sum_{t=0}^{T} \gamma^t C(x_t, a_t)|x_0, \mu) \right| \le \frac{O((\theta - 1) + \epsilon)}{1 - \gamma} + O(\gamma^n).$$

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.

V. Linear Interpolated EVaR with sample average approximation

In Section IV, 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 (5). 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*

$$Q^*(x, y, a) = \min_{\mu\in\Pi_H} \textit{EVaR}_y \left(\lim_{T\to\infty} \sum_{t=0}^{T} \gamma^t C(x_t, a_t)|x_0 = x, a_0 = a, u \right).$$

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*

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

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*

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

and the corresponding interpolated value iteration update:

$$Q(x, y, a) := C(x, a) + \gamma \max_{\xi\in\mathcal{U}_{\textit{EVaR}}(y, P(\cdot|x,a))} \sum_{x'\in\mathcal{X}} \frac{\mathcal{I}_{x'}[V](y\xi(x'))}{y} P(x'|x, a). \tag{13}$$

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 a sample average approximation (SAA) approach introduced in [21] and [22] 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 $\{x'^{,1}, \dots, x'^{,N_k}\} \sim P(x'|x,a)$. Based on these samples, we can calculate the empirical transition probability $P_{N_k}(x'|x,a)$ by

$$P_{N_k}(x'|x,a) = \frac{1}{N_k} \sum_{i=1}^{N_k} \mathbf{1}\{x'^{,i} = x'|x,a\}, \forall x, x' \in \mathcal{X}, a \in \mathcal{A}, \tag{14}$$

and replace the inner maximization problem in (13) 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 [24], 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 [21]. Now we can derive a sample-based EVaR algorithm as described in Algorithm 3.

Algorithm 3: Sample-based EVaR algorithm

```
1: 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 $(x'^{,1}, \dots, x'^{,N_k})$ and calculate the empirical transition probability
   $P_{N_k}(x'|x,a)$ by (14). Then, at iteration $k = 1, 2, \dots$, for each state $x$ and action $a$, update the
   state-action value function as follows:
   $Q_{k+1}(x,y,a) = Q_k(x,y,a) + \beta_k(x,y,a) \cdot \Big( - Q_k(x,y,a) + C(x,a) + \gamma \max_{\xi \in \mathcal{U}_{\text{EVaR}}(y, P_{N_k}(\cdot|x,a))} \frac{1}{N_k} \sum_{i=1}^{N_k} \frac{\mathcal{I}_{x'^{,i}}[V_k](y\xi(x'^{,i}))}{y} \Big).$   (15)
   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
   $\sum_k \beta_k(x,y,a) = \infty, \quad \sum_k \beta_k^2(x,y,a) < \infty.$   (16)
3: After the state-action value function converges, a near-optimal policy can be constructed as
   $\tilde{\mu}^*(x,y) \in arg \min_{a \in \mathcal{A}} Q_{\bar{k}(x,y,a)}, \quad \forall x \in \mathcal{X}, \forall y \in \mathbf{Y}$   (17)
   where $\bar{k}$ is the iteration index when the learning is stopped.
```

In Algorithm 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 $(x'^{,1}, \dots, x'^{,N_k})$ and calculate the empirical transition probability $P_{N_k}(x'|x,a)$ by (14). In the iteration process, we update the state-action value function by equation (15) with step size satisfying (16) 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 theorem, we provide the convergence of Algorithm 3.

Theorem 4. *Suppose the step size $\beta_k(x, y, a)$ follows the update rule in* (16) *and the sample size $N_k \to \infty$ as $k \to \infty$. Then recursively applying* (15) *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 D. □

VI. Experiments

In this section, we provide some numerical examples to illustrate the algorithms developed in this paper.

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 [11], 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 [11]. 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 [25], [26]. Furthermore, considering the cases where the transition probability is unknown, we also validate the algorithm equipped with SAA (Algorithm 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 and Algorithm 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 1, 2 and 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 in Figures 1, 2 and 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 part, we also reproduce the CVaR algorithm in [11] 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 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 despite some minor difference that can be further alleviated by choosing larger N_k.

In the second experiment, we apply both Algorithm 2 and Algorithm 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

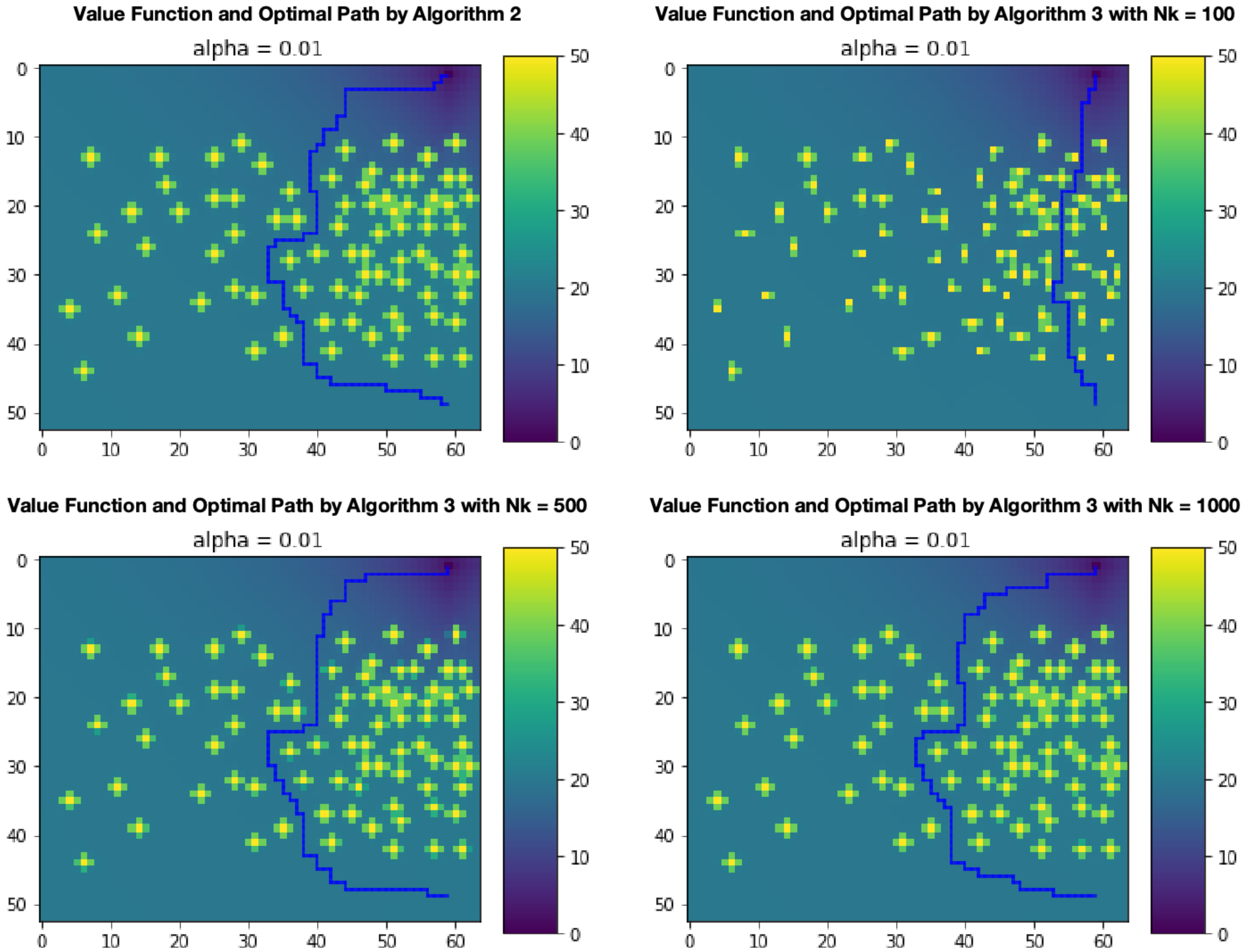


Fig. 1. The value function and corresponding optimal path for $\alpha = 0.01$ generated by Algorithm 2 and Algorithm 3 (with different values of N_k) in the obstacle's setting.

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 4, 5, 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 is exactly the same as the one we get in the first experiment. Moreover, for the results generated by Algorithm 3, when $N_k = 100$, all these optimal policies generated by Algorithm 3 are quite different with these in Algorithm 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 $\alpha = 0.01$.

VII. Conclusion

In this paper, we have applied EVaR to risk-sensitive reinforcement learning. We have proposed an EVaR value iteration algorithm based on Markov decision process and a more practical approximate version. Moreover, we have showed the convergence for these value iteration algorithms and bounded the approximation error. Furthermore, for the cases where the transition kernel of the underlying MDP is unknown, we have presented a sample-based EVaR synchronous Q-value update algorithm along with its convergence analysis. We have validated our approaches in simulation experiments.

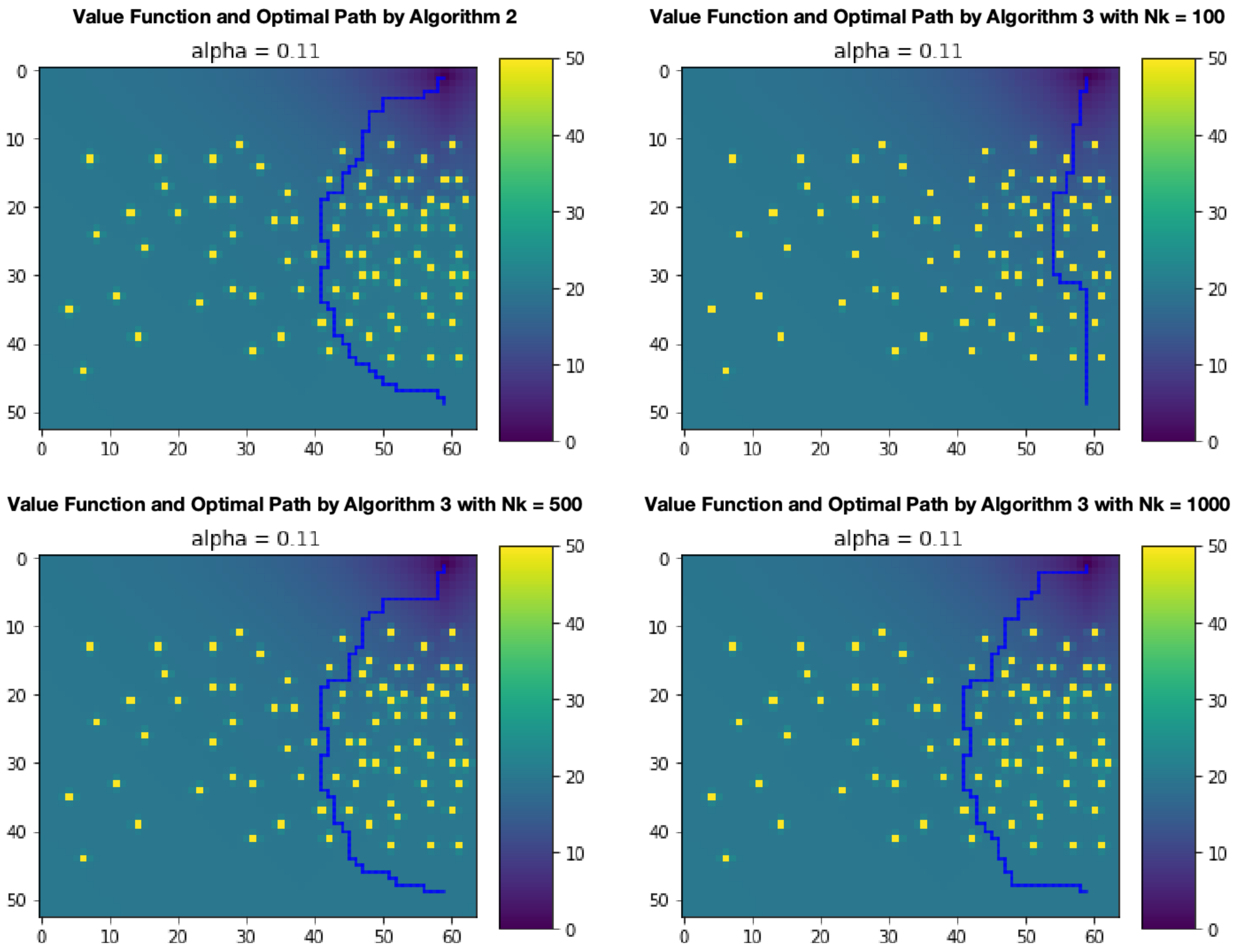


Fig. 2. The value function and corresponding optimal path for $\alpha = 0.11$ generated by Algorithm 2 and Algorithm 3 (with different values of N_k) in the obstacle's setting.

In terms of future work, it is important to analyze the the sample complexity, i.e., to analyze the scaling of the sample size N_k as the number of states and actions, and approximation error tolerance level etc. Furthermore, the algorithms developed in this paper are all offline algorithms, it is important to develop online algorithms for EVaR optimization. Finally, as the algorithms in the paper are value function based algorithms, it is interesting to develop policy gradient type algorithms.

References

[1] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*, MIT press, Cambridge, MA, 2018.

[2] D. Bertsekas, *Dynamic programming and optimal control: Volume I*, vol. 1, Athena scientific, Nashua, NA, 2012.

[3] S. Mannor, D. Simester, P. Sun, and J. N. Tsitsiklis, "Bias and variance approximation in value function estimates," *Management Science*, vol. 53, no. 2, pp. 308–322, Feb. 2007.

[4] R. A. Howard and J. E. Matheson, "Risk-sensitive Markov decision processes," *Management Science*, vol. 18, no. 7, pp. 356–369, Mar. 1972.

[5] P. Artzner, F. Delbaen, J. Eber, and D. Heath, "Coherent measures of risk," *Mathematical Finance*, vol. 9, no. 3, pp. 203–228, Jul. 1999.

[6] H. Föllmer and A. Schied, "Convex measures of risk and trading constraints," *Finance and Stochastics*, vol. 6, no. 4, pp. 429–447, Oct. 2002.

[7] A. Nilim and L. El Ghaoui, "Robust control of Markov decision processes with uncertain transition matrices," *Operations Research*, vol. 53, no. 5, pp. 780–798, Oct. 2005.

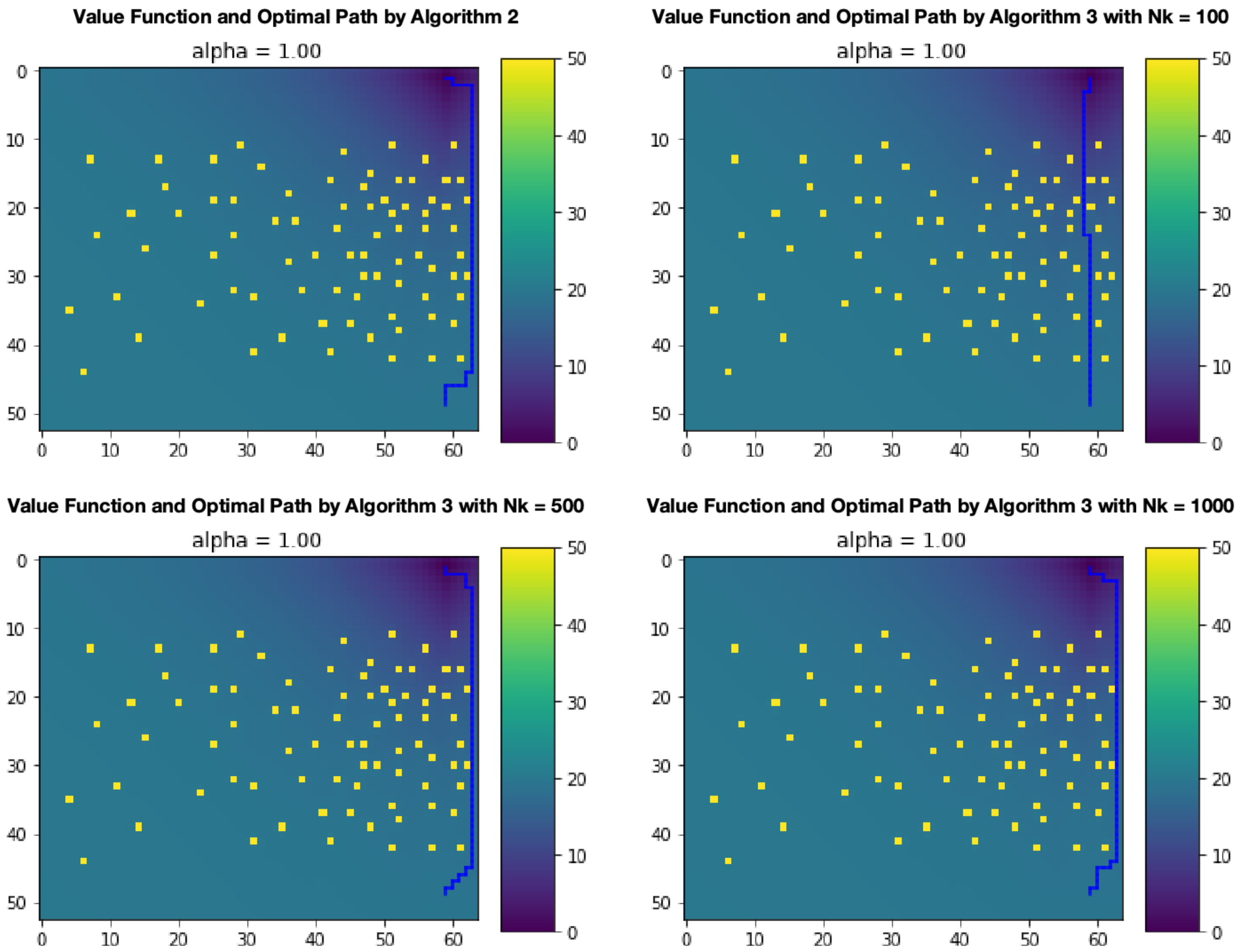


Fig. 3. The value function and corresponding optimal path for $\alpha = 1.00$ generated by Algorithm 2 and Algorithm 3 (with different values of N_k) in the obstacle's setting.

[8] T. Osogami, "Robustness and risk-sensitivity in Markov decision processes," in *Proc. Advances in Neural Information Processing Systems*, Lake Tahoe, NV, Dec. 2012, vol. 25, pp. 233–241.

[9] R. T. Rockafellar and S. Uryasev, "Optimization of conditional value-at-risk," *Journal of Risk*, vol. 2, pp. 21–42, Apr. 2000.

[10] R. T. Rockafellar and S. Uryasev, "Conditional value-at-risk for general loss distributions," *Journal of Banking & Finance*, vol. 26, no. 7, pp. 1443–1471, Jul. 2002.

[11] Y. Chow, A. Tamar, S. Mannor, and M. Pavone, "Risk-sensitive and robust decision-making: a CVaR optimization approach," *arXiv preprint arXiv:1506.02188*, 2015.

[12] A. Tamar, Y. Glassner, and S. Mannor, "Optimizing the CVaR via sampling," in *Proc. AAAI Conference on Artificial Intelligence*, Austin, TX, Feb. 2015, pp. 2993–2999.

[13] Y. Chow, M. Ghavamzadeh, L. Janson, and M. Pavone, "Risk-constrained reinforcement learning with percentile risk criteria," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 6070–6120, Jan. 2017.

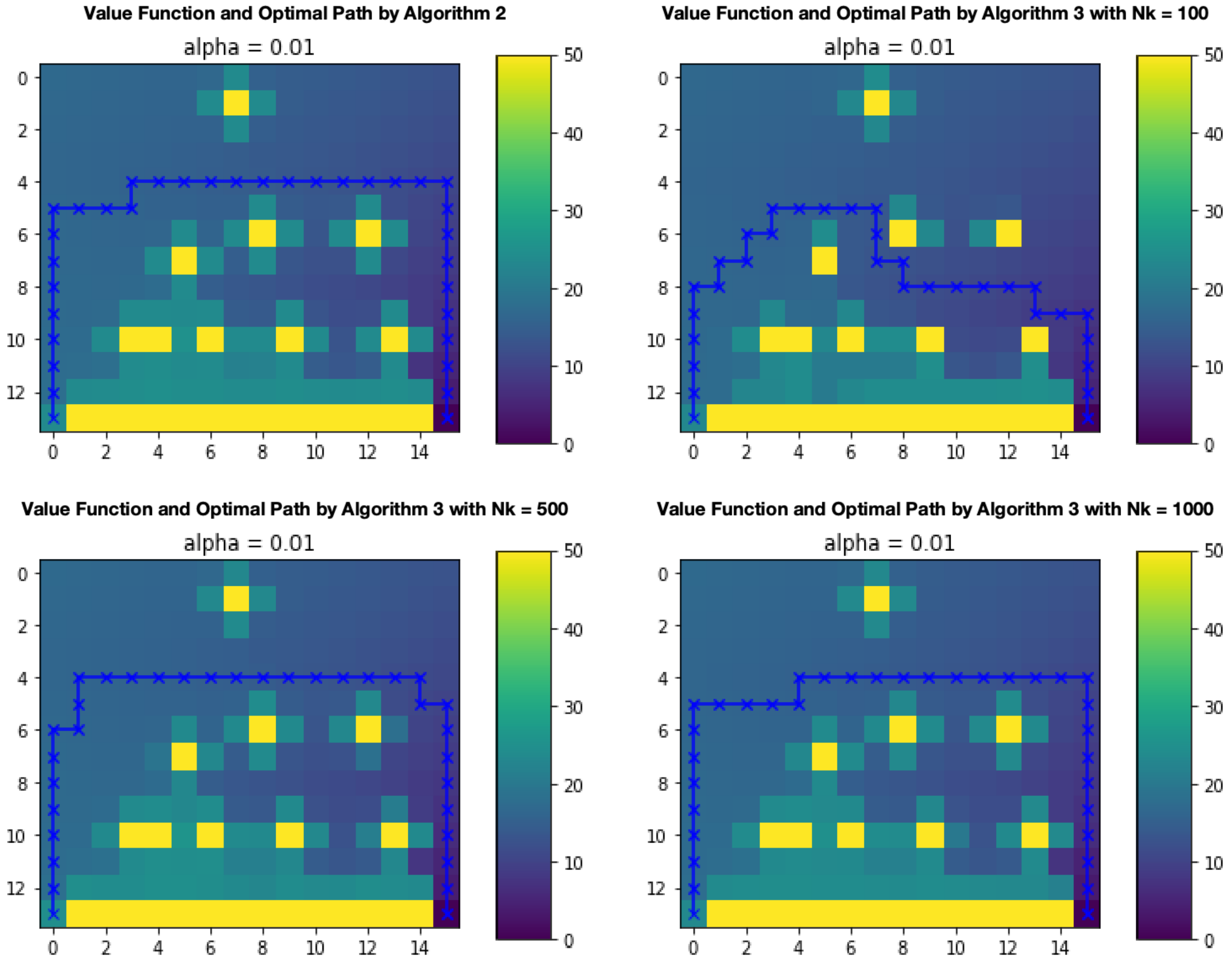
[14] A. Tamar, Y. Glassner, and S. Mannor, "Policy gradients beyond expectations: Conditional value-at-risk," *arXiv preprint arXiv:1404.3862*, 2014.

[15] H. Kashima, "Risk-sensitive learning via minimization of empirical conditional value-at-risk," *IEICE Transactions on Information and Systems*, vol. 90, no. 12, pp. 2043–2052, Dec. 2007.

[16] 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.

[17] 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*, vol. 12, no. 1, pp. 1–6, Feb. 2021.

[18] R. Singh, Q. Zhang, and Y. Chen, "Improving robustness via risk averse distributional reinforcement learning," in *Proc. Learning for Dynamics and Control*, Berkeley, CA, Jul. 2020, pp. 958–968.



Value Function and Optimal Path by Algorithm 2

Value Function and Optimal Path by Algorithm 3 with Nk = 100

Value Function and Optimal Path by Algorithm 3 with Nk = 500

Value Function and Optimal Path by Algorithm 3 with Nk = 1000

Fig. 4. The value function and corresponding optimal path for $\alpha = 0.01$ generated by Algorithm 2 and Algorithm 3 (with different values of N_k) in the cliff's setting.

[19] A. Ahmadi-Javid, "Entropic value-at-risk: A new coherent risk measure," *Journal of Optimization Theory and Applications*, vol. 155, no. 3, pp. 1105–1123, Dec. 2012.

[20] G. C. Pflug and A. Pichler, "Time-consistent decisions and temporal decomposition of coherent risk functionals," *Mathematics of Operations Research*, vol. 41, no. 2, pp. 682–699, May. 2016.

[21] A. Shapiro, D. Dentcheva, and A. Ruszczynski, *Lectures on stochastic programming: modeling and theory*, SIAM, Philadelphia, PA, 2021.

[22] A. Tamar, S. Mannor, and H. Xu, "Scaling up robust MDPs using function approximation," in *Proc. International conference on machine learning*, Beijing, China, Jul. 2014, pp. 181–189.

[23] M. Ang, J. Sun, and Q. Yao, "On the dual representation of coherent risk measures," *Annals of Operations Research*, vol. 262, no. 1, pp. 29–46, Mar. 2018.

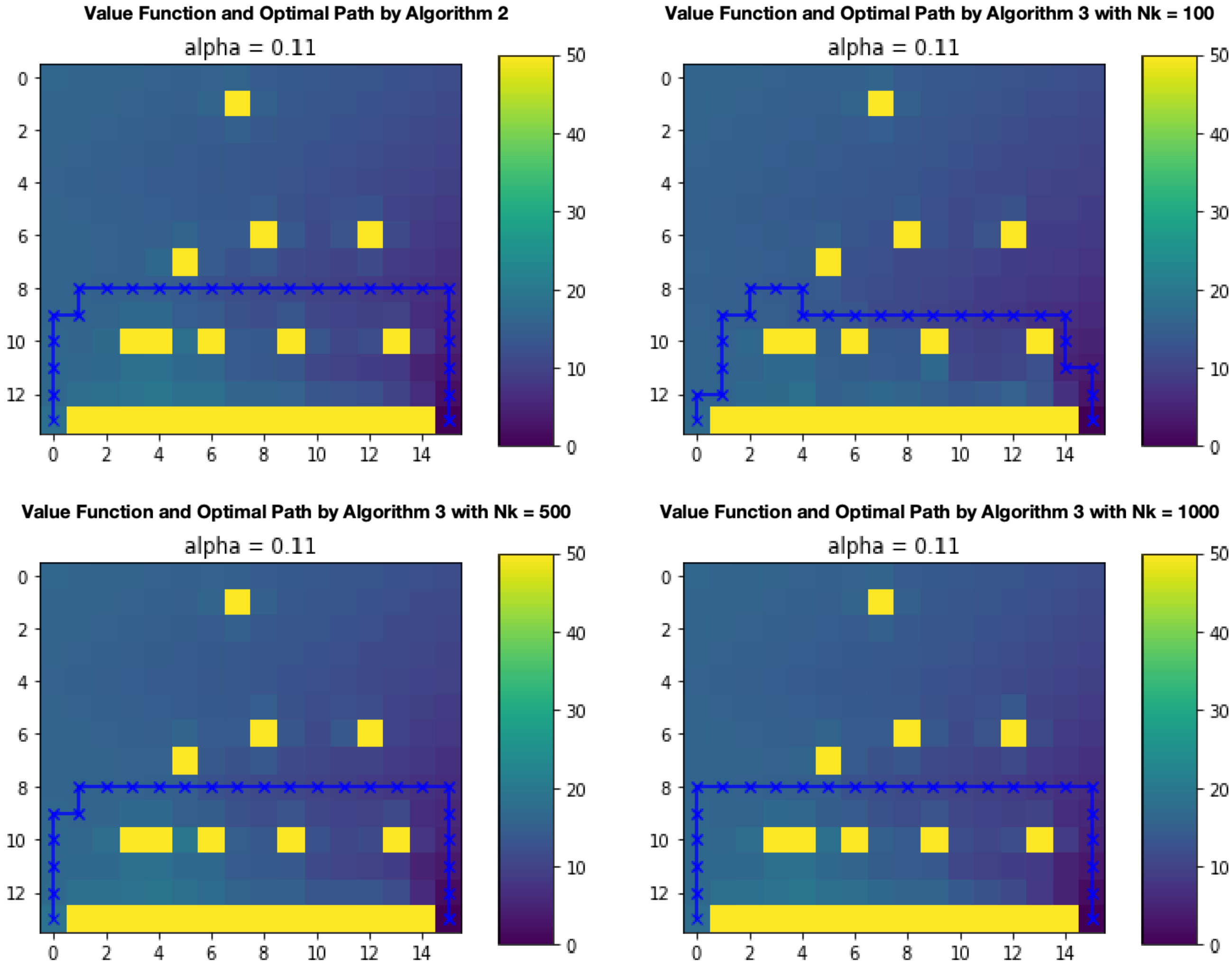
[24] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski, *Robust optimization*, Princeton University Press, Princeton, NJ, 2009.

[25] Bob Bixby, "The gurobi optimizer," *Transp. Re-search Part B*, vol. 41, no. 2, pp. 159–178, 2007.

[26] I Gurobi Optimization et al., "Gurobi optimizer reference manual, 2018," *URL http://www. gurobi. com*, 2018.

Appendix A
Proof of Lemma 1

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


Value Function and Optimal Path by Algorithm 2

Value Function and Optimal Path by Algorithm 3 with Nk = 100

Value Function and Optimal Path by Algorithm 3 with Nk = 500

Value Function and Optimal Path by Algorithm 3 with Nk = 1000

Fig. 5. The value function and corresponding optimal path for $\alpha = 0.11$ generated by Algorithm 2 and Algorithm 3 (with different values of N_k) in the cliff's setting.

EVaR Bellman operator. 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

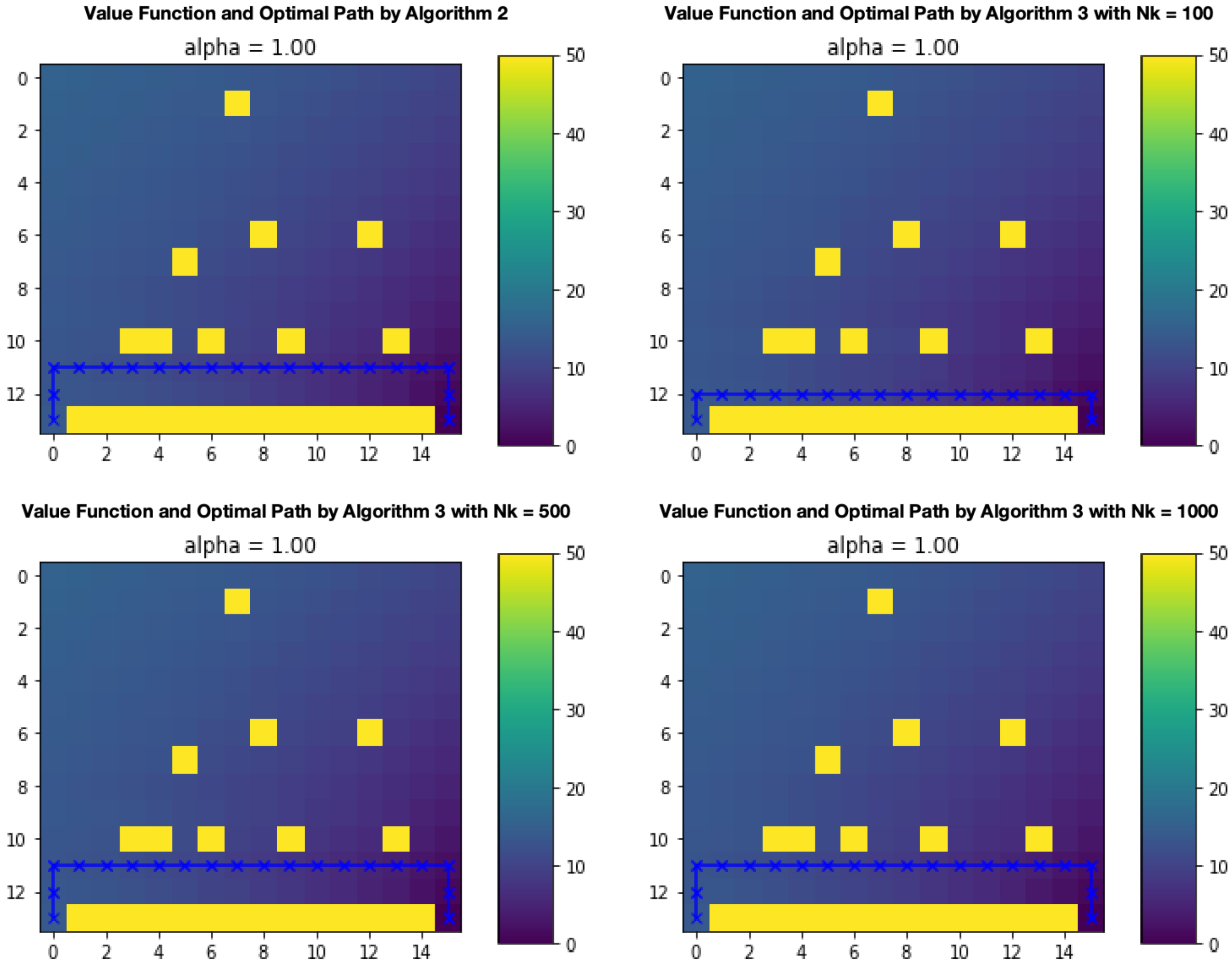


Fig. 6. The value function and corresponding optimal path for $\alpha = 1.00$ generated by Algorithm 2 and Algorithm 3 (with different values of N_k) in the cliff's setting.

$y_1, y_2 \in \mathcal{Y}$ and $\lambda \in [0,1]$ and define $y_\lambda = (1-\lambda)y_1 + \lambda y_2$. Then,

$$\begin{aligned}
&(1-\lambda)y_1\mathbf{T}[V](x,y_1)+\lambda y_2\mathbf{T}[V](x,y_2)\\
&=(1-\lambda)y_1\min_{a_1\in\mathcal{A}}\Big[C(x,a_1)+\gamma\max_{\xi_1\in\mathcal{U}_{\text{EVaR}}(y_1,P(\cdot|x,a_1))}\sum_{x'\in\mathcal{X}}\xi_1(x')V(x',y_1\xi(x'))P(x'|x,a_1)\Big]\\
&\qquad+\lambda y_2\min_{a_2\in\mathcal{A}}\Big[C(x,a_2)+\gamma\max_{\xi_2\in\mathcal{U}_{\text{EVaR}}(y_2,P(\cdot|x,a_2))}\sum_{x'\in\mathcal{X}}\xi_2(x')V(x',y_2\xi(x'))P(x'|x,a_2)\Big]\\
&\overset{(1)}{\le}\min_{a\in\mathcal{A}}\Big[y_\lambda C(x,a)+\gamma\max_{\substack{\xi_1\in\mathcal{U}_{\text{EVaR}}(y_1,P(\cdot|x,a))\\ \xi_2\in\mathcal{U}_{\text{EVaR}}(y_2,P(\cdot|x,a))}}\sum_{x'\in\mathcal{X}}P(x'|x,a)\big((1-\lambda)y_1\xi_1(x')V(x',y_1\xi_1(x'))\\
&\qquad+\lambda y_2\xi_2(x')V(x',y_2\xi_2(x'))\big)\Big]\\
&\overset{(2)}{\le}\min_{a\in\mathcal{A}}\Big[y_\lambda C(x,a)+\gamma\max_{\substack{\xi_1\in\mathcal{U}_{\text{EVaR}}(y_1,P(\cdot|x,a))\\ \xi_2\in\mathcal{U}_{\text{EVaR}}(y_2,P(\cdot|x,a))}}\sum_{x'\in\mathcal{X}}P(x'|x,a)\big((1-\lambda)y_1\xi_1(x')+\lambda y_2\xi_2(x')\big)\\
&\qquad V(x,(1-\lambda)y_1\xi_1(x')+\lambda y_2\xi_2(x')\big)\Big].
\end{aligned}\tag{18}$$

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

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

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

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

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

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

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

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

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

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

Combining these two inequalities, we can get

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

i.e,

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

Thus, we have shown that ξ also belongs to $\mathcal{U}_{\text{EVaR}}(y_\lambda, P(\cdot|x, a))$. Then, combining this fact with (18), we obtain

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

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 (8) 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.

Appendix B
Proof of Theorem 2

The proof of Theorem 2 follows the idea in the proof of Theorem 4 in [11].

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 $(x, y) \in \mathcal{X} \times \mathcal{Y}$,

$$\begin{aligned} V_n(x, y) &:= \mathbf{T}^n[V_0](x, y) \\ &= \min_{\mu \in \Pi_M} \text{EVaR}_y\big(\mathcal{C}_{0,n} + \gamma^n V_0(x_n, y_n)|x_0 = x, \mu\big), \end{aligned} \tag{19}$$

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

By induction hypothesis, firstly we need to verify (19) holds when $n = 1$. For $n = 1$, let (x_1, y_1) denotes $(x', y\xi(x'))$, from definition we have

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

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

Next, we assume that (19) holds at $n = k$.

Then for $n = k+1$,

$$\begin{aligned}
V_{k+1}(x,y) &:= \mathbf{T}^{k+1}[V_0](x,y) = \mathbf{T}[V_k](x,y) \\
&\overset{(1)}{=} \min_{a\in\mathcal{A}} \left[C(x,a) + \gamma \max_{\xi\in\mathcal{U}_{\text{EVaR}}(y,P(\cdot|x,a))} \sum_{x'\in\mathcal{X}} \xi(x') V_k(x', y\xi(x')) P(x'|x,a) \right] \\
&\overset{(2)}{=} \min_{a\in\mathcal{A}} \left[C(x,a) + \gamma \max_{\xi\in\mathcal{U}_{\text{EVaR}}(y,P(\cdot|x,a))} \sum_{x'\in\mathcal{X}} \xi(x') P(x'|x,a) \min_{\mu\in\Pi_M} \text{EVaR}_{y\xi(x')}(\mathcal{C}_{0,k} + \gamma^k V_0 | x_0 = x', \mu) \right] \\
&\overset{(3)}{=} \min_{a\in\mathcal{A}} [C(x,a) + \gamma \max_{\xi\in\mathcal{U}_{\text{EVaR}}(y,P(\cdot|x,a))} \sum_{x'\in\mathcal{X}} \xi(x') P(x'|x,a) \min_{\mu\in\Pi_M} \text{EVaR}_{y_1}(\mathcal{C}_{0,k} + \gamma^k V_0 | x_0 = x_1, \mu)] \\
&= \min_{a\in\mathcal{A}} \left[C(x,a) + \max_{\xi\in\mathcal{U}_{\text{EVaR}}(y,P(\cdot|x,a))} \sum_{x'\in\mathcal{X}} \xi(x') P(x'|x,a) \min_{\mu\in\Pi_M} \text{EVaR}_{y_1}(\gamma\mathcal{C}_{0,k} + \gamma^{k+1} V_0 | x_0 = x_1, \mu) \right] \\
&\overset{(4)}{=} \min_{a\in\mathcal{A}} \left[C(x,a) + \max_{\xi\in\mathcal{U}_{\text{EVaR}}(y,P(\cdot|x,a))} \mathbb{E}_{\xi P} \big[\min_{\mu\in\Pi_M} \text{EVaR}_{y_1}(\mathcal{C}_{1,k+1} + \gamma^{k+1} V_0 | x_1, \mu) \big] \right] \\
&\overset{(5)}{=} \min_{a\in\mathcal{A}} \big[\min_{\mu\in\Pi_M} \text{EVaR}_y(\mathcal{C}_{0,k+1} + \gamma^{k+1} V_0 | x_0 = x, \mu) \big] \\
&= \min_{\mu\in\Pi_M} \text{EVaR}_y(\mathcal{C}_{0,k+1} + \gamma^{k+1} V_0 | x_0 = x, \mu),
\end{aligned} \tag{20}$$

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 (19) 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,

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

and (5) is by the EVaR decomposition theorem. Thus, (20) is proved by induction.

The second part of the proof is to show that

$$V^*(x_0, y_0) = \min_{\mu\in\Pi_M} \text{EVaR}_{y_0} \left(\lim_{n\to\infty} \mathcal{C}_{0,n} | x_0, \mu \right). \tag{21}$$

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 [2]. The third equation is derived from the definition of V_n. Combining the above results, we have

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

The second equality is due to the boundedness of both state-wise cost and V_0. Recall the subadditivity property of EVaR, we obtain

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

which implies

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

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

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

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

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

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

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

where the tail policy sequence is equal to $\mu = \{\mu_t, \mu_{t+1}, \dots\}$ and the action is given by $a_j = \mu_j(h_j)$ for $j \geq t$.

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

Let μ^* denote the optimal policy of the t^{th}-subproblem mentioned above, then the policy $\tilde{\mu} = \{\mu^*_{t+1}, \mu^*_{t+2}, \dots\}$ is a feasible policy for the $(t+1)^{th}$-subproblem for any state x_{t+1} and confidence level y_{t+1}:

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

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

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

where (1) is by the decomposition theorem, (2) is by defining $\mathbb{V}^{\tilde{\mu}}(x_t, y_t) = \text{EVaR}_{y_t}(\lim_{n\to\infty} \mathcal{C}_{t,n}|x_t, \tilde{\mu})$, (3) is by $\mathbb{V}^{\tilde{\mu}}(x, y) \geq \mathbb{V}(x, y)$ 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 $\mu^* = \{\mu^*_{t+1}, \mu^*_{t+2}, \dots\} \in \Pi_H$ 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 $\tilde{\mu} = \{\tilde{\mu}_t, \tilde{\mu}_{t+1}, \dots\} \in \Pi_H$ for the t^{th} subproblem from μ^* by $\tilde{\mu}_t(x_t) = u^*(x_t, y_t)$ and $\tilde{\mu}_j(h_j) = \mu^*_j(h_j)$, 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

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

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

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

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

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

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

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

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

Appendix C
Proof of theorem 3

The proof of Theorem 3 follow the similar idea with the proof of Theorem 5 in [11].

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 $\mathcal{X} \times \mathcal{Y}$

$$\mathbf{T}[V^*](x, y) = \mathbf{T}_{u^*}[V^*](x, y), \tag{22}$$

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 (22) holds. From Theorem 2, we know that

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

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 (22) 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

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

APPENDIX D
PROOF OF THEOREM 5

The proof of Theorem 4 follows the idea of the proof of Theorem 7 in [11].

We can rewrite the (15) as

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

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 [21]) and for any $k \in \mathbb{N}$, let

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

We can rewrite the noise term as

$$\begin{aligned} M_k(x,y,a) &= T_1 - T_2 \\ &\leq |T_1 - T_2|. \end{aligned}$$

Then

$$\begin{aligned} M_k^2(x,y,a) &\leq |T_1 - T_2|^2 \\ &\leq |T_1|^2 + |T_2|^2 \\ &\leq 2 \max_{x,y,a} Q_k^2(x,y,a). \end{aligned}$$

Then the assumptions in Proposition 4.5 in [2] 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 $V_1(x,y) \geq V_2(x,y), \forall x \in \mathcal{X}, y \in \mathcal{Y}$, 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

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

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 [2] 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 $\mathbf{F}_{\mathcal{I}}[Q](x,y,a) = Q(x,y,a)$.