Lexicographic Optimisation of Conditional Value at Risk and Expected Value for Risk-Averse Planning in MDPs

Marc Rigter, Paul Duckworth, Bruno Lacerda, Nick Hawes,
Oxford Robotics Institute, University of Oxford, United Kingdom
{mrigter, pduckworth, bruno, nickh}@robots.ox.ac.uk

Abstract
Planning in Markov decision processes (MDPs) typically optimises the expected cost. However, optimising the expectation does not consider the risk that for any given run of the MDP, the total cost received may be unacceptably high. An alternative approach is to find a policy which optimises a risk-averse objective such as conditional value at risk (CVaR). In this work, we begin by showing that there can be multiple policies which obtain the optimal CVaR. We formulate the lexicographic optimisation problem of minimising the expected cost subject to the constraint that the CVaR of the total cost is optimal. We present an algorithm for this problem and evaluate our approach on three domains, including a road navigation domain based on real traffic data. Our experimental results demonstrate that our lexicographic approach attains improved expected cost while maintaining the optimal CVaR.

Introduction
Markov decision processes (MDPs) are a common framework for sequential decision making under uncertainty, and have been applied to many domains such as planning for vehicle navigation (Nardi and Stachniss 2019) and mobile service robots (Lacerda et al. 2019). The solution for an MDP typically optimises the expected total cost. However, for any single run of the MDP, the total cost received is uncertain due to the MDP’s stochastic transitions. In some applications we wish to compute risk-averse policies which prioritise avoiding the worst outcomes, rather than simply minimising the expected total cost irrespective of the variability.

For example, consider a route-finding system for an automated taxi service. The system attempts to minimise travel duration under the uncertainty of unknown traffic conditions. A direct route, i.e. along the main roads through the centre of the city, optimises the expected journey duration. However, occasionally this route yields extremely long journey durations due to traffic jams. These rare outcomes will result in unhappy customers for the taxi service. On the other hand, a risk-averse route through quiet side streets is slower in expectation but avoids the risk of severe traffic jams, and thus unhappy customers.

In this work, we focus on conditional value at risk (CVaR), a common coherent risk metric (Rockafellar, Uryasev et al. 2000), in the static risk setting. In MDPs, the static CVaR at confidence level \( \alpha \) corresponds to the mean total cost in the worst \( 100\alpha \% \) of runs. The objectives of CVaR and expected cost are in conflict. A policy for expected cost can incur arbitrarily bad outcomes, provided that the expectation is minimised. On the other hand, a policy for CVaR only optimises the worst \( 100\alpha \% \) of runs.

To address these conflicting objectives, we first show that there can be multiple policies which obtain the optimal CVaR. This motivates us to propose a lexicographic approach which optimises the expected total cost subject to the constraint that the CVaR of the total cost is optimal. During execution, the resulting policy is initially risk-averse. However, the policy may begin taking more aggressive actions to improve the expected cost, provided that there is no longer any risk of incurring a bad run which would influence the CVaR. Our lexicographic formulation is suitable for domains where the first priority is to avoid risks, but it is desirable to optimise expected value as a secondary objective.

The main contributions of this paper are: 1) proposing and formalising the lexicographic problem of optimising expected value in MDPs subject to the constraint of minimising CVaR, and 2) an algorithm to solve this problem, based on reducing it to a two-stage optimisation in a stochastic game. To the best of our knowledge, this is the first work to address lexicographic optimisation of CVaR and expected value in sequential decision making problems.

We evaluate our algorithm on three domains, including a...
road network navigation domain which uses real data from traffic sensors (Chen et al. 2001) and Google Maps to simulate journey times (see Figure 1). Our experimental results demonstrate that our approach significantly improves the expected cost on all domains while attaining the optimal conditional value at risk.

Related Work

Many existing works address risk-averse optimisation in MDPs. Early work on this subject considered exponential utility (Howard and Matheson 1972), however this risk metric is difficult to interpret. Another line of work considers the mean-variance criterion (Sobel 1982), but this metric has undesirable properties as it is not a coherent risk metric. See Artzner et al. (1999) or Majumdar and Pavone (2020) for an introduction to coherent risk metrics.

Conditional value at risk (CVaR) (Rockafellar, Uryasev et al. 2000) is a common coherent risk metric which has been applied to MDPs. In this paper, we consider static risk, where the risk metric is applied to the total cost, rather than dynamic risk which penalises risk at each time step (Ruszczynski 2010). For this static CVaR setting, approaches based on value iteration over an augmented state space have been proposed (Bauerle and Ott 2011; Chow et al. 2015). Other works propose policy gradient methods to find a locally optimal solution for CVaR (Borkar and Jain 2010; Chow and Ghavamzadeh 2014; Tamar et al. 2015). Prashanth et al. (2017) present an RL algorithm based on optimism in the face of uncertainty, while Rigter, Lacerda, and Hawes (2021) propose an approach based on Monte Carlo tree search and also consider model uncertainty.

Existing works have considered trading-off the conflicting objectives of expected cost and risk. Petrik and Subramanian (2012) optimise a weighted combination of dynamic CVaR objectives of expected cost and risk. Petrik and Subramanian also consider model uncertainty.

Lexicographic approaches to multi-objective decision making in MDPs have been proposed (Mouaddib 2004; Wray, Zilberstein, and Mouaddib 2015; Lacerda, Parker, and Hawes 2015). These approaches optimise the expected value for each objective in a lexicographic ordering. Unlike these approaches, we consider risk-averse planning. To the best of our knowledge, this is the first work to propose a lexicographic approach to the optimisation of CVaR and expected value in sequential decision making problems.

Preliminaries

Conditional Value at Risk

Let $Z$ be a bounded-mean random variable, i.e. $E|Z| < \infty$, on a probability space $(\Omega, \mathcal{F}, \mathcal{P})$, with cumulative distribution function $F(z) = \mathcal{P}(Z \leq z)$. In this paper, we interpret $Z$ as the total cost which is to be minimised. The value at risk (VaR) at confidence level $\alpha \in (0, 1]$ is defined as $\text{VaR}_\alpha(Z) = \min \{z | F(z) \geq 1 - \alpha\}$. The conditional value at risk at confidence level $\alpha$ is defined as

$$
\text{CVaR}_\alpha(Z) = \frac{1}{\alpha} \int_{-\infty}^{\text{VaR}_1-\gamma(Z)} \mathcal{P}(\gamma)d\gamma.
$$

If $Z$ has a continuous distribution, $\text{CVaR}_\alpha(Z)$ can be defined using the more intuitive expression: $\text{CVaR}_\alpha(Z) = E[Z | Z \geq \text{VaR}_\alpha(Z)]$. Thus, $\text{CVaR}_\alpha(Z)$ may be interpreted as the expected value of the $\alpha$-portion of the right tail of the distribution of $Z$. CVaR may also be defined as the expected value under a perturbed distribution using its dual representation (Rockafellar, Uryasev et al. 2000; Shapiro, Dentcheva, and Ruszczyński 2014):

$$
\text{CVaR}_\alpha(Z) = \max_{\xi \in \mathcal{B}(\alpha, \mathcal{P})} E_{\xi}[Z],
$$

where $E_{\xi}[Z]$ denotes the $\xi$-weighted expectation of $Z$, and the risk envelope, $\mathcal{B}$, is given by

$$
\mathcal{B}(\alpha, \mathcal{P}) = \left\{ \xi \mid \xi(\omega) \in [0, \frac{1}{\alpha}], \int_{\omega \in \Omega} \xi(\omega)\mathcal{P}(\omega)d\omega = 1 \right\}.
$$

where $\mathcal{P}(\omega)$ is the probability density function if $Z$ is continuous, and the probability mass function if $Z$ is discrete.

Therefore, the CVaR of a random variable $Z$ may be interpreted as the expectation of $Z$ under a worst-case perturbed distribution, $\xi \mathcal{P}$. The risk envelope is defined so that the probability of any outcome can be increased by a factor of at most $1/\alpha$, whilst ensuring the perturbed distribution is a valid probability distribution.

Markov Decision Processes

A stochastic shortest path (SSP) Markov decision process (MDP) is a tuple, $\mathcal{M} = (S, A, C, T, G, h_0)$, where $S$ and $A$ are finite state and action spaces; $C : S \times A \rightarrow \mathbb{R}_+$ is the cost function; $T : S \times A \times S \rightarrow [0, 1]$ is the probabilistic transition function; $G \subset S$ is the set of absorbing goal states, from which the model incurs zero cost; and $h_0$ is the initial state. A history of $\mathcal{M}$ is a sequence $h = s_0a_0s_1a_1 \ldots$ such that $T(s_i, a_i, s_i+1) > 0$ for all $i \in \{0, \ldots, |h|\}$, where $|h|$ denotes the length of $h$. We denote the set of all finite-length
histories over $\mathcal{M}$ as $\mathcal{H}_M^*$, and the set of all infinite-length histories over $\mathcal{M}$ as $\mathcal{H}_{iM}^*$, and define the set of all histories over $\mathcal{M}$ as $\mathcal{H}^M = \mathcal{H}_M^* \cup \mathcal{H}_{iM}^*$. The cumulative cost function $\text{cumul}_C : \mathcal{H}^M \to \mathbb{R}^+$ is defined such that, given a history $h = s_0a_0s_1a_1 \ldots$, $\text{cumul}_C(h) = \sum_{i=0}^{\infty} C(s_i, a_i)$. A history-dependent policy is a function $\pi : \mathcal{H}_{iM}^* \to A$, and we write $\Pi^M$ to denote the set of all history-dependent policies for $\mathcal{M}$. If $\pi$ only depends on the last state $s_t$ of $h$, then we say $\pi$ is Markovian, and we denote the set of all Markovian policies as $\Pi^M$. A policy $\pi$ induces a probability distribution $\mathcal{P}^\pi_M$ over $\mathcal{H}_{iM}^*$ and we define the cumulative cost distribution $C^\pi_M$ as the distribution over the value of $\text{cumul}_C$ for infinite-length histories of $\mathcal{M}$ under policy $\pi$. A policy is proper if it is proper at all states. In an SSP, a proper policy incurs infinite cost at all states where it is improper. These assumptions ensure the expected value of the cumulative cost distribution is finite for at least one policy in the SSP.

**Stochastic Games**

In this paper, we formulate CVaR optimisation as a turn-based two-player zero-sum stochastic shortest path game (SSPG). An SSPG between an agent and an adversary is a generalisation of an SSP MDP and can be defined using a $(SSPG)$. An SSPG is a two-player zero-sum stochastic shortest path game. In this paper, we formulate CVaR optimisation as a turn-based two-player zero-sum stochastic shortest path game.

1. A policy is proper if it is proper at all states. In an SSP, a proper policy incurs infinite cost at all states where it is improper. These assumptions ensure the expected value of the cumulative cost distribution is finite for at least one policy in the SSP.

**CVaR Optimisation in MDPs**

Many existing works have addressed the problem of optimising the CVaR of $C^\pi_{\alpha}$, defined as follows.

**Problem 1.** Let $\mathcal{M}$ be an MDP. Find the optimal CVaR of the cumulative cost at confidence level $\alpha$:

$$\min_{\pi \in \Pi^\mathcal{M}} \text{CVaR}_\alpha(C^\mathcal{M}_{\pi}).$$

Note that the optimal policy for Problem 1 may be history-dependent (Bäuerle and Ott 2011). Methods based on dynamic programming have been proposed to solve Problem 1 by Bäuerle and Ott (2011) and Chow et al. (2015). Pflug and Pichler (2016). In particular, Chow et al. (2015) formulate Problem 1 as the expected value in an SSPG against an adversary which modifies the transition probabilities. This formulation is based on the CVaR representation theorem in Eq. 2 where CVaR may be represented as the expected value under a perturbed probability distribution. The SSPG is defined so that the ability for the adversary to perturb the transition probabilities corresponds to the risk envelope given in Eq. 3.

Formally, the CVaR SSPG is defined by the tuple $G = (S^+, A^+, C^+, T^+, G^+, s_0^+)$. The state space $S^+ = S \times [0, 1] \times (A \cup \{\bot\})$ is the original MDP state space augmented with a continuous state factor, $y \in [0, 1]$, representing the "budget" of the adversary to perturb the probabilities; and a state factor $a \in A \cup \{\bot\}$ indicating the most recent agent action if it is the adversary’s turn to choose an action, and $\bot$ if it is the agent’s turn. The action space is defined as $A^+ = A \cup \Xi$, i.e. the agent actions are the actions in the original MDP, and the adversary actions are a set $\Xi$ that will be defined next.

The SSPG transition dynamics are as follows. Given an agent state, $(s, y, \bot) \in S^\text{agt}$, the agent applies an action, $a \in A$, and receives cost $C^+(s, y, \bot, a) = C(s, a)$. The state then transitions to the adversary state $(s, a, s') \in S^\text{adv}$. The adversary then chooses an action to perturb the original MDP transition probabilities from a continuous action space defined as:

$$\Xi(s, a, s') = \{\xi \in \mathbb{R}^{|S|} \mid 0 \leq \xi(s') \leq \eta \ \forall s'$$

$$\text{and} \sum_{s' \in S} \xi(s') = 1\}.$$

where $\eta = \infty$ if $y = 0$, and $\eta = 1/y$ otherwise; $T$ is the original MDP transition function. Eq. 6 restricts the adversary actions so the probability of any history is increased by at most $1/y$, and the perturbed transition probabilities remain a valid probability distribution. After the adversary chooses the perturbation action $\xi \in \Xi(s, a, s')$, the game transitions back to an agent state $(s', y, \xi(s'), \bot) \in S^\text{agt}$ according to the following transition function where the probability of each successor $s'$ in the original MDP is perturbed by the factor $\xi(s')$:

$$T^+(s', y, a, \xi(s'), \xi(s'), \bot) = \xi(s')T(s', a, s').$$

Finally, we define the initial augmented state as $s_0^+ = (s_0, \alpha, \bot) \in S^+_\text{agt}$, where the state variable is set to the CVaR confidence level $\alpha$. We also define $G^+$ as the set of goal states on the augmented state space corresponding to goal states in the original MDP.

Chow et al. (2015) showed that the minimax expected value equilibrium for $G^+$ corresponds to the optimal CVaR.

**Proposition 1.** (Chow et al. 2015) Let $G^+$ be a CVaR SSPG corresponding to original MDP $\mathcal{M}$. Then:

$$\min_{\pi \in \Pi^\mathcal{M}} \text{CVaR}_\alpha(C^\mathcal{M}_{\pi}) = \min_{\pi \in \Pi^\mathcal{M}} \max_{\sigma \in \Sigma^\mathcal{M}} \mathbb{E}[C^\mathcal{G}^+_{\pi, \sigma}].$$
Proposition 1 holds because the state variable keeps track of the total multiplicative perturbation to the probability of any history. Thus, the constraints in Eq. 6 ensure that the maximum perturbation to the probability of any history from the initial state is 1/α, and that the perturbed distribution over histories is a valid probability distribution. Therefore, the admissible adversarial perturbation actions in G* correspond to the risk envelope in Eq. 5. According to Eq. 2, the CVaR is the expected value under the perturbations in the risk-envelope that maximise the expected cost.

To compute the solution to Eq. 8, we denote the value function for the augmented state by $V_{CV}(s,y,\bot) = \min_{\pi \in \Pi_M^\alpha} \text{CVaR} \left( C(\pi) \right)$. This value function can be computed using minimax value iteration over $G^*$ using the following Bellman equation:

\[
V_{CV}(s,y,\bot) = \min_{a \in A} \left[ C(s,a) + \max_{\xi \in \Xi(s,y,a)} \sum_{s' \in S} \xi(s') T(s,a,s') V_{CV}(s',y,\xi(s'),\bot) \right]. \tag{9}
\]

We denote by $\pi_{CV}$ as the policy corresponding to the value function obtained by solving Eq. 9. For some history, $h \in H^\alpha$, the adversary has unlimited power to perturb all histories assigned zero probability under the adversarial perturbations obtained by solving Eq. 8. For example, all histories passing through s1 to s2 are not reachable under the adversarial perturbations, as illustrated in Fig. 2b. Intuitively, this suggests that such histories do not contribute to the CVaR, as the CVaR can be computed as the expected value under the perturbed transition probabilities (Proposition 1).

**Remark 1.** State s1 is reachable in the original MDP, but assigned zero probability by the adversary. At states assigned zero probability by the adversary, $y = 0$ in the CVaR SSPG. When $y = 0$, the adversary has unlimited power to perturb the transition probabilities (Eq. 6). Therefore, when $y = 0$ the minimax value iteration in Eq. 9 optimises for the minimum worst-case cost, as the adversary has the power to make the agent transition deterministically to the worst possible successor state for all future transitions. We will refer to the approach from [Chow et al. 2015] as CVaR-Worst-Case because of this property that for histories assigned zero probability by the adversary, this method optimises for the minimum worst-case total cost. In the following section we show that this behaviour is unnecessarily conservative to optimise CVaR. We propose a method for finding an alternative policy to execute in such situations that optimises the expected value while maintaining the optimal CVaR.

**Lexicographic Optimisation of CVaR**

In this section we present the contributions of this paper. We begin with a formal problem statement.

**Problem 2.** Let $\mathcal{M}$ be an MDP. Find the policy $\pi^*$ that optimises the expected cost subject to the constraint that $\pi^*$ obtains the optimal CVaR at confidence level $\alpha$:

\[
\pi^* = \arg \min_{\pi \in \Pi_M^\alpha} \mathbb{E}[C(\pi)], \quad \text{s.t.} \quad \text{CVaR}_\alpha(C(\pi)) = \min_{\pi' \in \Pi_M^\alpha} \text{CVaR}_\alpha(C(\pi')). \tag{10}
\]

Our approach to this problem extends the approach to CVaR optimisation from Chow et al. (2015), which we have outlined. We begin by emphasising that in the approach in Chow et al. (2015), some histories in the original MDP have zero probability under the adversarial perturbations obtained by solving Eq. 8. For example, all histories passing through s1 in Fig. 2a are not reachable under the adversarial perturbations, as illustrated in Fig. 2b. Intuitively, this suggests that such histories do not contribute to the CVaR, as the CVaR can be computed as the expected value under the perturbed transition probabilities (Proposition 1).

In this paper, we investigate finding an alternative policy to execute when we reach such histories which are assigned zero probability by the adversary in the CVaR SSPG. By finding an appropriate alternative policy to execute in these situations, we are able to optimise the expected value whilst maintaining the optimal CVaR. We now state two properties of these histories which will be useful to ensure the policies obtained by our algorithm maintain the optimal CVaR. Full proofs are in the supplementary material.

**Proposition 2.** Let $\pi$ denote a CVaR-optimal policy for confidence level $\alpha$, and let $\sigma$ denote the corresponding adversary policy from Eq. 8. For some history, $h$, if $p_{\pi,h}(\sigma) > 0$ and $p_{(\pi,\sigma)}(h) = 0$, there exists a policy $\pi' \in \Pi_M^\alpha$ which may be executed from $h$ onwards for which the total cost received over the run is guaranteed to be less than or equal to $\text{VaR}_\alpha(C(\pi'))$.
Proposition 2 shows that if a state is reached which is assigned zero probability by the adversary, then there must exist a policy which can be executed from that state onwards for which the total cost is guaranteed not to exceed the VaR. The following proposition states that by switching to any policy that guarantees that the cost will not exceed the VaR, the same CVaR is maintained.

**Proposition 3.** During execution of policy $\pi$ optimal for $\text{CVaR}_\alpha(C^M_{\pi})$, we may switch to any policy $\pi'$ and still attain the same CVaR, provided that $\pi'$ is guaranteed to incur total cost of less than or equal to $\text{VaR}_\alpha(C^M_{\pi})$.

**CVaR-Expected-Value**

Proposition 3 establishes that during execution of $\pi_{CV}$ we can switch to another policy, $\pi'$, without influencing the CVaR, provided that the total cost of executing $\pi'$ is guaranteed not to exceed $\text{VaR}_\alpha(C^M_{\pi_{CV}})$. Proposition 3 establishes that such a policy exists if the current history is assigned zero probability by the adversary in the CVaR SSPG. However, as we shall illustrate in the following example there may be multiple policies which satisfy this criterion. Of these policies, we would like to find the one which optimises our secondary objective of expected value as stated in Problem 2.

**Example 2.** Consider again the model in Fig. 2. We already established in Example 1 that the optimal CVaR at $\alpha = 0.1$ is 10, which can be computed as the expected value under the Markov chain in Fig. 2b. The corresponding VaR at $\alpha = 0.1$ is also 10. All of the histories passing through state $s_1$ are assigned zero probability by the adversarial perturbations in the CVaR SSPG. At $s_1$, we can continue to execute any policy for which all histories are guaranteed to reach the goal with total cost less than or equal to the VaR threshold of 10. Executing either $d$ or $e$ at $s_1$ satisfies this property, and maintains the optimal CVaR of 10. The approach of Chow et al. (2015), which we refer to as CVaR-Worst-Case, would choose action $e$ as in this situation it optimises for the minimum worst-case cost (see Remark 1). However, $d$ achieves better expected value and still attains the optimal CVaR of 10. Therefore, to solve Problem 2 the policy should choose $d$. On the other hand, executing $e$ attains a sub-optimal CVaR of greater than 10, as some histories would receive a total cost of 20.

We wish to develop a general approach to finding the policy, $\pi'$, which optimises the expected value, subject to the constraint that the worst-case total cost must not exceed $\text{VaR}_\alpha(C^M_{\pi_{CV}})$. We know that such a policy exists for histories assigned zero probability by the adversary in the CVaR SSPG. Therefore, by switching from $\pi_{CV}$ to $\pi'$ when such histories occur, we can optimise the expected value while still attaining the optimal CVaR, thus solving Problem 2.

We first compute the minimum worst-case total cost at each state, $V_{\text{worst}}(s)$, using the following minimax Bellman equation which assumes that the agent always transitions deterministically to the worst-case successor state:

$$V_{\text{worst}}(s) = \min_{a \in A} \left[ C(s, a) + \max_{s' \in S} \left( \mathbb{I}(T(s, a, s') > 0) \cdot V_{\text{worst}}(s') \right) \right],$$

(11)

where $\mathbb{I}$ denotes the indicator function. Let $\pi_{\text{worst}}$ denote the policy corresponding to $V_{\text{worst}}$. We also compute the optimal worst-case $Q$-values:

$$Q_{\text{worst}}(s, a) = C(s, a) + \max_{s' \in S} \mathbb{I}(T(s, a, s') > 0) \cdot V_{\text{worst}}(s').$$

(12)

Now, assume that the history so far is $h$, and that the cost so far in the history is $\text{cumul}_C(h)$. To maintain the optimal CVaR according to Proposition 3 we can allow an action $a$ to be executed at $h$ if

$$\text{cumul}_C(h) + Q_{\text{worst}}(s, a) \leq \text{VaR}_\alpha(C^M_{\pi_{CV}}).$$

(13)

For any action $a$ which satisfies Eq. 13 after taking $a$ we can then execute $\pi_{\text{worst}}$ and guarantee that the total cost will not exceed $\text{VaR}_\alpha(C^M_{\pi_{CV}})$. Thus, by finding a policy which optimises the expected value subject to the constraint that actions must satisfy Eq. 13 we can find the policy with best expected value which is guaranteed to have total cost less than or equal to $\text{VaR}_\alpha(C^M_{\pi_{CV}})$ and maintain the optimal CVaR.

Note that which actions are allowed in Eq. 13 depends not only on the current state, but also on the cost received so far. To arrive at an offline solution we create an augmented MDP, $M' = (S', A, T', C', G', s'_0)$, where we restrict which actions can be executed depending on the cost received so far. We start by augmenting the state space of the original MDP such that $S' = S \times [0, \text{VaR}_\alpha(C^M_{\pi_{CV}})]$. This augmented state keeps track of how much cost has been incurred. The action set $A$ is the same as the original MDP. The transition function ensures that actions are only enabled if they satisfy Eq. 13 and propagates the accumulated cost appropriately:

$$T'((s, c), a, (s', c')) =
\begin{cases}
T(s, a, s') & \text{if } c + Q_{\text{worst}}(s, a) \leq \text{VaR}_\alpha(C^M_{\pi_{CV}}) \\
0 & \text{otherwise}.
\end{cases}$$

(14)

The cost function is $C'((s, c), a) = C(s, a)$; the goal states are $G' = G \times [0, \text{VaR}_\alpha(C^M_{\pi_{CV}})]$; and the initial state is $s'_0 = (s_0, 0)$. The value function corresponding to the following standard MDP Bellman equation is the optimal expected value subject to the constraint that $\text{VaR}_\alpha(C^M_{\pi_{CV}})$ will not be exceeded for any possible history:

$$V^{M'}(s, c) =
\min_{a \in A} \left[ C'(s, a) + \sum_{(s', c')} T'((s, c), a, (s', c')) \cdot V^{M'}(s', c') \right].$$

(15)

Solving Eq. 15 is not straightforward due to the additional continuous state variable which prohibits standard discrete value iteration. Therefore, we instead apply value iteration with linear interpolation (Bertsekas 2008) for the continuous variable in the same manner as Chow et al. (2015).

We write $\pi'$ to denote the optimal policy corresponding to the value function $V^{M'}$. By first executing $\pi_{CV}$, and then executing $\pi'$ on histories assigned zero probability by the adversary in the CVaR SSPG, we switch to using the policy with best expected value subject to the constraint that the optimal CVaR is maintained. Therefore, this approach
solves Problem $2$ This approach, which we refer to as CVaR-Expected-Value is described in Algorithm $1$.

In this approach, we decide which actions are pruned out based on VaR$_{\alpha}(C_{\pi \text{CV}})$. To estimate VaR$_{\alpha}(C_{\pi \text{CV}})$, one approach would be to compute the worst-possible history in the Markov chain induced by Eq. $8$. However, this is computationally challenging as the number of reachable histories in the continuous state space may be extremely large.

Therefore, we take the simple approach of first executing $\pi_{\text{CV}}$ (CVaR-Worst-Case) for many episodes and compute an empirical estimate for VaR$_{\alpha}(C_{\pi \text{CV}})$, which we then use to restrict actions according to Eq. $14$. The algorithm is as follows:

$$\text{Algorithm 1: CVaR-Expected-Value}$$

- get $\pi_{\text{CV}}$ and $V_{\text{CV}}(s, y)$ by solving Equation $9$.
- get $\pi'$ by solving Equation $15$.
- $s^+ \leftarrow s_0'$
- $c \leftarrow 0$
- function ExecuteEpisode()
  - $a \leftarrow \pi_{\text{CV}}(s^+)$
  - $\xi = \arg \max_{\xi \in \mathcal{E}(s, y, a)} \sum_{s' \in S} \xi(s') \cdot T(s', y, \xi(s'), \perp)$
  - $s^+ \leftarrow (s', y, \xi(s'), \perp)$, where $s'$ is successor after executing $a$ in the MDP
  - $c \leftarrow c + C(s, a)$
  - if $s^+ \in G^+$,
    - return
  - while $s \notin G$:
    - $a \leftarrow \pi'(s, c)$
    - $s \leftarrow s'$, where $s'$ is successor after executing $a$ in the MDP
    - $c \leftarrow c + C(s, a)$
  - return

The code and data used to run the experiments is included in the supplementary material and will be made publicly available. We experimentally evaluate the following three approaches: CVaR-Worst-Case (CVaR-WC), CVaR-Expected-Value (CVaR-EV), and Expected Value (EV). Expected Value is the policy which optimises the expected value only. All algorithms are implemented in C++ and Gurobi is used to solve linear programs where necessary. We use 30 interpolation points for $\alpha$ to solve Eq. $9$ and 100 interpolation points for the cumulative cost to solve Eq. $15$. The experiments used a 3.2 GHz Intel i7 processor with 64 GB of RAM.

We compare the approaches on the following three domains. The two synthetic domains are from the literature, and the third domain we introduce is based on real data.

**Betting Game** We adapt this domain from the literature on CVaR in MDPs [Bäuerle and Ott 2011]. The state is represented by two factors: $(\text{money}, \text{stage})$. The agent begins with $(\text{money} = 5)$. The amount of money that the agent can have is limited between 0, and $\max(\text{money}) = 100$. At each stage the agent may choose to place a bet from $\text{bets} = \{0, 1, 2, 3, 4, 5\}$ provided that sufficient money is available. If the agent wins the game at that stage, an amount of money equal to the bet placed is received. If the agent loses the stage, the money bet is lost. If the agent wins the jackpot, the agent receives $10 \times$ the amount bet. For each stage, the probability of winning is 0.7, the probability of winning the jackpot is 0.05, and the probability of losing is 0.25. After 10 stages, the cost received is $\max(\text{money}) - \text{money}$, i.e. lower costs mean more money has been won.

**Deep Sea Treasure** This domain is adapted from the literature on multi-objective optimisation in MDPs [Vamplew et al. 2011]. A submarine navigates a gridworld to collect one of many treasures, each of which is associated with a reward value, $r$ (illustrated in the supplementary material). At each timestep, the agent chooses from 8 actions corresponding to directions of travel and moves to the corresponding square with probability 0.6 and each of the adjacent squares with probability 0.2. The episode ends when the agent reaches a treasure or the horizon of 15 steps is reached. The agent incurs a cost of 5 at each timestep, and a terminal cost of 500 $- r$. There is a tradeoff between risk and expected cost as if the submarine travels further it may collect more valuable treasure, and therefore incur less cost, but risks running out of time before reaching any treasure.

**Autonomous Vehicle Navigation** An autonomous vehicle must plan routes across Los Angeles, USA between a start and goal location as illustrated in Figure 1. We access real road traffic data collected by Caltrans Performance Measurement System (PeMS) by over 39,000 real-time traffic sensors deployed across the major metropolitan areas of California [Chen et al. 2001]. We select a subset of 263 sensors along the major freeways, shown as yellow markers in Figure 1. We specify two types of transitions: $i)$ freeway transitions (red) along a specified freeway where the transition time distribution is generated from historical PeMS traffic data (discrete with 10 bins), and $ii)$ between-freeway transitions (blue) where each state is connected to its three nearest neighbours on other freeways and the transition time is normally distributed around the expected duration obtained from querying the Google Routes API. In this domain, the cost is the journey duration in minutes. To simulate rare traffic jams on the freeways, uniform noise in $[0, 0.1]$ is added to the slowest freeway transitions (and probabilities renormalised). This is motivated by knowledge that rare but severe traffic jams can affect transition durations on freeways, and introduces a tradeoff between risk and expected cost.

**Results** The results of our experiments are presented in Table 1. The rows in the table indicate the method and the confidence level that the method is set to optimise. The columns indicate the performance of the policy for each objective over 20,000 evaluation episodes. We measure performance for CVaR$_{0.02}$ (i.e. the mean cost of the worst 2% of runs), CVaR$_{0.2}$ (worst 20%), and the expected value. We expect the CVaR-EV method to match the CVaR-WC performance on its CVaR measurement column. We also expect CVaR-EV to achieve lower cost than CVaR-WC in expectation.

In the Betting Game domain, EV performs the best for
Table 1: Results from evaluating the returns of each method for 20,000 episodes. Brackets indicate standard errors.

| Method             | CVaR-EV | Expected Value |
|--------------------|---------|----------------|
| CVaR-Worst-Case (α = 0.02) | 95.0 (0.0) | 95.0 (0.0) |
| CVaR-Expected-Value (α = 0.02) | 95.0 (0.0) | 95.0 (0.0) |
| Expected Value     | 100.0 (0.0) | 58.26 (0.22) |

| Method             | CVaR-EV | Expected Value |
|--------------------|---------|----------------|
| CVaR-Worst-Case (α = 0.2) | 91.97 (0.08) | 82.95 (0.06) |
| CVaR-Expected-Value (α = 0.2) | 91.86 (0.08) | 75.63 (0.16) |
| Expected Value     | 97.36 (0.07) | 58.26 (0.22) |

Betting Game domain

| Method             | CVaR-EV | Expected Value |
|--------------------|---------|----------------|
| CVaR-Worst-Case (α = 0.02) | 502.25 (0.78) | 402.74 (0.43) |
| CVaR-Expected-Value (α = 0.02) | 502.98 (0.83) | 352.06 (0.57) |
| Expected Value     | 575.0 (0.0) | 315.15 (0.66) |

| Method             | CVaR-EV | Expected Value |
|--------------------|---------|----------------|
| CVaR-Worst-Case (α = 0.2) | 422.29 (0.62) | 349.45 (0.42) |
| CVaR-Expected-Value (α = 0.2) | 422.80 (0.63) | 340.12 (0.49) |
| Expected Value     | 453.07 (1.09) | 315.15 (0.66) |

Deep Sea Treasure domain

| Method             | CVaR-EV | Expected Value |
|--------------------|---------|----------------|
| CVaR-Worst-Case (α = 0.02) | 210.20 (1.56) | 167.35 (0.11) |
| CVaR-Expected-Value (α = 0.02) | 211.16 (1.40) | 164.14 (0.12) |
| Expected Value     | 315.99 (2.12) | 120.19 (0.38) |

| Method             | CVaR-EV | Expected Value |
|--------------------|---------|----------------|
| CVaR-Worst-Case (α = 0.2) | 172.05 (0.80) | 134.72 (0.21) |
| CVaR-Expected-Value (α = 0.2) | 171.58 (0.80) | 132.77 (0.22) |
| Expected Value     | 217.86 (0.71) | 120.19 (0.38) |

Autonomous Vehicle Navigation domain

| Method             | CVaR-EV | Expected Value |
|--------------------|---------|----------------|
| CVaR-Worst-Case (α = 0.02) | 6215 | 8156 |
| CVaR-Expected-Value (α = 0.02) | 6526 | 10653 |
| Expected Value     | 34.5 | 505 |

Table 2: Computation times for each approach in seconds.

| Method             | Betting Game | Deep Sea Treasure | Vehicle Navigation |
|--------------------|--------------|------------------|---------------------|
| CVaR-Worst-Case    | 6215         | 8156             | 18327               |
| CVaR-Expected-Value | 6526         | 10653            | 23020               |
| Expected Value     | 34.5         | 505              | 1876                |

expected value, but worse for CVaR0.02 and CVaR0.2 compared to the policies which optimise CVaR. For CVaR-WC and CVaR-EV with $\alpha = 0.02$, the optimal policy is never to bet, and this policy attains the best performance for CVaR0.02. For $\alpha = 0.2$, both CVaR-WC and CVaR-EV achieve similar performance for CVaR0.2. However, CVaR-EV attains significantly lower cost in expectation. This occurs because winning the jackpot is usually sufficient to guarantee that the VaR will not be exceeded. In these situations, CVaR-WC stops betting. On the other hand, CVaR-EV bets aggressively in these situations, as bets can safely be made without the risk of having a bad run which would influence the CVaR.

Histograms for the total costs received by CVaR-WC and CVaR-EV with $\alpha = 0.2$, and EV are shown in Figure (a) for the betting domain. Equivalent plots for the other domains are in the supplementary material. We see that while EV performs the best in expectation, it also incurs the most poor runs where the cost is 100, corresponding to money = 0. This causes EV to perform worse at CVaR0.2. For CVaR-WC and CVaR-EV, we observe that the right side of the distributions are equivalent, resulting in the same performance for CVaR0.2. However, for CVaR-EV, the left side of the distribution is spread further left, improving the expected value.

For Deep Sea Treasure, both CVaR-WC and CVaR-EV achieve the same CVaR performance when optimising each of the CVaR0.02 and CVaR0.2 objectives. However, in both cases CVaR-EV obtains better expected value performance. EV obtains the best expected performance, but performs poorly at the CVaR objectives.

The same observations are made for the Autonomous Vehicle Navigation domain, but with a more modest improvement made in the expected value performance of CVaR-EV over CVaR-WC. However, even these small improvements in efficiency could result in substantial cost savings for a large-scale autonomous taxi business.

The computation times in Table indicate that the computation required for CVaR-EV is only a moderate (5%-30%) increase over the computation required for CVaR-WC.

Conclusion

In this paper, we have presented an approach to optimising the expected value in MDPs subject to the constraint that the CVaR is optimal. Our experimental evaluation on three domains has demonstrated that our approach is able to attain optimal CVaR while improving the expected performance over previous work. In future work, we wish to improve scalability by extending our approach to use labelled real-time dynamic programming (Bonet and Geffner2003) rather than value iteration over the entire state space.
References

Altman, E. 1999. Constrained Markov decision processes, volume 7. CRC Press.

Artzner, P.; Delbaen, F.; Eber, J.-M.; and Heath, D. 1999. Coherent measures of risk. Mathematical finance, 9(3): 203–228.

Bäuerle, N.; and Ott, J. 2011. Markov decision processes with average-value-at-risk criteria. Mathematical Methods of Operations Research, 74(3): 361–379.

Bertsekas, D. P. 2008. Approximate dynamic programming. B. B.; and G. 2003. Labeled RTDP: Improving the Convergence of Real-Time Dynamic Programming. In ICAPS, volume 3, 12–21.

Borkar, V.; and Jain, R. 2010. Risk-constrained Markov decision processes. In 49th IEEE Conference on Decision and Control (CDC), 2664–2669. IEEE.

Chen, C.; Petty, K.; Skabardonis, A.; Varaiya, P.; and Jia, Z. 2001. Freeway performance measurement system: mining loop detector data. Transportation Record, 1748(1): 96–102.

Chow, Y.; and Ghavamzadeh, M. 2014. Algorithms for CVaR optimization in MDPs. In Advances in neural information processing systems, 3509–3517.

Chow, Y.; Ghavamzadeh, M.; Janson, L.; and Pavone, M. 2017. Risk-constrained reinforcement learning with percentile risk criteria. The Journal of Machine Learning Research, 18(1): 6070–6120.

Chow, Y.; Tamar, A.; Mannor, S.; and Pavone, M. 2015. Risk-sensitive and robust decision-making: a cvr optimization approach. In Advances in Neural Information Processing Systems, 1522–1530.

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

Keramati, R.; Dann, C.; Tamkin, A.; and Brunskill, E. 2020. Being optimistic to be conservative: Quickly learning a cvr policy. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 34, 4436–4443.

Kolobov, A. 2012. Planning with Markov decision processes: An AI perspective. Synthesis Lectures on Artificial Intelligence and Machine Learning, 6(1): 1–210.

Lacerda, B.; Faruq, F.; Parker, D.; and Hawes, N. 2019. Probabilistic planning with formal performance guarantees for mobile service robots. The International Journal of Robotics Research, 38(9): 1098–1123.

Lacerda, B.; Parker, D.; and Hawes, N. 2015. Nested Value Iteration for Partially Satisfiable Co-Safe LTL Specifications (Extended Abstract). In AAAI Fall Symposium on Sequential Decision Making for Intelligent Agents (SDMIA).

Majumdar, A.; and Pavone, M. 2020. How should a robot assess risk? Towards an axiomatic theory of risk in robotics. In Robotics Research, 75–84. Springer.

Mouaddib, A.-I. 2004. Multi-objective decision-theoretic path planning. In IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA’04, 2004, volume 3, 2814–2819. IEEE.
Proof of Proposition 2
Let $\pi$ denote a CVaR-optimal policy for confidence level $\alpha$, and let $\sigma$ denote the corresponding adversary policy from Equation 8. For some history, $h$, if $P_{\pi}^M(h) > 0$ and $P_{(\pi, \sigma)}^M(h) = 0$, there exists a policy $\pi' \in \Pi_{G}^M$ which may be executed from $h$ onwards for which the total cost received over the run is guaranteed to be less than or equal to $VaR_{\alpha}(C_\pi^M)$.

Proof. We prove Proposition 2 by showing that the original policy $\pi$ must satisfy this condition. Denote by $\mathcal{H}_g$ the set of all histories ending at a goal state. We write $h_g \in \mathcal{H}_g^M$ to denote a specific history ending at a goal state.

Consider a perturbation function, $\delta(h_g)$, such that $0 \leq \delta(h_g) \leq \frac{1}{\alpha}$ and $\sum_{h_g} \delta(h_g) \cdot P_{\pi}^M(h_g) = 1$. The expected value of $C_\pi^M$ under a distribution perturbed by $\delta$ is:

$$E_\delta[C_\pi^M] = \sum_{h_g \in \mathcal{H}_g^M} \delta(h_g) \cdot P_{\pi}^M(h_g) \cdot \text{cumul}_C(h_g). \quad (16)$$

By the CVaR representation theorem in Equation 2, we have:

$$\max_\delta E_\delta[C_\pi^M] = \text{CVaR}_{\alpha}(C_\pi^M). \quad (17)$$

Let $\mathcal{H}_{g, \text{VaR}}^M \subseteq \mathcal{H}_g^M$ denote the set of histories for which $\text{cumul}_C(h_g) = \text{VaR}_{\alpha}(C_\pi^M)$. Any perturbation function which maximises Equation 17 denoted by $\delta^*$, must satisfy the following conditions:

$$\delta^*(h_g) = 1/\alpha \quad \text{if} \quad \text{cumul}_C(h_g) > \text{VaR}_{\alpha}(C_\pi^M), \quad (18)$$

$$\delta^*(h_g) = 0 \quad \text{if} \quad \text{cumul}_C(h_g) < \text{VaR}_{\alpha}(C_\pi^M), \quad (19)$$

$$\sum_{h_g \in \mathcal{H}_{g, \text{VaR}}^M} \delta^*(h_g) \cdot P_{\pi}^M(h_g) = 1 - \frac{1}{\alpha} \cdot \text{Pr}(C_{\pi}^M > \text{VaR}_{\alpha}(C_\pi^M)). \quad (20)$$

Equation 20 ensures that the perturbed distribution sums to 1. Notably, $\delta^*(h_g) = 0$ only if $\text{cumul}_C(h_g) \leq \text{VaR}_{\alpha}(C_\pi^M)$.

From Proposition 1 we also have that:

$$\text{CVaR}_{\alpha}(C_\pi^M) = \sum_{h_g \in \mathcal{H}_g^M} P_{(\pi, \sigma)}^M(h_g) \cdot \text{cumul}_C(h_g). \quad (21)$$

Therefore:

$$\sum_{h_g \in \mathcal{H}_g^M} \delta^*(h_g) \cdot P_{\pi}^M(h_g) \cdot \text{cumul}_C(h_g) =$$

$$\sum_{h_g \in \mathcal{H}_g^M} P_{(\pi, \sigma)}^M(h_g) \cdot \text{cumul}_C(h_g) =$$

$$\sum_{h_g \in \mathcal{H}_g^M} \delta^*(h_g) \cdot P_{\pi}^M(h_g) \cdot \text{cumul}_C(h_g) \quad (22)$$

where we define $\delta^*(h_g) = P_{(\pi, \sigma)}^M(h_g)/P_{\pi}^M(h_g)$. The values of $\delta^*(h_g)$ are constrained in the same manner as $\delta(h_g)$ by the definition of the CVaR SSPG. Therefore, Equation 22 implies that $\delta^*(h_g)$ satisfies the same conditions as $\delta^*(h_g)$ in Equations 18-20. We have that $P_{(\pi, \sigma)}^G(h_g) = 0$ when $P_{\pi}^M(h_g) > 0$ only if $\delta^*(h_g) = 0$. $\delta^*(h_g) = 0$ only if $\text{cumul}_C(h_g) \leq \text{VaR}_{\alpha}(C_\pi^M)$ by Equations 18-20. Therefore:

$$P_{(\pi, \sigma)}^G(h_g) = 0 \quad \text{and} \quad P_{\pi}^M(h_g) > 0 \iff \text{cumul}_C(h_g) \leq \text{VaR}_{\alpha}(C_\pi^M). \quad (23)$$

Consider some history, $h$, where $P_{(\pi, \sigma)}^G(h_g) = 0$ and $P_{\pi}^M(h_g) > 0$. For all histories $h_g \in \mathcal{H}_g^M$ reachable after $h$ under $\pi$ (i.e. for which $P_{\pi}^M(h_g) > 0$), $P_{(\pi, \sigma)}^G(h_g) = 0$. Therefore, by Equation 23 all histories $h_g$ reachable after $h$ under $\pi$ are guaranteed to have $\text{cumul}_C(h_g) \leq \text{VaR}_{\alpha}(C_\pi^M)$. This completes the proof that for any history $h$, where $P_{(\pi, \sigma)}^G(h_g) = 0$ and $P_{\pi}^M(h_g) > 0$, by continuing to execute policy $\pi$, the total cost is guaranteed to be less than or equal to $\text{VaR}_{\alpha}(C_\pi^M)$.

Proof of Proposition 3
During execution of policy $\pi$ optimal for CVaR$_{\alpha}(C_\pi^M)$, we may switch to any policy $\pi'$ and still attain the same CVaR, provided that $\pi'$ is guaranteed to incur total cost of less than or equal to $\text{VaR}_{\alpha}(C_\pi^M)$.

Proof. Let $\mathcal{H}_{g, \text{VaR}}^M \subseteq \mathcal{H}_g^M$ denote the set of histories where $\text{cumul}_C(h_g) \leq \text{VaR}_{\alpha}(C_\pi^M)$, and $\mathcal{H}_{g, \text{VaR}}^M \subseteq \mathcal{H}_g^M$ denote the set of histories where $\text{cumul}_C(h_g) > \text{VaR}_{\alpha}(C_\pi^M)$. If for any $h_g \in \mathcal{H}_{g, \text{VaR}}^M$, there exists a policy $\pi' \neq \pi$ such that executing $\pi'$ partway through $h_g$ is guaranteed to receive total cost less than or equal to $\text{VaR}_{\alpha}(C_\pi^M)$, then switching to $\pi'$ would improve the CVaR, contradicting that $\pi$ is a CVaR-optimal policy. Therefore, there will never be a suitable $\pi' \neq \pi$ to switch to along any $h_g \in \mathcal{H}_{g, \text{VaR}}^M$.

If partway through any $h_g \in \mathcal{H}_{g, \text{VaR}}^M$, we switch to a valid $\pi'$ instead of $\pi$, this can result in no total costs above the VaR by the definition of $\pi'$ in Proposition 3. Therefore, switching to $\pi'$ cannot modify the distribution of total costs which exceed $\text{VaR}_{\alpha}(C_\pi^M)$. Additionally, because no total costs which are less than or equal to $\text{VaR}_{\alpha}(C_\pi^M)$ can be increased above $\text{VaR}_{\alpha}(C_\pi^M)$ by switching to $\pi'$, the VaR cannot increase.

Because switching from $\pi$ to $\pi'$ cannot change the distribution of total costs above the VaR, and it cannot increase the VaR, we observe from the definition of CVaR in Equation 1 that the CVaR cannot be increased. Because $\pi$ already attains the optimal CVaR, the strategy of switching to $\pi'$ must attain the optimal CVaR.
Histograms of Total Costs

Figure 4: Histograms for the total cost received over 20000 evaluation episodes in the Deep Sea Treasure domain.

Illustration of Deep Sea Treasure Domain

Figure 6: Deep Sea Treasure domain: golden squares indicate different treasure reward values.