Predictive CVaR Q-learning

Ju-Hyun Kim (KRISS) & Seungki Min (SNU)

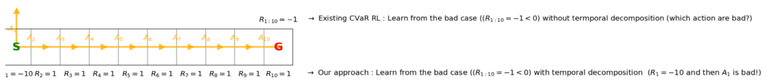
jhkim@kriss.re.kr, mskyt@snu.ac.kr

ICLR 2026

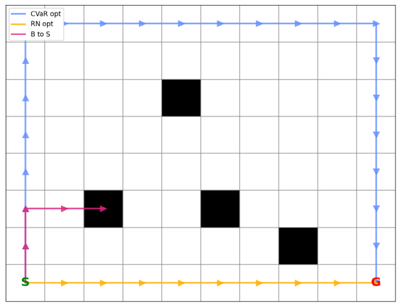
Motivation

- **Sample Inefficiency in CVaR RL:** Optimizing CVaR in RL is notoriously sample-inefficient because it relies on a narrow subset of worst-case trajectories. This stems from two fundamental issues:
  - Noisy policy evaluation due to a lack of temporal decomposition
  - Ineffective exploration due to the worst-case outcomes focusing
- **Issue 1: Lack of Temporal Decomposition**
  - Traditionally, CVaR objective is treated as a single, **non-decomposable** terminal total reward.
  - This collapses the learning signal, preventing the agent from assessing the immediate impact of its actions.

→ We reformulate the CVaR objective into a **temporally decomposable structure** that admits a risk-neutral Bellman-style recursion and step-by-step feedback.



- **Issue 2: Focus on Worst-case Outcomes**
  - Because learning is driven by failures, **high-return trajectories are ignored**. This phenomenon, known as **"blindness to success"**, causes the agent to stagnate in overly conservative, suboptimal policies.



→ In addition to conventional action-level exploration (ϵ-greedy), we introduce novel **risk-level exploration** that randomizes the agent's risk sensitivity.

Problem

- Our goal is to find an optimal policy π^* that maximizes the CVaR of the total return $R_{1:T}$:

$$\sup_{\pi\in\Pi}\left\{q\cdot \mathrm{CVaR}_q^\pi[R_{1:T}]\right\} = \max_{\eta\in\mathbb{R}}\left\{q\eta + \sup_{\pi\in\Pi}\mathbb{E}^\pi\left[-(\eta - R_{1:T})^+\right]\right\}.$$

  - **State Space Augmentation:** This variational form naturally introduces an auxiliary variable for the risk budget.
  - We introduce a **residual tail budget process** $(Y_t^\eta)_{t=1}^{T+1}$ defined as

$$Y_t^\eta := \eta - R_{1:t-1},$$

    where $\eta \in \mathbb{R}$ is an auxiliary variable specifying the initial budget.
- **Augmented Markov Policies (Bäuerle & Ott, 2011):** For dynamic CVaR optimization, it is sufficient to search over Markov policies defined on the augmented state space (S_t, Y_t), rather than all history-dependent policies.
  - Such a policy is specified by an augmented Markov policy *kernel* $\chi = (\chi_t)_{t=1}^T$, mapping the current state and residual budget to an action distribution: $\chi_t : \mathcal{S}\times\mathbb{R}\to\Delta^{|\mathcal{A}|}$.
  - Actions are sampled as:

$$A_t \sim \chi_t(\cdot \mid S_t, Y_t^\eta).$$

- **Transformed Objective:** Our problem is now reduced to finding the optimal initial budget η and policy kernel χ:

$$\sup_{\pi\in\Pi}\left\{q\cdot \mathrm{CVaR}_q^\pi[R_{1:T}]\right\} = \max_{\eta\in\mathbb{R}}\left\{q\eta + \sup_{\chi\in\mathcal{X}}\mathbb{E}^{\chi,\eta}\left[-(\eta - R_{1:T})^+\right]\right\}.$$

Sol 1. Recursive Formulation for the CVaR Objective

- **Key Idea:** To resolve the noisy policy evaluation caused by delayed terminal rewards, we introduce a novel recursive formulation for the CVaR objective utilizing a risk-neutral Bellman-style approach.
- **Predictive Functions:** We define a pair of predictive functions that permit temporal decomposition, decoupling the estimation of tail value and tail probability.

Definition 1 (Predictive tail value/probability functions). *Given an augmented Markov policy kernel χ, its predictive tail value function $f^\chi = (f_t^\chi : \mathcal{S}\times\mathbb{R}\times\mathcal{A}\to\mathbb{R})_{t=1}^{T+1}$ is defined as*

$$f_t^\chi(s,y,a) := \mathbb{E}^{\chi,\eta=0}\left[\mathbb{I}\{R_{t:T}\le y\}R_{t:T} \;\middle|\; S_t = s, Y_t^{\eta=0} = y, A_t = a\right],$$

with $f_{T+1}^\chi(s,y,a) := 0$. Additionally, its predictive tail probability function $g^\chi = (g_t^\chi : \mathcal{S}\times\mathbb{R}\times\mathcal{A}\to[0,1])_{t=1}^{T+1}$ is defined as

$$g_t^\chi(s,y,a) := \mathbb{P}^{\chi,\eta=0}\left(R_{t:T}\le y \;\middle|\; S_t = s, Y_t^{\eta=0} = y, A_t = a\right),$$

with $g_{T+1}^\chi(s,y,a) := \mathbb{I}\{0\le y\}$.

Here, the choice $\eta = 0$ is arbitrary. Above notion of predictive tail values and probabilities are invariant in η.

- **Bellman Equation (Theorem 1):** The predictive tail value function exhibits a recursive structure analogous to the standard Bellman equation in the risk-neutral setting, enabling efficient value propagation.

Theorem 1 (Bellman equation). *Given an augmented Markov policy kernel χ, its predictive tail value function f^χ and predictive tail probability function g^χ satisfy*

$$f_t^\chi(s,y,a) = \mathbb{E}_{(R_t,S_{t+1})\sim\mathcal{P}_t(\cdot|s,a),A_{t+1}\sim\chi_{t+1}(\cdot|S_{t+1},y-R_t)}\left[f_{t+1}^\chi\big(S_{t+1},y-R_t,A_{t+1}\big) + g_{t+1}^\chi\big(S_{t+1},y-R_t,A_{t+1}\big)\times R_t\right],$$

for all $s\in\mathcal{S}$, $y\in\mathbb{R}$, $a\in\mathcal{A}$, and $t\in\{1,\dots,T\}$.

- **Bellman Optimality Equation (Theorem 2):** Building on this recursion, we derive the optimality conditions that the optimal kernel must satisfy and establish its connection to the CVaR objective.

Theorem 2 (Bellman optimality equation). *Define*

$$v_t^\chi(s,y) := \mathbb{E}_{A_t\sim\chi_t(\cdot|s,y)}\left[f_t^\chi(s,y,A_t) - g_t^\chi(s,y,A_t)\times y\right], \quad v_t^*(s,y) := \sup_{\chi\in\mathcal{X}} v_t^\chi(s,y).$$

Then, the following holds:

1. *Let $\Pi(\chi)$ be the set of augmented Markov policies induced by a kernel χ across all values of $\eta\in\mathbb{R}$. Then,*

$$\sup_{\pi\in\Pi(\chi)}\left\{q\cdot \mathit{CVaR}_q^\pi[R_{1:T}]\right\} = \max_{\eta\in\mathbb{R}}\left\{q\eta + v_1^\chi(s_1,\eta)\right\}.$$

2. *With respect to all non-anticipating policies,*

$$\sup_{\pi\in\Pi}\left\{q\cdot \mathit{CVaR}_q^\pi[R_{1:T}]\right\} = \max_{\eta\in\mathbb{R}}\left\{q\eta + v_1^*(s_1,\eta)\right\}.$$

3. *$v^\chi \equiv v^*$ if and only if χ is greedy with respect to (f^χ, g^χ).*

- **Policy Improvement (Theorem 3):** We guarantee that state-wise greedy updates of the kernel in the augmented state space yield monotonic improvements in the overall CVaR performance.

Theorem 3 (Policy improvement). *Consider an augmented Markov policy kernel χ along with its predictive tail value function f^χ and predictive tail probability function g^χ. Let χ' be the greedy kernel with respect to (f^χ, g^χ). Then,*

$$v_t^\chi(s,y)\le v_t^{\chi'}(s,y), \quad \forall s\in\mathcal{S}, y\in\mathbb{R}, t\in\{1,\dots,T\}.$$

Consequently,

$$\sup_{\pi\in\Pi(\chi)}\mathit{CVaR}_q^\pi[R_{1:T}] \le \sup_{\pi\in\Pi(\chi')}\mathit{CVaR}_q^\pi[R_{1:T}],$$

for any $q\in(0,1]$.

Sol 2. Two-way Randomized Exploration

- **Key Idea:** Mitigate the *blindness to success* phenomenon by encouraging the agent to experience trajectories under varying degrees of risk sensitivity.
- **Action-Level Exploration (ϵ-greedy):** With probability ϵ, choose a random action to ensure sufficient exploration of the action space.
- **Risk-Level Exploration:** Instead of fixing the initial budget to a single estimator η, sample $Y_1\sim\mathcal{N}(\eta,\sigma_k^2)$ around the current estimator η.
  → This prompts the agent to experience trajectories under varying degrees of risk sensitivity (sometimes acting boldly, sometimes conservatively), thus exploring the space of risk preferences more comprehensively.

Algorithm Overview

- **Initialization:** Initialize the risk budget parameter η and neural network parameters θ,ϕ for the predictive function approximators $\hat{f}^\theta$ and $\hat{g}^\phi$.
- **Generating Sample Trajectories (Two-way Exploration):**
  - Risk-level exploration: Sample the initial budget $Y_1\sim\mathcal{N}(\eta,\sigma_k^2)$.
  - Action-level exploration: With probability $1-\epsilon$, choose the greedy action:

$$A_t \leftarrow \arg\max_{a\in\mathcal{A}}\left\{\hat{f}_t^\theta(S_t,Y_t,a) - \hat{g}_t^\phi(S_t,Y_t,a)\cdot Y_t\right\}$$

    Otherwise, take a random action with probability ϵ.
- **Updating the Predictive Functions and Risk Budget:**
  - Update function parameters θ and ϕ via TD losses derived from the Bellman recursion:

$$\theta\leftarrow\theta-\alpha_{\theta,k}\nabla_\theta\mathcal{L}_f(\theta;\mathcal{B}), \quad \phi\leftarrow\phi-\alpha_{\phi,k}\nabla_\phi\mathcal{L}_g(\phi;\mathcal{B}).$$

    where

$$\mathcal{L}_f(\theta;\mathcal{B}) := \frac{1}{B|H|}\sum_{\eta'\in H}\sum_{j=1}^{B}\Big(\hat{f}_j^\theta(S_j,\eta'-R_{1:j-1},A_j) - \left[\hat{f}_{j+1}^\theta(S_{j+1},\eta'-R_{1:j},A'_{j+1}) + \hat{g}_{j+1}^\phi(S_{j+1},\eta'-R_{1:j},A'_{j+1})\cdot R_j\right]\Big)^2,$$

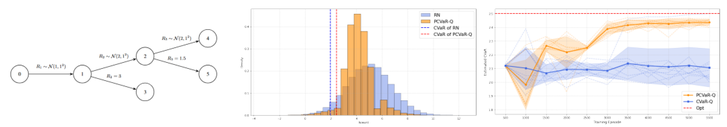
$$\mathcal{L}_g(\phi;\mathcal{B}) := \frac{1}{B|H|}\sum_{\eta'\in H}\sum_{j=1}^{B}\Big(\hat{g}_{j+1}^\phi(S_{j+1},\eta'-R_{1:j},A'_{j+1}) - \hat{g}_j^\phi(S_j,\eta'-R_{1:j-1},A_j)\Big)^2.$$

  - Periodically update the central risk budget η by solving the outer optimization problem:

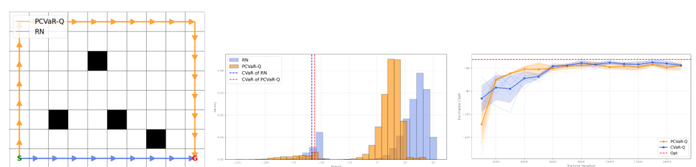
$$\eta\leftarrow\arg\max_{\eta'\in H}\max_{a\in\mathcal{A}}\left\{\hat{f}_1^\theta(s_1,\eta',a) + \eta'\cdot\left(q-\hat{g}_1^\phi(s_1,\eta',a)\right)\right\}.$$

Numerical Experiments

- **Sequential Decision Tree (Tree-structured MDP)**
  - **Result Summary:** In an environment with clear risk-return trade-offs, PCVaR-Q algorithm successfully identifies the CVaR-optimal policy. It demonstrates a highly stable learning curve and much faster convergence compared to the baseline CVaR-Q method.



- **Stochastic Grid-world with Obstacles**
  - **Result Summary:** In the above environment, PCVaR-Q learns a robust, obstacle-avoiding path. It significantly improves the lower-tail return distribution while maintaining superior sample efficiency throughout training.



Contribution

- Temporal decomposition of the CVaR objective via predictive tail value/probability functions
- Bellman equation, optimality condition, and policy improvement theorem tailored to CVaR
- Two-way randomized exploration strategy to mitigate blindness to success
- Development and empirical validation of the sample-efficient PCVaR-Q algorithm