Efficient Risk–Sensitive Planning via Entropic Risk Measures

Alexandre Marthe[1] Samuel Bounan[1] Aurélien Garivier[1] Claire Vernade[2]

[1] ENS de Lyon [2] Eberhard Karls Universität Tübingen

Setting

MDP. We consider an undiscounted, finite-horizon MDP $(\mathcal{X}, \mathcal{A}, P, r, H)$. The (random) return of a policy π is

$$\mathcal{R}^\pi := \sum_{t=1}^{H} r(x_t, a_t).$$

Risk Sensitive Planning. For a functional ρ of the return (measure of risk), we want to compute an optimal policy

$$\pi^* \in \arg\max_\pi \rho(\mathcal{R}^\pi).$$

We aim to find ρ that is both *tractable* and *interpretable*.

Risk Sensitive Objectives

Entropic Risk Measure (EntRM).

$$\rho_\beta(X) = \frac{1}{\beta} \log \mathbb{E}\left[e^{\beta X}\right],$$

Tractable in MDPs: optimizable by dynamic programming.

Hard to interpret: how should we choose β?

Value at Risk family:

$$\mathrm{VaR}_\alpha(X) = \inf\{x \in \mathbb{R} \mid \mathbb{P}(X \le x) \ge \alpha\},$$
$$\mathrm{CVaR}_\alpha(X) = \mathbb{E}[X \mid X \le \mathrm{VaR}_\alpha(X)],$$

Hard to optimize: no easy dynamic programming.

Easily interpretable: α is a quantile level.

Main Question

Can we compute EntRM-optimal policies across β and pick one that maximizes a quantile ?

Approximating a quantile

Chernoff link. The Chernoff inequality approximates quantiles with the EntRM:

$$\left.\begin{matrix}\mathrm{CVaR}_\alpha(X) \\ \mathrm{VaR}_\alpha(X)\end{matrix}\right\} \approx \sup_{\beta \in \mathbb{R}^-} \mathrm{EntRM}_\beta(X) - \frac{1}{\beta}\log\frac{1}{\alpha}.$$

In terms of MDP, with $\pi^*_\beta = \arg\max_\pi \mathrm{EntRM}_\beta(\mathcal{R}^\pi)$,

$$\max_\pi \mathrm{(C)VaR}_\alpha(\mathcal{R}^\pi) \approx \sup_\beta \mathrm{EntRM}_\beta(\mathcal{R}^{\pi^*_\beta}) - \frac{1}{\beta}\log\frac{1}{\alpha}$$

Intuition. There is a parameter β that recovers the quantile level α, and gives a good policy for $\mathrm{(C)VaR}_\alpha$ and VaR_α.

General Policy Improvement. We solve this simplified problem instead,

$$\arg\max_{\beta \in \mathbb{R}} \rho(\mathcal{R}^{\pi^*_\beta})$$

and return π^*_β as the final policy.

Algorithm

1. Compute the EntRM-optimal policies.

1: Initialize $\beta \leftarrow 0$, $\Pi \leftarrow \emptyset$.
2: **while** $\beta \ge \beta_{\min}$ **do**
3: Compute the optimal policy π^*_β for the current β, using DP.
4: Update the set of optimal policies: add π^*_β to Π.
5: Compute ε_β such that π^*_β is also optimal for $\beta - \varepsilon_\beta$.
6: Choose $\Delta \ge \varepsilon_\beta$ and update $\beta \leftarrow \beta - \Delta$.
7: **end while**

- $\varepsilon_\beta = \frac{|\beta| H}{2} \min_{a,x} A_\beta(x,a)$, where A_β is the EntRM advantage function of π^*_β, ensures that π^*_β is also optimal for any $\beta' \in [\beta - \varepsilon_\beta, \beta]$.

2. Compute the distributions of returns.

- For each policy $\pi \in \Pi$, compute the distribution of returns $\eta^\pi = \mathrm{Law}(\mathcal{R}^\pi)$ using Distributional Policy Evaluation.
- If needed, use an approximation scheme for the distributions.

3. Compute the Objective and return the best policy.

$$\pi^* = \arg\max_{\pi \in \Pi} \rho(\eta^\pi)$$

Structure of EntRM-Optimal Policies

Piecewise-constancy across β. There exist breakpoints $\beta_1 < \cdots < \beta_K$ such that π^*_β is constant on each interval (β_k, β_{k+1}).

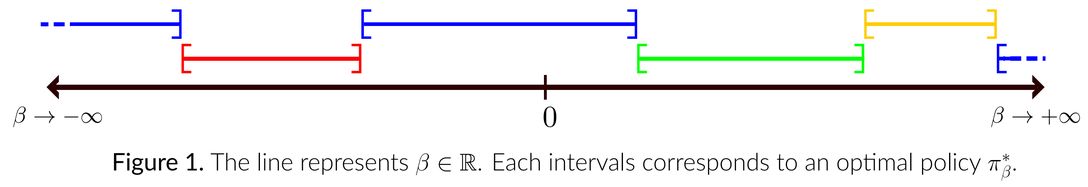

Figure 1. The line represents $\beta \in \mathbb{R}$. Each intervals corresponds to an optimal policy π^*_β.

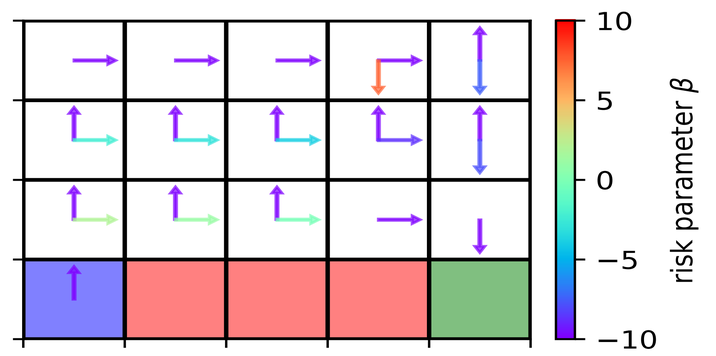

Figure 2. Optimal policies for different values of β on Windy Cliff environment. The color represents the first value of the risk parameter β for which the action is optimal. For low values of β (blue arrows, risk-averse), the agent avoids the cliff. For higher values of β (green to red arrows, risk-neutral to risk-seeking), the agent follows the cliff to get potentially higher rewards.

Experimental Results

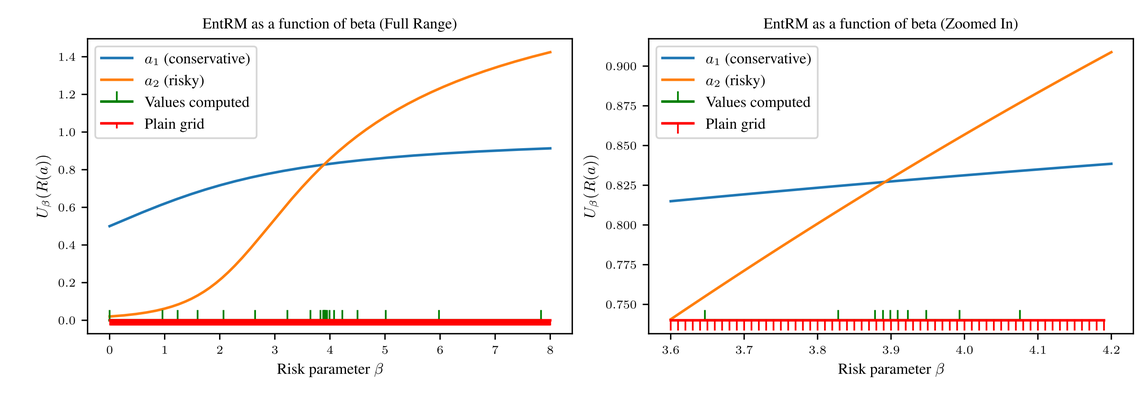

Figure 3. Illustration of the computation of the EntRM-optimal policies. The two curves represent the value of $\beta \mapsto \mathrm{EntRM}_\beta$ for two different policies. The highest curve at one point is the optimal policy for this value of β. The green values are the value computed using $\Delta = \varepsilon_\beta$. We observe that this method allows for efficient computation by avoiding some unnecessary evaluations.

Risk Measure	VaR		CVaR	
Risk parameter α	0.05	0.1	0.05	0.1
Optimality Front	**1.25**	**1.33**	**1.14**	**1.21**
Proxy Optimization	1.22	1.30	1.13	1.20
Risk neutral optimal	1.22	**1.33**	1.11	1.19
Nested Risk Measure	0.88	0.95	0.75	0.84

Table 1. Performance of different methods on the Cliff World environment. The best performance for each risk measure and parameter is in bold. We observe that our approach (Optimality Front) outperforms the other Dynamic Programming methods for computing VaR and CVaR policies.

EWRL18
September 17–19 2025, Tübingen