Soft Robust MDPs and Risk-Sensitive MDPs: Equivalence, Policy Gradient, and Sample Complexity

Runyu (Cathy) Zhang, Yang Hu, Na Li

Harvard University, School of Engineering and Applied Sciences

Robust / Risk-sensitive Decision Making

Robust Markov Decision Processes (RMDPs)

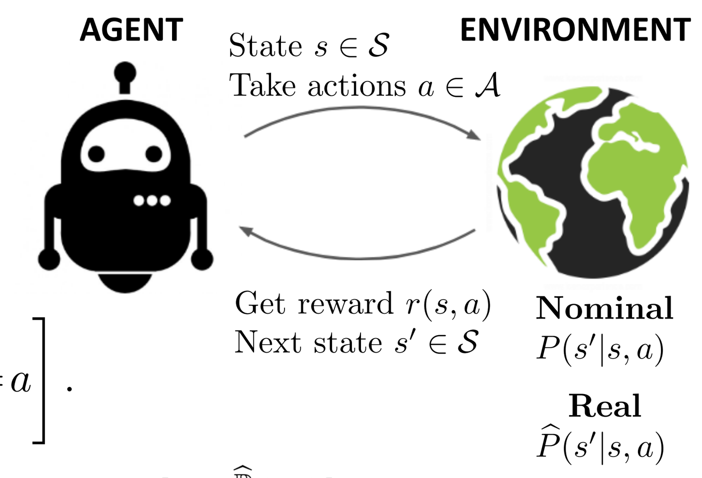
$\mathcal{S}$: state space, $\mathcal{A}$: action space, γ: discount factor, ρ: initial state distribution.

$\mathbb{P}(s'|s,a)$: transition kernel, $r(s,a)$: reward.

$\pi(a|s)$: (stationary Markovian) policy.

$$V^{\pi,\mathbb{P}}(s) := \mathbb{E}_{\pi,\mathbb{P}}\left[\sum_{t=0}^{\infty}\gamma^t r(s_t,a_t)\,\middle|\, s_0=s\right],$$

$$Q^{\pi,\mathbb{P}}(s,a) := \mathbb{E}_{\pi,\mathbb{P}}\left[\sum_{t=0}^{\infty}\gamma^t r(s_t,a_t)\,\middle|\, s_0=s, a_0=a\right].$$



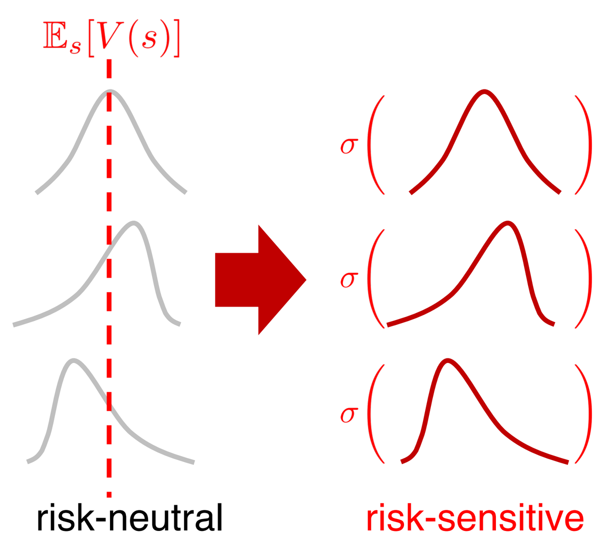
Objective for robustness: $\pi^* = \arg\max_\pi \min_{\widehat{\mathbb{P}}\in\mathcal{P}} \mathbb{E}_{s_0\sim\rho}[V^{\pi,\widehat{\mathbb{P}}}(s_0)]$.

Convex Risk Measures

A function $V:\mathbb{R}^{\mathcal{S}}\to\mathbb{R}$ is called a *convex risk measure* if and only if:

1. Monotonicity: $V(s)\le V'(s),\ \forall s\in\mathcal{S} \Longrightarrow \sigma(V)\le\sigma(V')$.
2. Translation invariance: $\forall d\in\mathbb{R}, \sigma(V+d)=\sigma(V)-d$.
3. Convexity: $\forall\lambda\in[0,1]$, $\sigma(\lambda V+(1-\lambda)V')\le \lambda\sigma(V)+(1-\lambda)\sigma(V')$.

Given a reference distribution $s\sim\mu$, write $\sigma(\mu, V)$.



Contribution #1: Equivalences

Soft-RMDPs (generalization of RMDPs)

$$\overline{V}^{\pi}(s) := \inf_{\{\widehat{\mathbb{P}}_t\}_{t\ge0}} \mathbb{E}_{\pi,\widehat{\mathbb{P}}}\left[\sum_t \gamma^t\left(r(s_t,a_t)+\gamma D(\widehat{\mathbb{P}}_{t;s_t,a_t},\mathbb{P}_{s_t,a_t})\right)\right].$$

e.g. $D(\hat{P},P)=KL(\hat{P}||P)$

⇕

Risk-sensitive MDPs

$$\widetilde{V}^{\pi}(s) = \sum_{a\in\mathcal{A}}\pi(a|s)\left(r(s,a)-\gamma\sigma(\mathbb{P}_{s,a},\widetilde{V}^{\pi})\right).$$

e.g. $\sigma(P,V)=\log\mathbb{E}_{s\sim P}e^{-\beta V(s)}$

Our equivalence theorem of risk-sensitive MDPs and soft-robust MDPs.

- $\overline{V}^{\pi}=\widetilde{V}^{\pi}, \overline{V}^{*}=\widetilde{V}^{*}; \overline{Q}^{\pi}=\widetilde{Q}^{\pi}, \overline{Q}^{*}=\widetilde{Q}^{*}$.
- Worst-case kernel $\widehat{\mathbb{P}}^{\pi}_{t;s,a}\equiv\arg\min_{\widehat{\mathbb{P}}_{s,a}}\left[D(\widehat{\mathbb{P}}_{s,a},\mathbb{P}_{s,a})+\mathbb{E}_{s'\sim\widehat{\mathbb{P}}}\overline{V}^{\pi}(s')\right] =: \widehat{\mathbb{P}}^{\pi}_{s,a}$.

Proof Enabler: Dual representation theorem [Föllmer & Schied, 2002]

risk measure		penalty function
$\sigma(V)=\sup_{\widehat{\mu}\in\Delta(\mathcal{S})}\left(-\mathbb{E}_{s\sim\widehat{\mu}}[V(s)]-D(\widehat{\mu})\right)$	$\Longleftrightarrow$	$D(\widehat{\mu})=\sup_V\left(-\sigma(V)-\mathbb{E}_{s\sim\widehat{\mu}}[V(s)]\right)$

Contribution #2: Soft-robust Policy Gradient

Soft-robust PG theorem:

$$\nabla_\theta V^{\pi_\theta}(s)=\mathbb{E}_{\pi_\theta,\widehat{\mathbb{P}}^{\pi_\theta}}\left[\sum_{t=0}^{\infty}\gamma^t Q^{\pi_\theta}(s_t,a_t)\nabla_\theta\log\pi_\theta(a_t|s_t)\,\middle|\, s_0=s\right]$$

$\Rightarrow$ (direct parametrization: $\pi(a|s)=\theta_{a,s}$)

$$\frac{\partial(\mathbb{E}_{s_0\sim\rho}V^{\pi_\theta}(s_0))}{\partial\theta_{s,a}}=\frac{1}{1-\gamma}d^{\pi_\theta,\widehat{\mathbb{P}}^{\pi_\theta}}(s)Q^{\pi_\theta}(s,a)$$

$\Rightarrow$ **Gradient dominance:**

$$\mathbb{E}_{s_0\sim\rho}[V^*(s_0)-V^{\pi_\theta}(s)]\le\left\|\frac{d^{\pi^*,\widehat{\mathbb{P}}^{\pi_\theta}}(\cdot)}{d^{\pi_\theta,\widehat{\mathbb{P}}^{\pi_\theta}}(\cdot)}\right\|_\infty \max_{\hat{\pi}}\langle\hat{\pi}-\pi_\theta,G(\theta)\rangle.$$

Iteration Complexity of Policy Gradient

$$\theta^{(k+1)}\leftarrow \text{Proj}_{\Delta(\mathcal{A})^{\mathcal{S}}}\left(\theta^{(k)}+\eta G(\theta^{(k)})\right),\ \text{where } \left[G(\theta^{(k)})\right]_{s,a}:=\frac{1}{1-\gamma}d^{\pi_\theta,\widehat{\mathbb{P}}^{\pi_\theta}}(s)Q^{\pi_\theta}(s,a).$$

Achieves ε-suboptimality in $\frac{16|\mathcal{A}|M^4}{(1-\gamma)^4\epsilon^2}$ iterations.

assuming sufficient exploration (i.e., $\min_{s,\pi} d^{\pi,\widehat{\mathbb{P}}^{\pi}}(s)\ge\frac{1}{M}$)

Sample-based generalization? Impractical to sample from an unknown kernel $\widehat{\mathbb{P}}^{\pi_\theta}$.

Contribution #3: Offline Sample-Based Learning

Intuition: 1) Define Bellman Operator $\mathcal{T}_Q$, apply $Q_{k+1}=\mathcal{T}_Q Q_k$ ➔ $Q_k\to Q^*$, yet $\mathcal{T}_Q$ hard to approximate by samples…
2) Define the Z-function and its corresponding Bellman Operator $\mathcal{T}_Z$
3) Approximate T_Z with the sample-based estimation $\widehat{\mathcal{T}_Z}$.

$$[\mathcal{T}_Q Q](s,a):=r(s,a)-\gamma\beta^{-1}\log\mathbb{E}_{s'\sim\mathbb{P}_{s,a}}e^{-\beta\max_{a'}Q(s',a')}$$
$$=: [r(s,a)-\gamma\beta^{-1}\log Z(s,a)$$

$\Downarrow$

$$[\mathcal{T}_Q[\mathcal{T}_Q Q]](s,a)=r(s,a)-\gamma\beta^{-1}\log\underbrace{\mathbb{E}_{s'\sim\mathbb{P}_{s,a}}\left[e^{-\beta\max_{a'}(r(s',a')-\gamma\beta^{-1}\log Z(s',a'))}\right]}_{[\mathcal{T}_Z Z](s,a)}$$

finite function class $\mathscr{F}$
offline dataset $\mathcal{D}\sim\mu$ $\Downarrow$

$$[\widehat{\mathcal{T}_Z}Z]=\arg\min_{Z'\in\mathscr{F}}\underbrace{\frac{1}{|\mathcal{D}|}\sum_{(s,a,s',a')\in\mathcal{D}}\left[Z'(s,a)-e^{-\beta\max_{a'}\left(r(s',a')-\gamma\beta^{-1}\log Z(s',a')\right)}\right]^2}_{\widehat{\mathcal{L}}(Z',Z)}$$
$$\approx[\mathcal{T}_Z Z](s,a)$$

Algorithm: $Z_{k+1}\leftarrow\widehat{\mathcal{T}_Z}Z_k,\quad \pi_k\leftarrow\arg\max_a\left[r(s,a)-\gamma\beta^{-1}\log Z_k(s,a)\right]$

Convergence: under mild regularity conditions, with probability at least $1-\delta$:

$$\mathbb{E}_{s_0\sim\rho}[V^*(s_0)-V^{\pi_K}(s_0)]\le\frac{2\gamma^K}{(1-\gamma)^2}+\gamma\beta^{-1}e^{\frac{\beta}{1-\gamma}}\frac{2C}{(1-\gamma)^2}\left(4\sqrt{\frac{2\log(|\mathscr{F}|)}{N}}+5\sqrt{\frac{2\log(\frac{8}{\delta})}{N}}+\epsilon_{\mathscr{F}}\right)$$

Bellman contraction — *statistical error* — *function approximation*

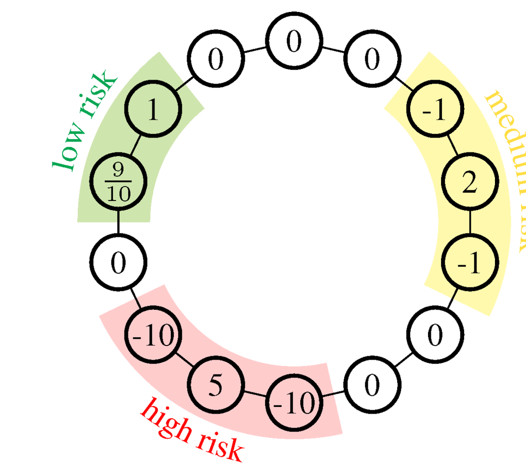
Numerical Simulations

Setting: an n-state environment, $\mathcal{S}=[n]$, $\mathcal{A}=\{\leftarrow,\downarrow,\rightarrow\}$.

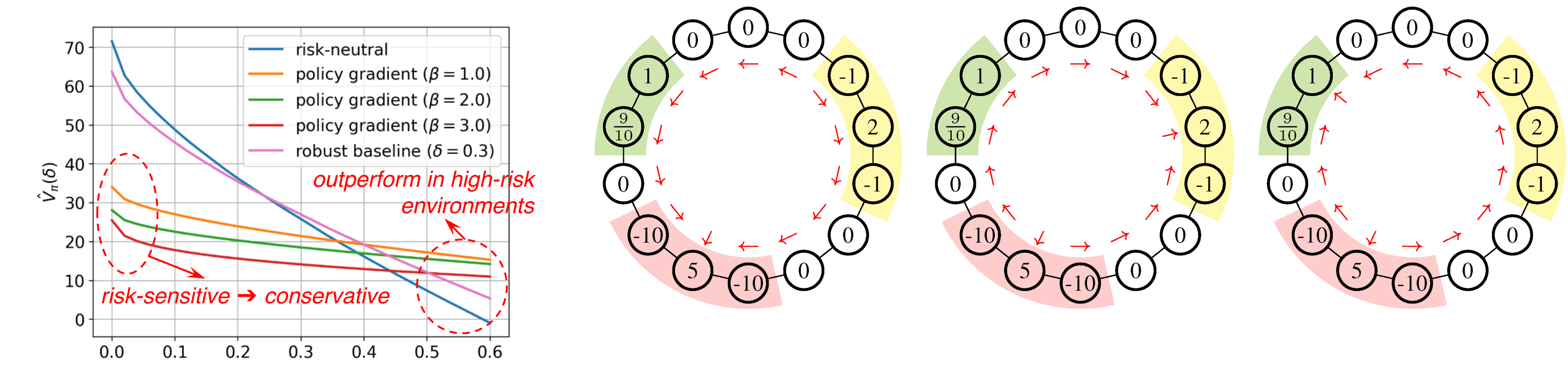
$$\mathbb{P}(s'|s,a)=\begin{cases}\alpha & s'=(s+a\pm1)\mod n\\ 1-2\alpha & s'=(s+a)\mod n\\ 0 & \text{otherwise}\end{cases},\quad \alpha\in(0,\tfrac{1}{2}).$$

Metrics: optimality gap: $\mathbb{E}_{s_0\sim\rho}[V^*(s_0)-V^{\pi}(s_0)]$;

KL-robust value: $\widehat{V}^{\pi}(\delta):=\inf_{P\in\mathcal{P}_\delta}\mathbb{E}^{\pi,P}\left[\sum_{t=0}^{H}\gamma^t r(s_t,a_t)\right]$.

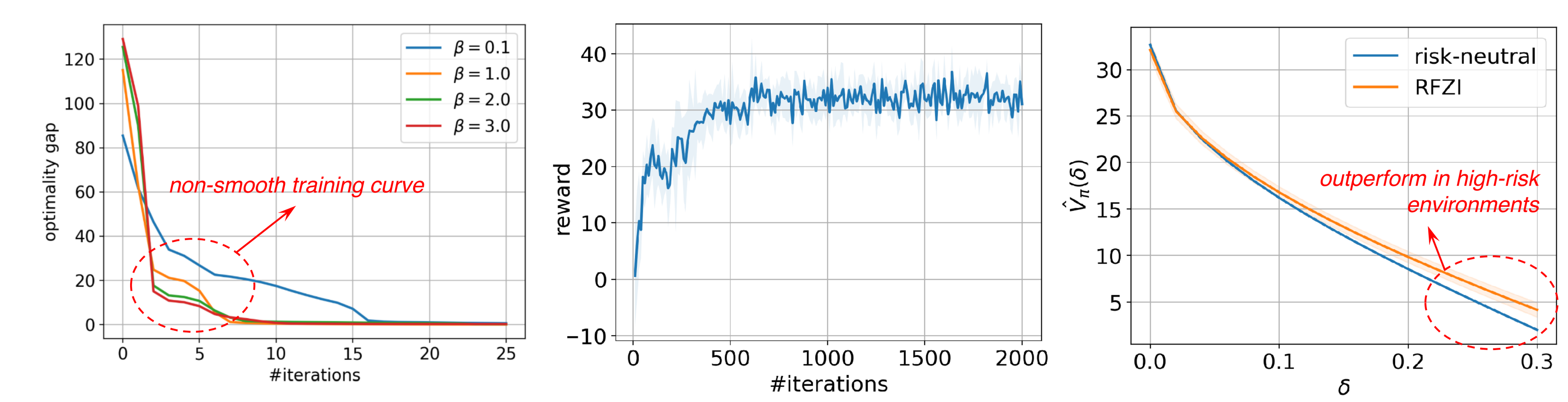


Planning: Soft-robust Policy Gradient



(a) $\alpha=0.01, \beta=0.1$ *low risk, low risk-sensitivity*
(b) $\alpha=0.01, \beta=1.0$ *low risk, high risk-sensitivity*
(c) $\alpha=0.15, \beta=1.0$ *high risk, high risk-sensitivity*

Learning: RFZI Algorithm



Take-away Messages

Main contributions:

(1) Risk-sensitive MDPs and soft robust MDPs are equivalent.
(2) Show the iteration complexity of soft-robust Policy Gradient for planning.
(3) Propose a value-based RFZI algorithm for offline learning in entropic-risk-sensitive MDPs.

Future work:

(1) Extend the policy gradient algorithm to work with the learning setting.
(2) Generalize the RFZI algorithm for a wider range of risk measures.
(3) Settle the scalability concerns to support large state-action spaces.
(4) …