Efficient Risk-Averse Reinforcement Learning (RL)

Ido Greenberg[1], Yinlam Chow[2], Mohammad Ghavamzadeh[2], Shie Mannor[1,3]

NeurIPS, 2022

[1]Technion, Israel; [2]Google research; [3]Nvidia research

TECHNION Israel Institute of Technology

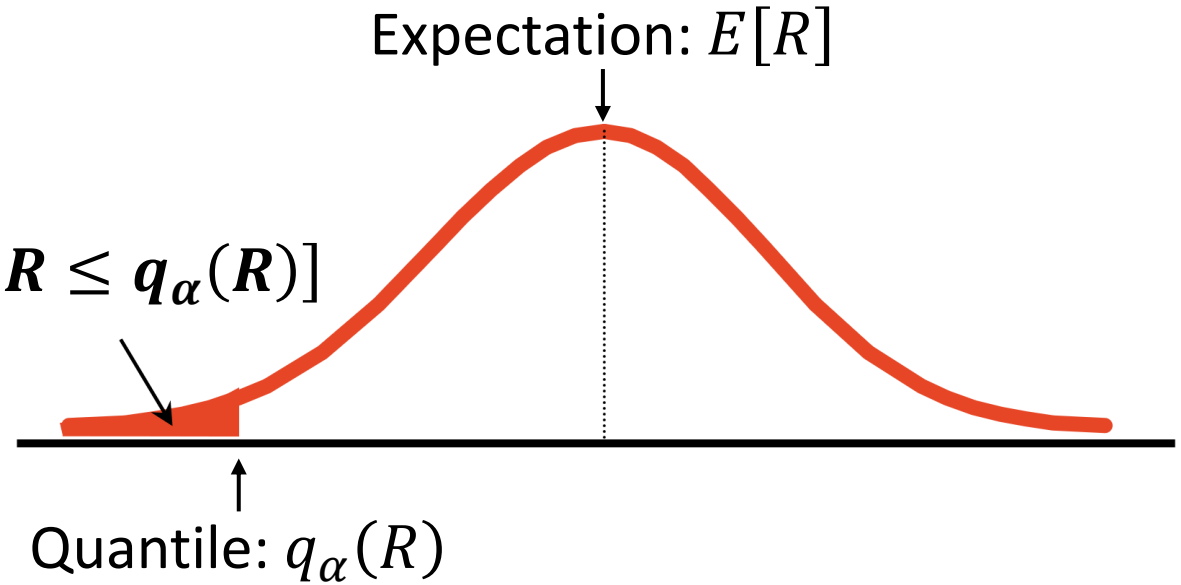
Google Research

NVIDIA

Risk-Averse Reinforcement Learning

- Instead of ***expected*** return – optimize ***Conditional Value at Risk***
  - Average over the α-tail (α worst quantiles)
  - (synonyms: CVaR, AVaR , ES, ETL)

$$\boldsymbol{CVaR_{\alpha}(R) = E[R|R \leq q_{\alpha}(R)]}$$



Risk-Averse Policy Gradient

- Optimizing CVaR using Policy Gradient (**CVaR-PG**):
  - while true:
    - roll N episodes
    - take the worst αN episodes
    - optimize using a standard PG step:

$$\Delta_\theta \propto \frac{1}{\alpha N} \sum_{i=1}^{N} \mathbf{1}_{R_i \leq q_\alpha(R)} \left(R_i - q_\alpha(R)\right) \sum_{t=0}^{T} \nabla_\theta \, log \, \pi_\theta(a_t|s_t)$$

(Tamar et al., 2015)

Example: The Guarded Maze

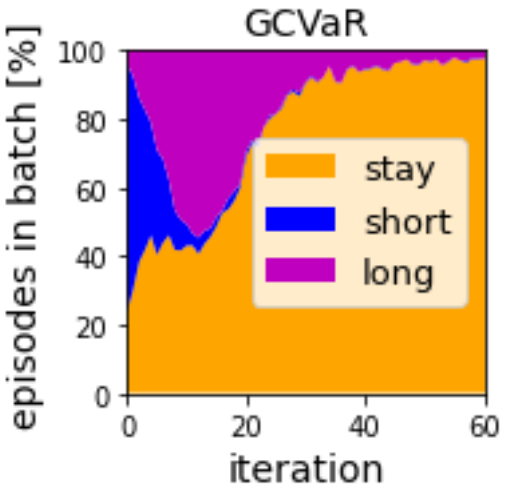
- Goal: get to the target fast
  - Red zone: random guard-bribery cost
  - Small cost every time step
- Paths:
  - Mean-optimal: short path
  - CVaR-optimal ($\alpha = 5\%$): long (safe) path

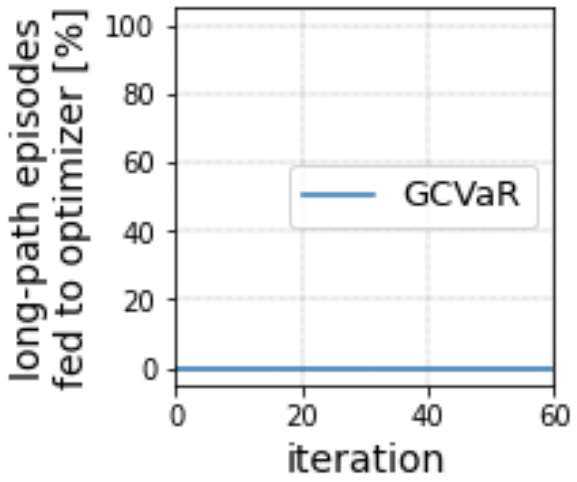
CVaR-PG on the Guarded Maze

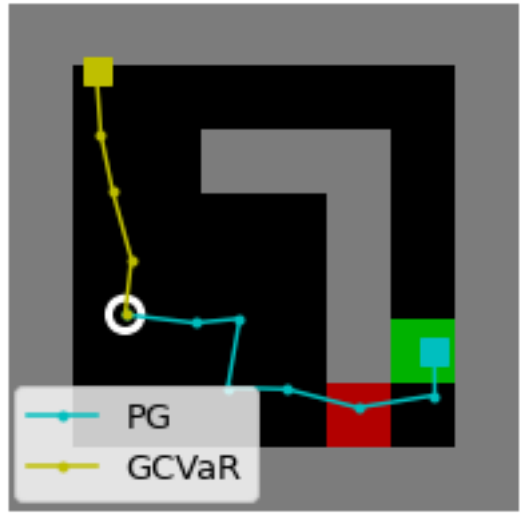
- Staying is learned as better than short path
- Long path is never on worst αN episodes

➔ never fed to the optimizer ➔ never learned

Blindness to Success



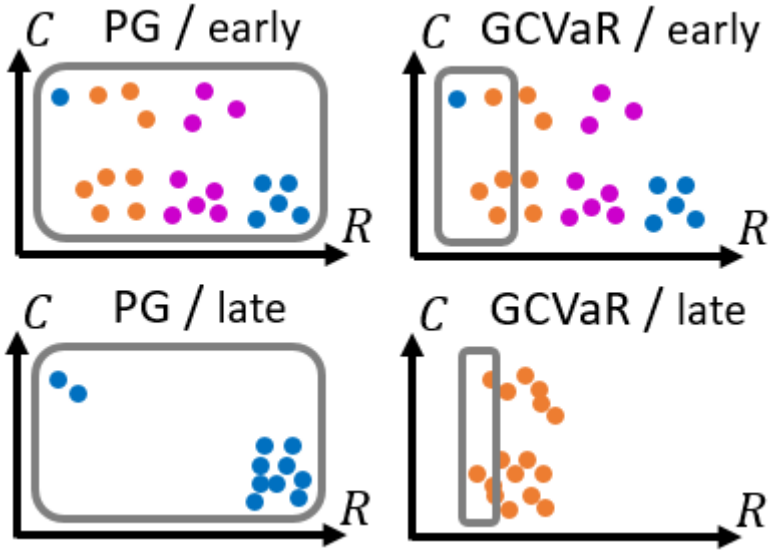


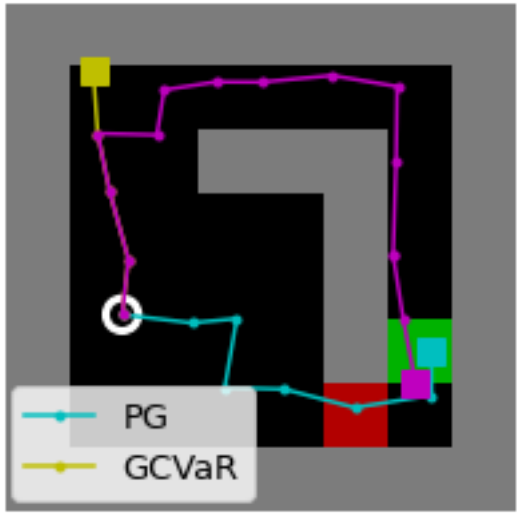


(GCVaR = implementation of CVaR-PG)

Mean-PG vs. CVaR-PG

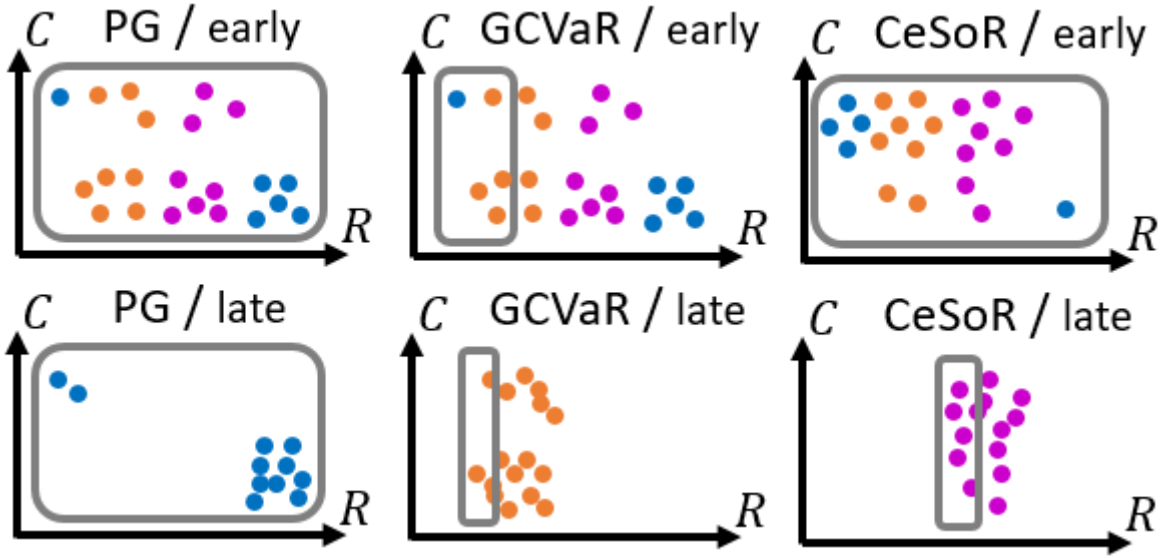
- *C* = guard cost
  - (in general: environment conditions)
- The PG training process:

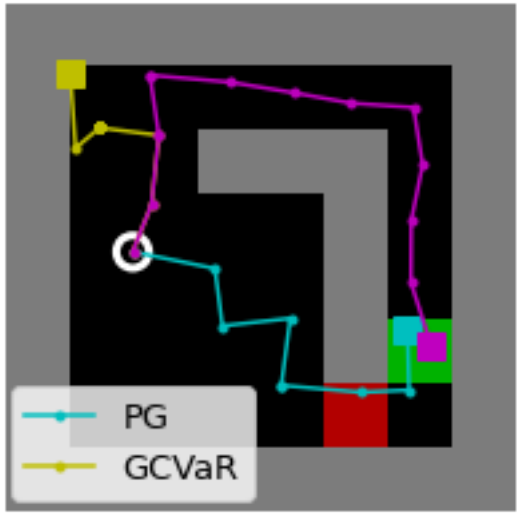




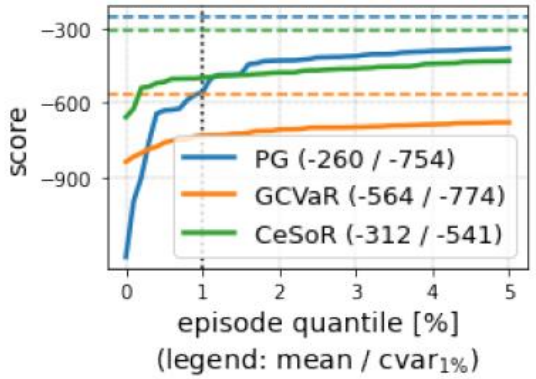
Mean-PG vs. CVaR-PG

- **Idea: focus on top (hard conditions), not left (bad strategy)**
  - *Top* – **Cross Entropy Method**: Learn which *C*'s are more difficult, and over-sample them
  - *Not left* – **Soft Risk**: In the beginning, don't limit to bad returns
  - **CeSoR** = Cross entropy Soft Risk

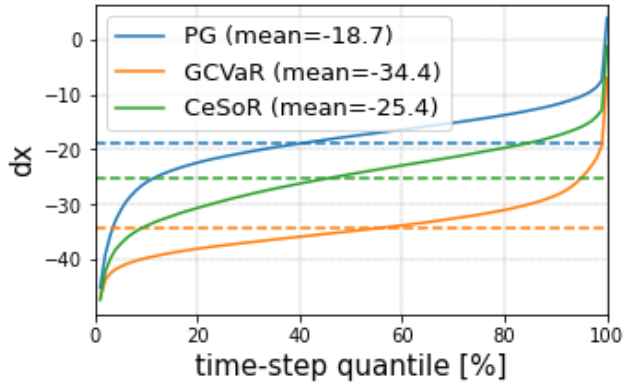




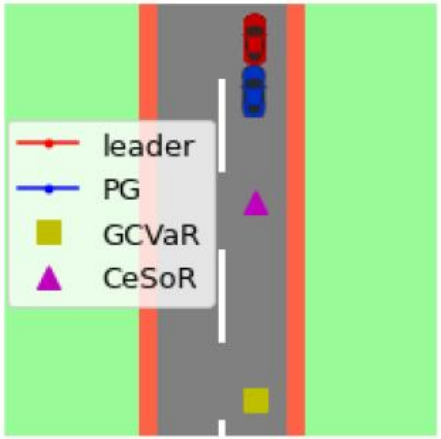
Driving: Intuitive Risk-Averse Policy



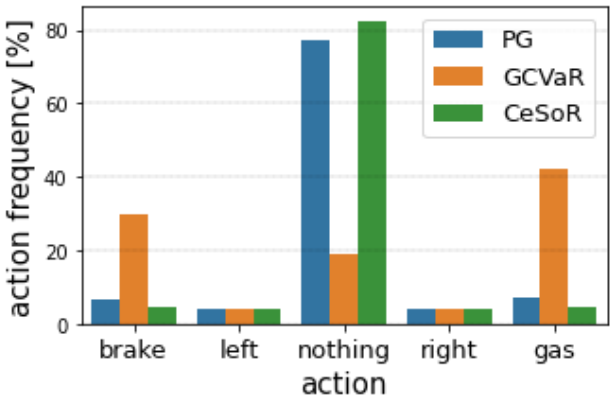
(b) Driving Game



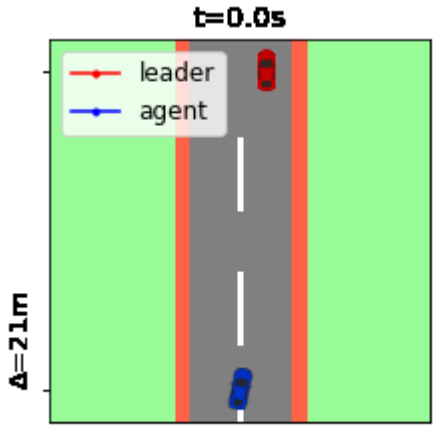
Keeps slightly more distance than vanilla
PG: sufficient to prevent all accidents



(e) An accident of PG. In the same situation, CeSoR maintains a safe margin from the leader, without losing as much distance as GCVaR.

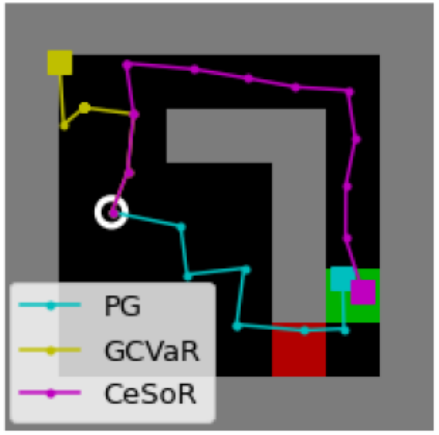


Slightly less use of gas & brakes

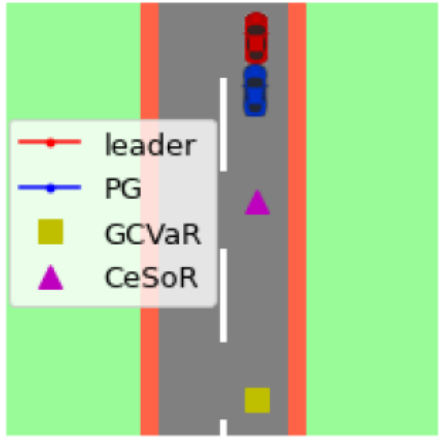


Summary

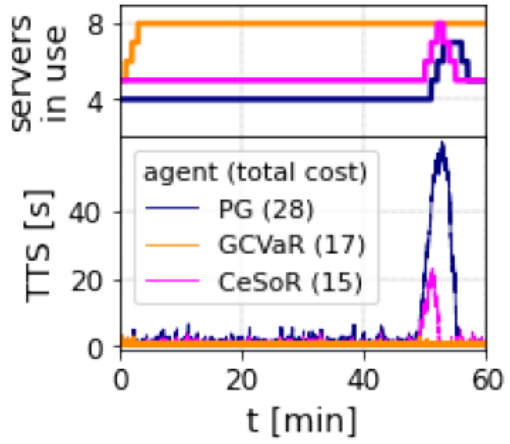
- Problem: optimize the CVaR risk-measure in RL
- Standard methods: optimize wrt worst episodes
  - Small part of data ➔ sample inefficient
  - Worst part of data ➔ blindness to success
- **CeSoR**: optimize wrt hard conditions (CEM), not bad strategies (soft risk)



(d) CeSoR learns to avoid the risk (red) and take the long path to the target (green), whereas GCVaR suffers from *blindness to success*.



(e) An accident of PG. In the same situation, CeSoR maintains a safe margin from the leader, without losing as much distance as GCVaR.



(f) CeSoR handles the exceptional peak in user-requests without paying for as many servers as GCVaR, leading to a higher total value.