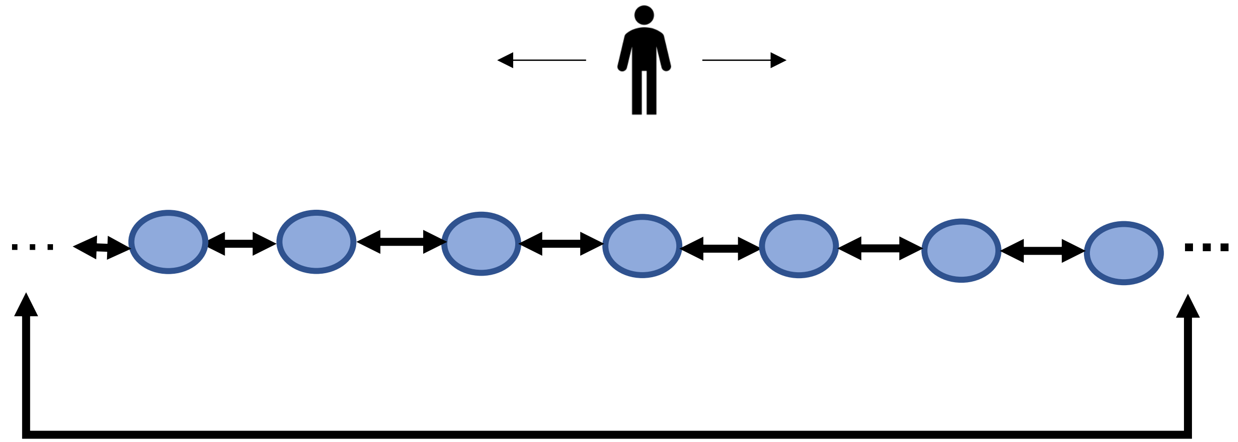
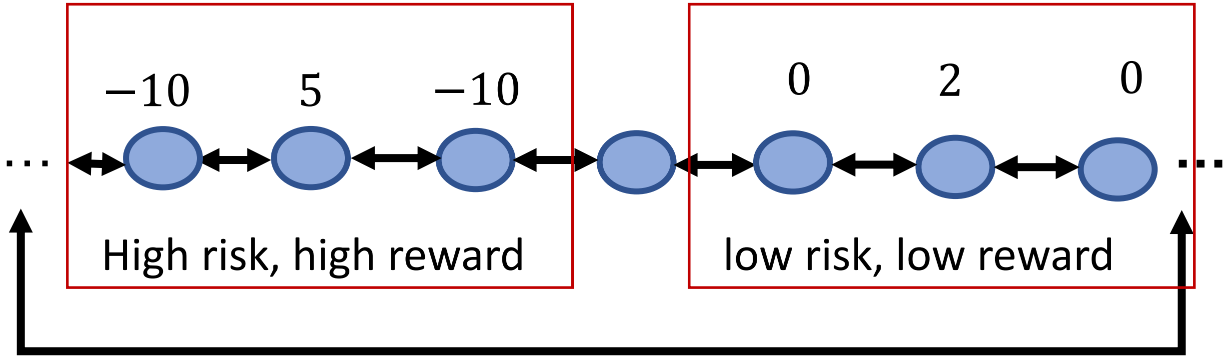
A toy example – travel on a circle



Agent's action:
Left/right/stay

State-transition:
Agent's action + random perturbation ϵ

Objective:
Collect as many reward as possible



Classic: $\mathrm{M}ax_{\pi}.\ \ E_{\pi} \sum_{h=0}^{H} r(s_h^{\pi})$

Risk-sensitive: $\mathrm{M}ax_{\pi}.\ \ E_{\pi} \exp(\beta \sum_{h=0}^{H} r(s_h^{\pi}))$

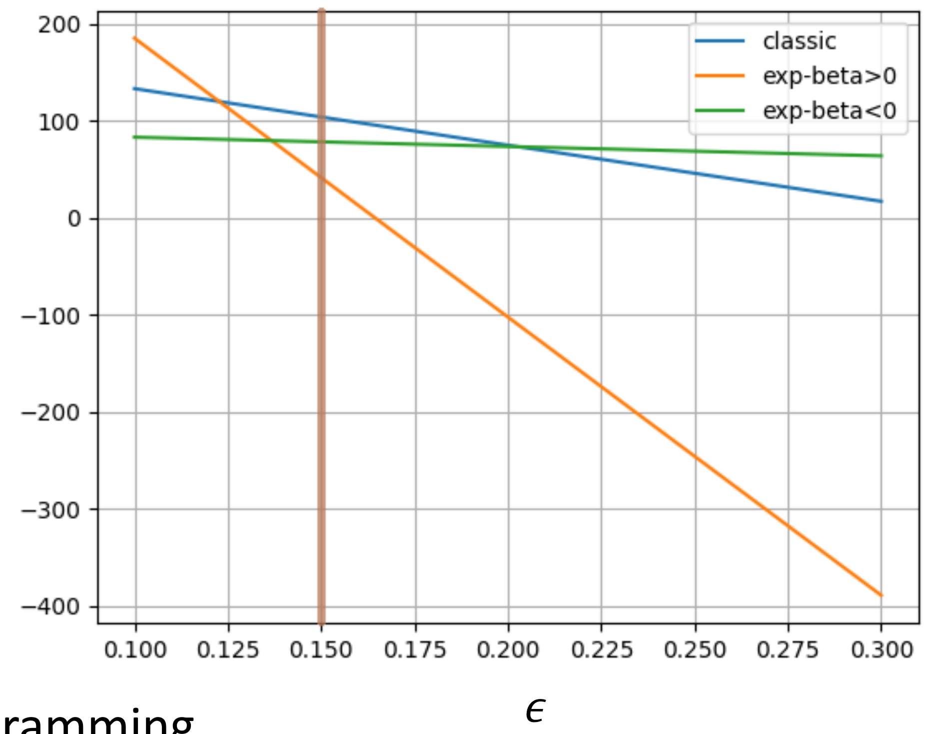
Numerical simulation result

- Reward list:

[0,-10,5,-10,0,1,1,0, 0,0,-1,2,-1,0]

High risk Low risk Medium risk

- Three different objective functions:
  - Classic (Risk Neutral)
  - Risk-averse (beta < 0)
  - Risk-seeking (beta > 0)
- Implementation: calculate the optimal policy for the three different objectives under $\epsilon_0 = .15$, then calculate The performance for different $\epsilon = .1, .15, .2, \ldots, .3$



- Algorithm for finding the optimal policy: Dynamical programming

Robustness

- Criteria for testing robustness

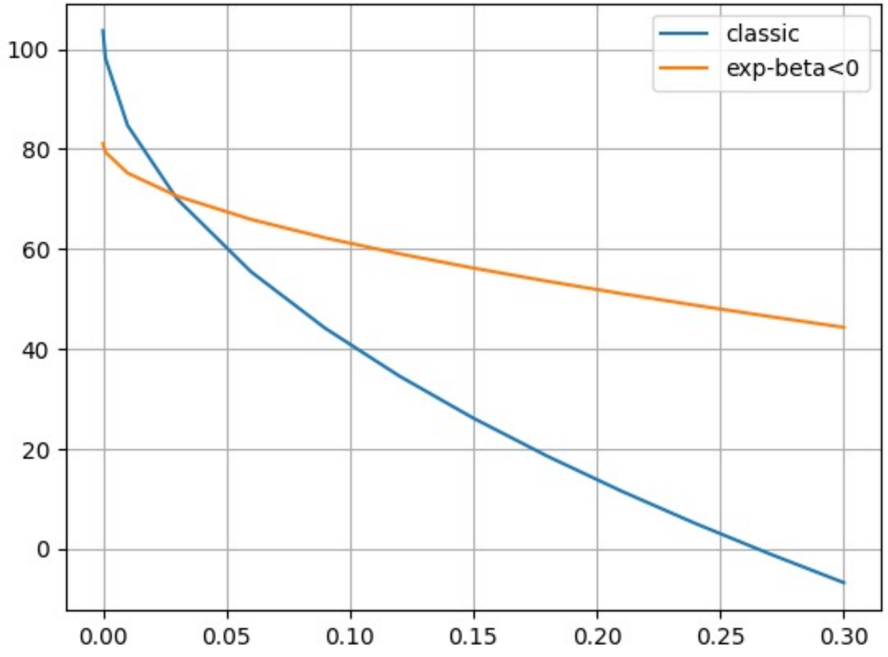
$$f(\pi) = \inf_{P \in \mathcal{P}} \mathbb{E}^{P,\pi}\left[\sum_{t=0}^{\infty} \lambda^t r(s_t, a_t, s_{t+1}) \,\middle|\, s_0 \sim p_0\right]$$

where the ambiguity set is chosen as:

$$\mathcal{P} := \{\widetilde{P} : \mathsf{KL}(\widetilde{P}(\cdot|s,a)||P(\cdot|s,a)) \leq \delta\}$$

Note that at different nodes the perturbation error can be different in this choice of ambiguity set

$f(\pi)$



δ

- Additional figure (with risk seeking):