When Maximum Entropy Misleads Policy Optimization

Ruipeng Zhang, Ya-Chien Chang, Sicun Gao

UC San Diego

Background

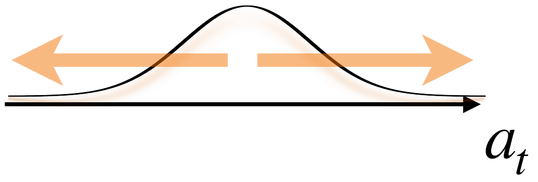
Standard RL v.s. Maximum Entropy RL

- **Objective**

Standard RL $J(\pi) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)\right]$

MaxEnt RL $J_{soft}(\pi) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \left(r(s_t, a_t) + \alpha H(\pi(\cdot \mid s_t))\right)\right]$

$H = -\mathbb{E}_{a\sim\pi(\cdot|s)}[\log \pi(a \mid s)]$



Exploitation ***Exploration***

Standard RL | **MaxEnt RL** | **Uniform**

$\pi^*(a \mid s) = \mathbf{1}\{a = \arg\max_a Q(s, a)\}$

$0 \longleftarrow \alpha \longrightarrow \infty$

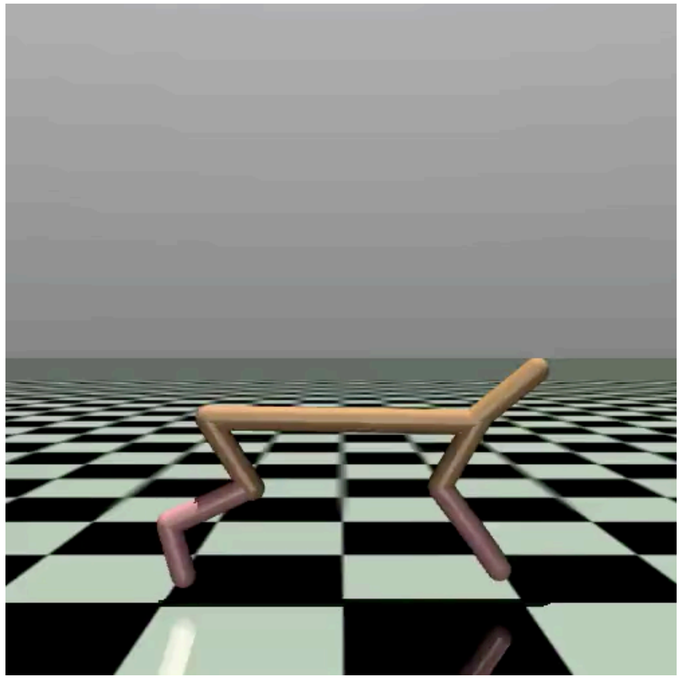
$\pi^*_{soft}(a \mid s) = \frac{\exp\left(\frac{Q(s,a)}{\alpha}\right)}{\sum_{a'} \exp\left(\frac{Q(s,a')}{\alpha}\right)}$

$\pi_{uniform}(a \mid s) \to \frac{1}{|A|}$

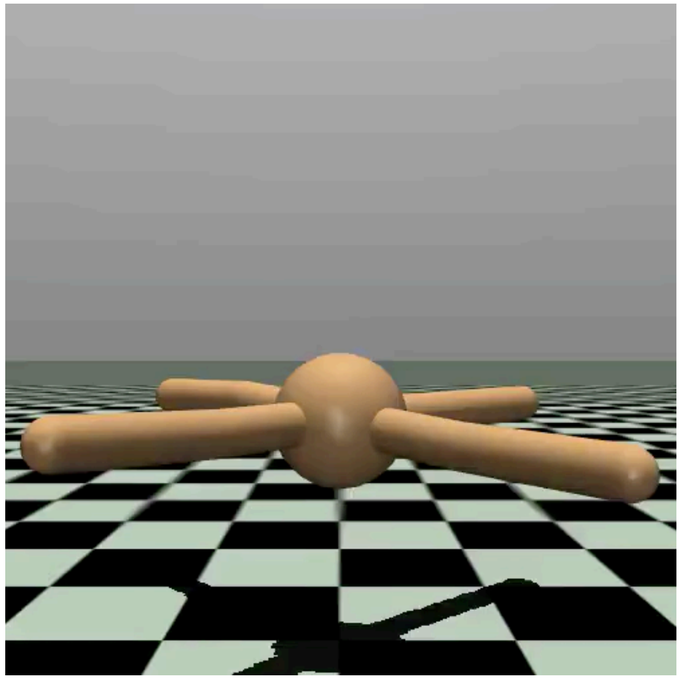
MaxEnt Policy Optimization Strategies

- **Soft Actor Critic**

Control Robots in Mujoco Simulation Environments

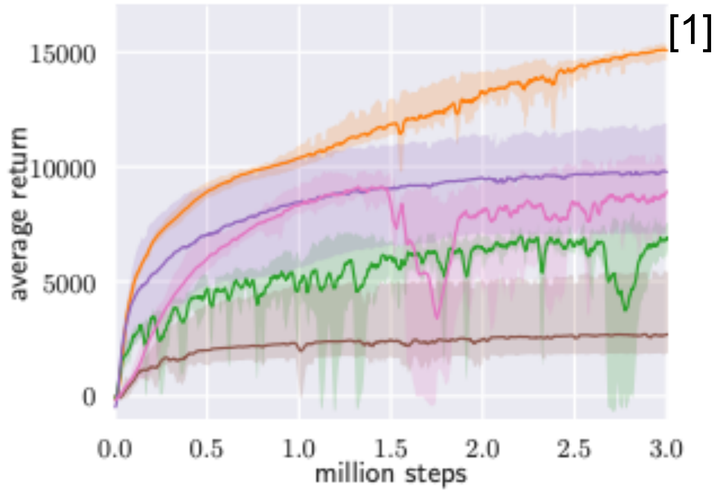


HalfCheetah



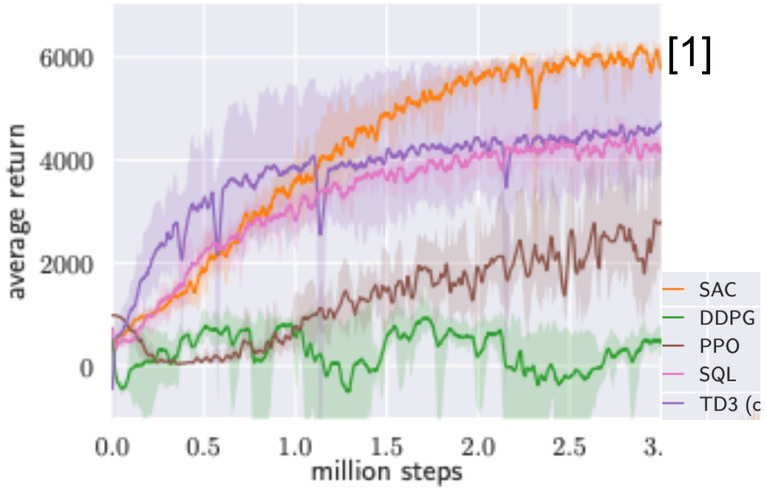
Ant

HalfCheetah



[1]

Ant



[1]

[1] Haarnoja et al. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor, ICML 2018

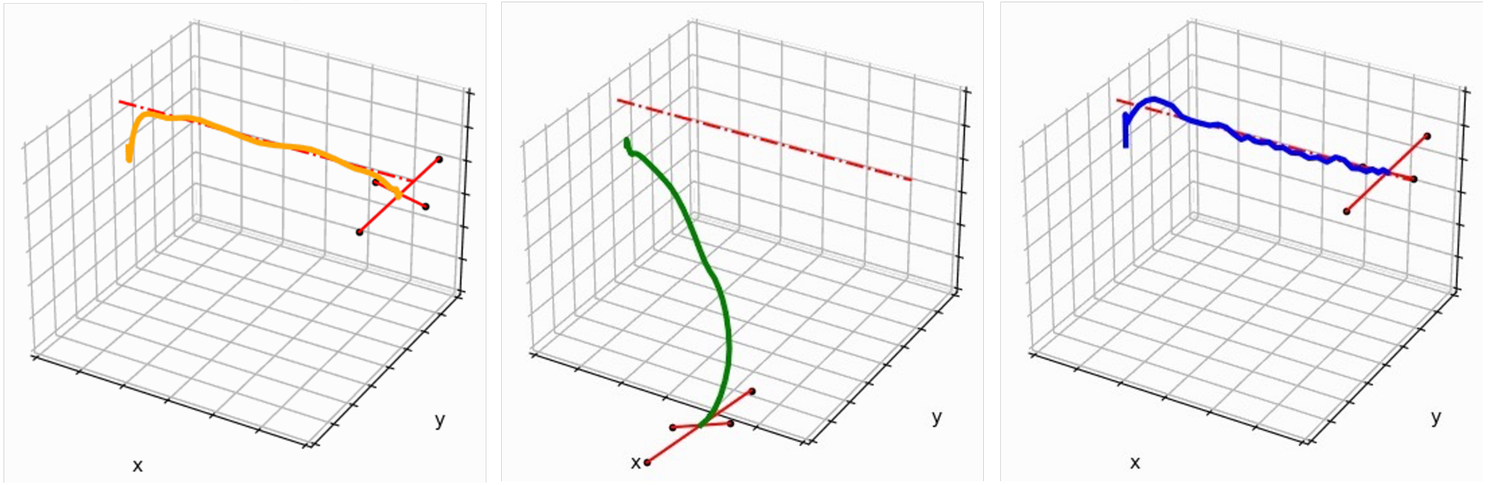
Is entropy always superior?

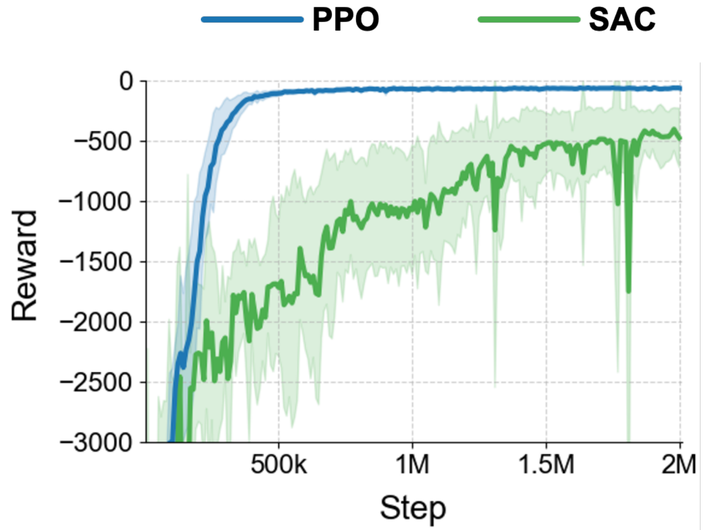
Failures

Standard RL (PPO) v.s. MaxEnt-RL (SAC) algorithms

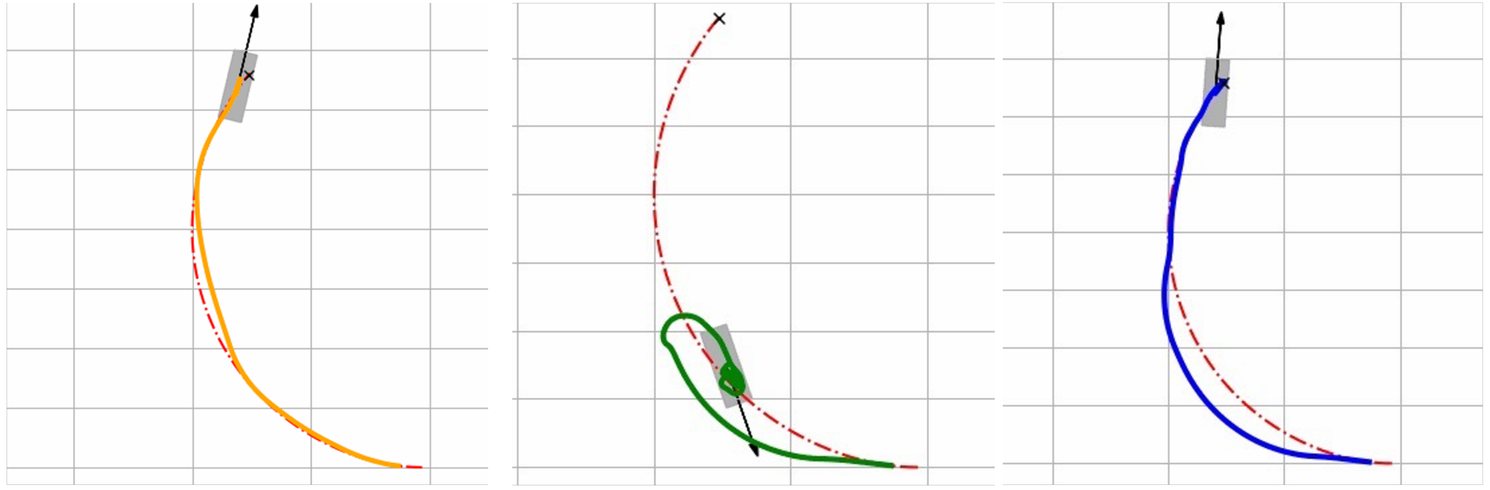
SAC-Simplified | **SAC-Realistic** | **PPO-Realistic**

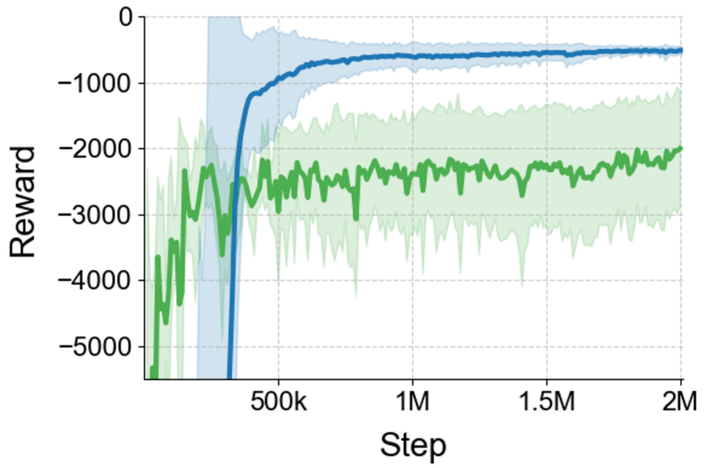
Quadrotor





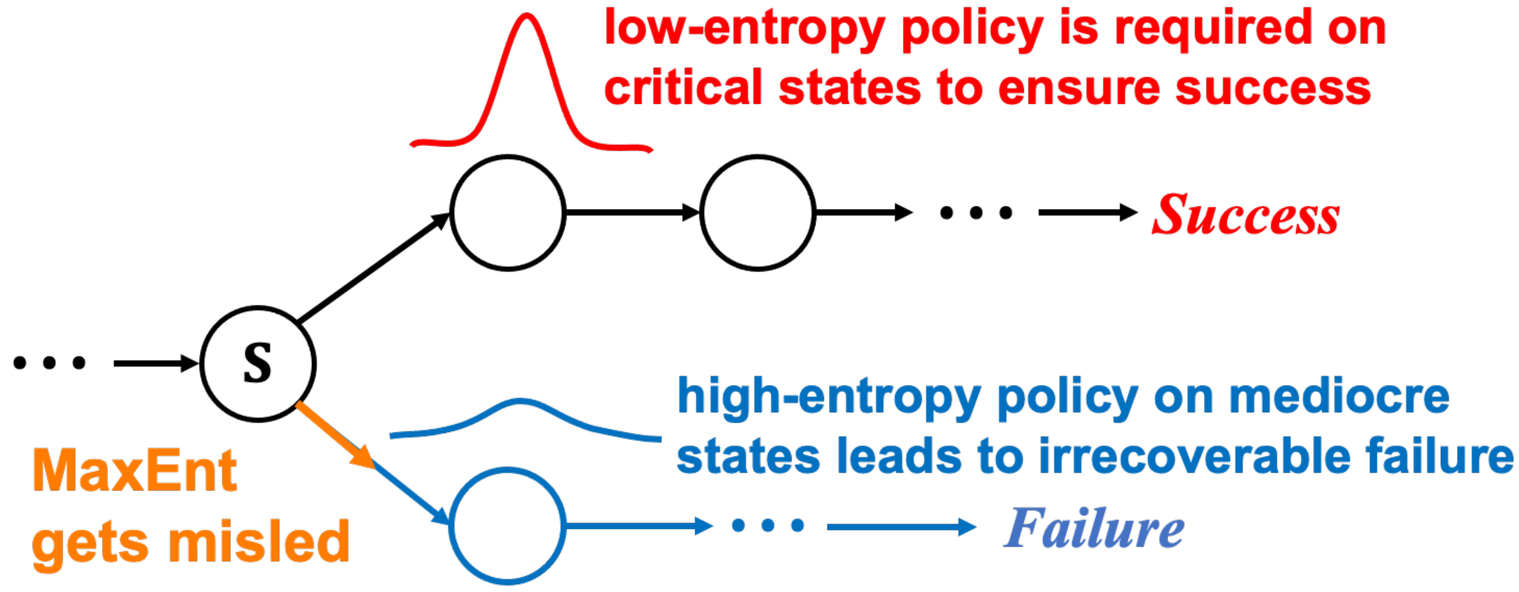
Vehicle



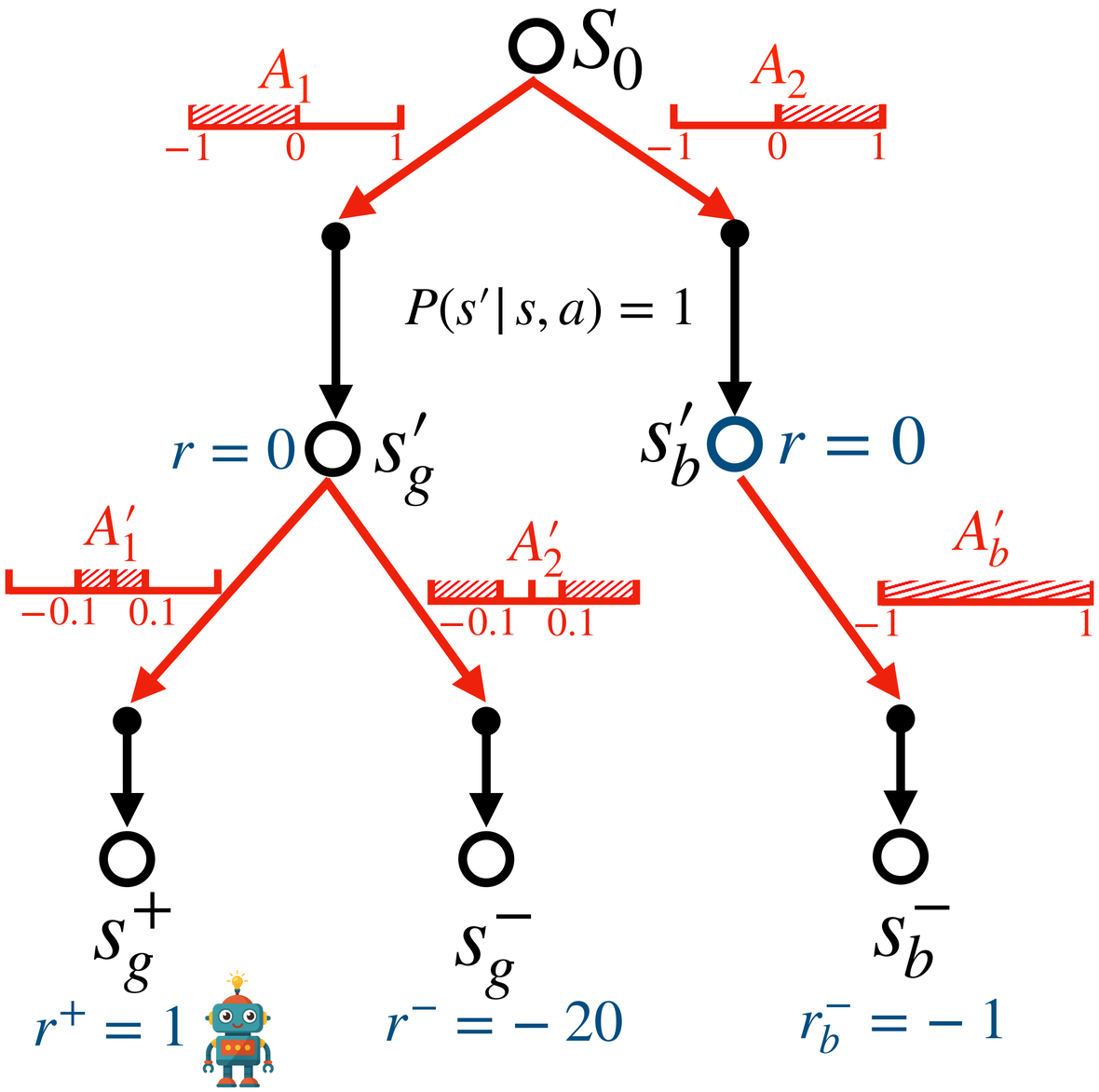


Why Failures?

Policy optimization can be misled by entropy



Toy Example

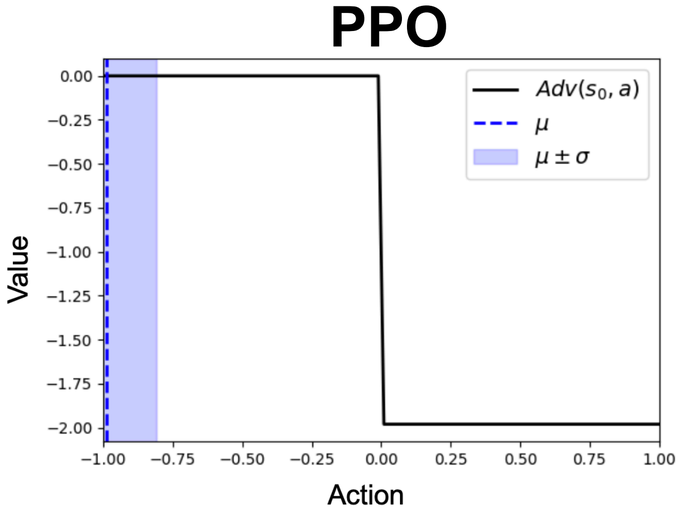


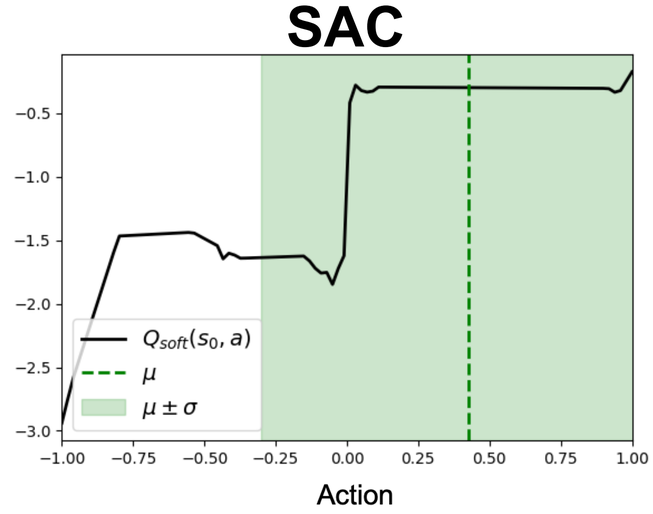
Standard RL

$$\pi^*(a \in A_1 \,|\, s_0) > \pi^*(a \in A_2 \,|\, s_0)$$

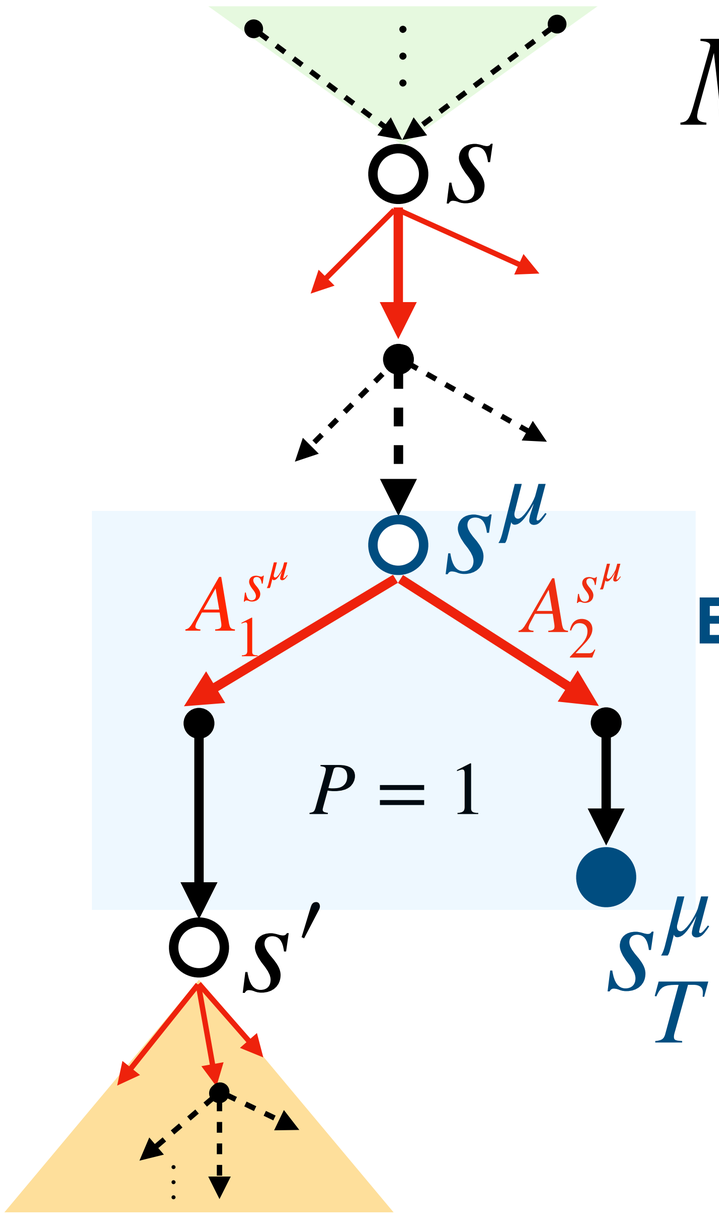
MaxEnt RL

$$\pi^*_{soft}(a \in A_1 \,|\, s_0) < \pi^*_{soft}(a \in A_2 \,|\, s_0)$$





Entropic Bifurcation Extension



Entropic Bifurcation Extension

Theorem

Bifurcation Extension Misleads MaxEnt RL

We can construct $\hat{M}$ s.t.

- The optimal policy on original M doesn't change.
- MaxEnt-optimal policy at s after entropy bifurcation extension can follow an arbitrary distribution $\pi(\cdot \mid s)$ over the actions by choosing $|A_1^{s^{\mu}}|, |A_2^{s^{\mu}}|, r(s_{\mu}^{T})$

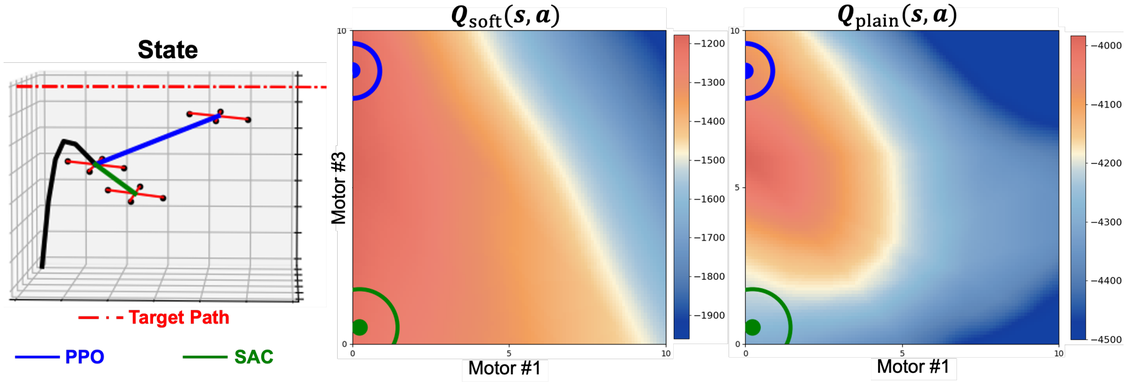
How Entropy Misleads?

- **Method: Investigating Q-value Landscapes**

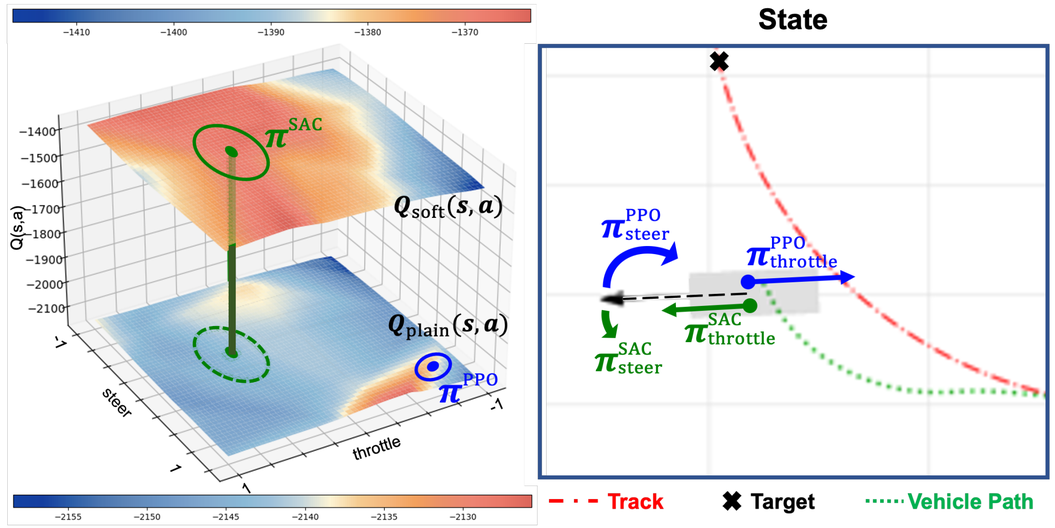
Train soft-Q (used for optimization) and plain-Q networks with SAC

$Q_{soft}(s_t, a_t)$ $\quad \mathcal{T}^{\pi}Q_{soft}(s_t, a_t) = r(s_t, a_t) + \gamma \mathbb{E}_{s_{t+1}, a_{t+1}}[Q_{soft}(s_{t+1}, a_{t+1}) - \alpha \log \pi(a_{t+1} \mid s_{t+1})]]$

$Q_{plain}(s_t, a_t)$ $\quad \mathcal{T}^{\pi}Q_{plain}(s_t, a_t) = r(s_t, a_t) + \gamma \mathbb{E}_{s_{t+1}, a_{t+1}}[Q_{plain}(s_{t+1}, a_{t+1})]]$



Quadrotor

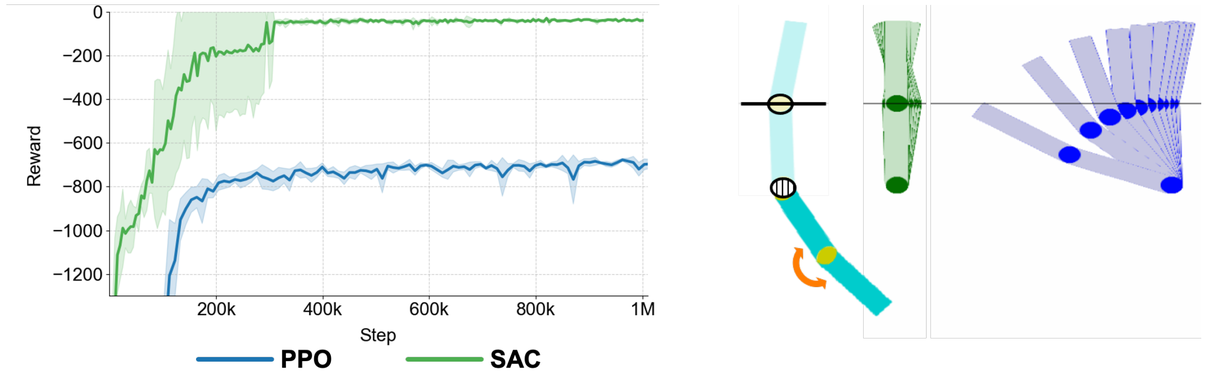


Vehicle

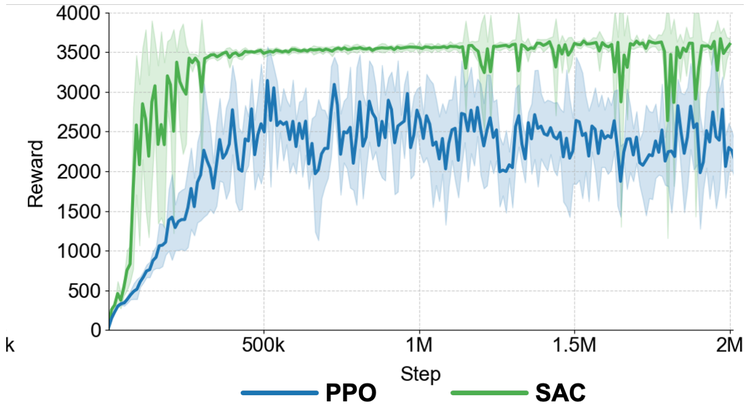
Benefits of Misleading Landscapes

- **Entropy smooths the Q landscapes**

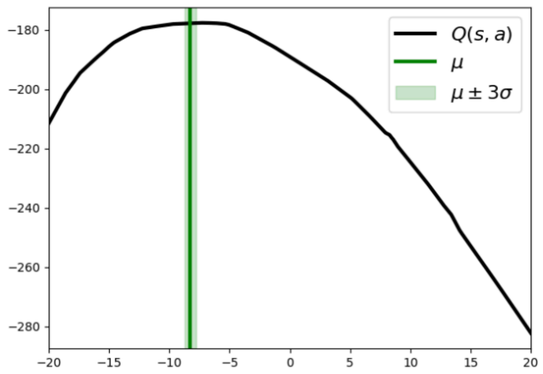
Acrobot



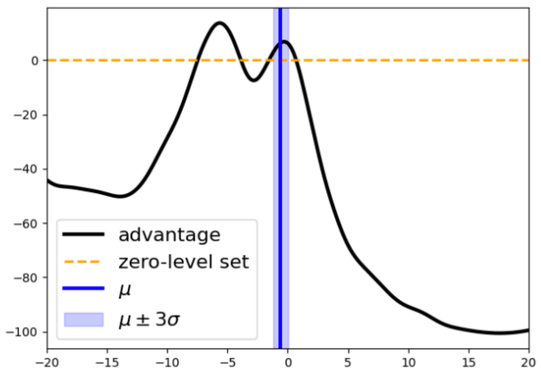
Hopper

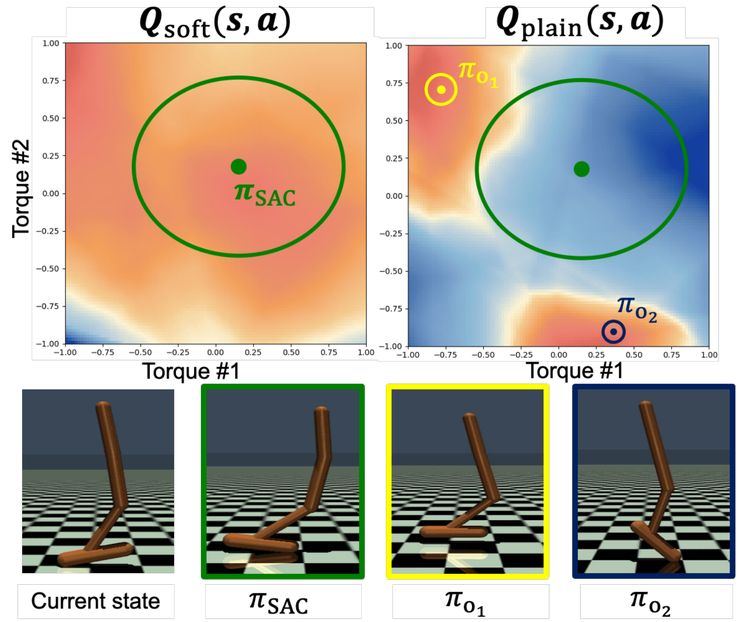


SAC



PPO





Can Adaptive Entropy improve?

- **SAC-AdaEnt: Adaptively using Q_{soft} and Q_{plain} as target**

Algorithm 1 SAC with Adaptive Entropy (SAC-AdaEnt)

Initialize: Actor network π_θ, Q networks and their paired target networks for Q-target with/without entropy $\phi_1, \phi_2, \phi_{targ,1}, \phi_{targ,2}$ (w/ entropy), $\phi'_1, \phi'_2, \phi'_{targ,1}, \phi'_{targ,2}$ (w/o entropy), replay buffer $\mathcal{D}$, similarity threshold ϵ

for each training step **do**

Sample action $a_t \sim \pi_\theta(a_t|s_t)$ and observe s_{t+1}, r_t

Store (s_t, a_t, r_t, s_{t+1}) in replay buffer $\mathcal{D}$

for each gradient update step **do**

Sample minibatch of transitions (s, a, r, s') from $\mathcal{D}$

Compute target value:

$$y = r + \gamma \left(\min_{i=1,2} \hat{Q}_{\phi_i}(s', a') - \alpha \log \pi_\theta(a'|s') \right), \; y' = r + \gamma \left(\min_{i=1,2} \hat{Q}_{\phi'_i}(s', a') \right)$$

Update Q networks:

$$\phi_i \leftarrow \phi_i - \eta_Q \nabla_{\phi_i} \frac{1}{N} \sum_{n=1}^{N} (\phi_i(s,a) - y)^2, \; \phi'_i \leftarrow \phi'_i - \eta_Q \nabla_{\phi'_i} \frac{1}{N} \sum_{n=1}^{N} (\phi'_i(s,a) - y')^2$$

For each s, sample actions using current policy $A_s = \{a_s | a_s \sim \pi_\theta(\cdot|s)\}$

Compute similarity score:

$$\text{sim}(\mathbf{Q}, \mathbf{Q}') = \frac{\mathbf{Q}(s) \cdot \mathbf{Q}'(s)}{\|\mathbf{Q}(s)\| \|\mathbf{Q}'(s)\|}, \text{where } \mathbf{Q}(s) = \left[\min_{i=1,2} \hat{Q}_{\phi_i}(s, a_s) \right]_{a_s \sim \pi_\theta(a|s)}, \; \mathbf{Q}'(s) = \left[\min_{i=1,2} \hat{Q}_{\phi'_i}(s, a_s) \right]_{a_s \sim \pi_\theta(a|s)}$$

Update actor policy using reparameterization trick:

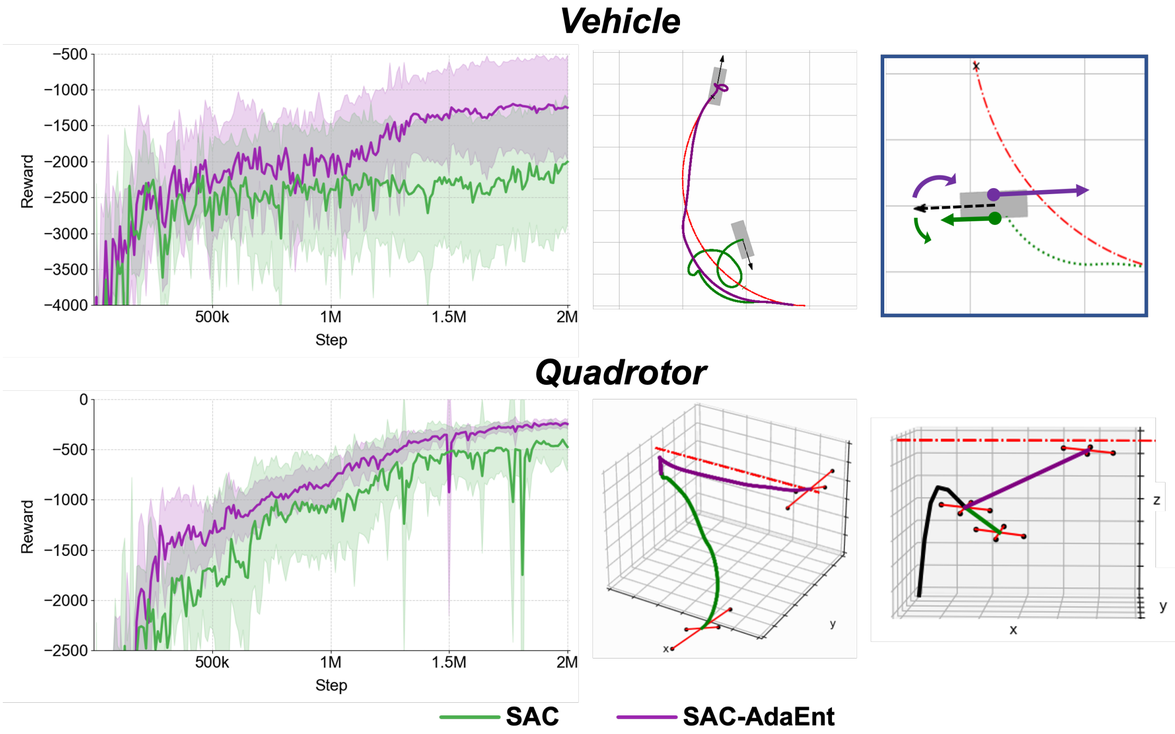
$$\theta \leftarrow \theta - \eta_\pi \nabla_\theta \mathbb{E}_{s \sim \mathcal{D}, a \sim \pi_\theta} \begin{cases} \alpha \log \pi_\theta(a|s) - Q_{\phi_1}(s,a), & \text{if sim}(\mathbf{Q}, \mathbf{Q}') > \epsilon \\ \alpha \log \pi_\theta(a|s) - Q_{\phi'_1}(s,a), & \text{otherwise} \end{cases}$$

Update target networks:

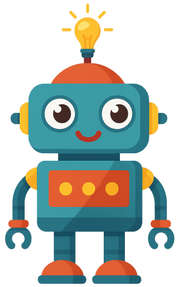
$$\hat{Q}_{\phi_i} \leftarrow \tau Q_{\phi_i} + (1-\tau)\hat{Q}_{\phi_i}, \; \hat{Q}_{\phi'_i} \leftarrow \tau Q_{\phi'_i} + (1-\tau)\hat{Q}_{\phi'_i}$$

end for

end for



Summary



- We identify a **critical failure mode** of MaxEnt RL (e.g., SAC) in precision-demanding control tasks.
- We propose **Entropic Bifurcation Extension,** a theoretical framework explaining how entropy can mislead policy optimization.
- We analyze the **Q-value landscapes** of soft and true Q-values, and visually demonstrate how entropy causes the policy to favor suboptimal actions.
- We propose **-AdaEnt,** an adaptive entropy strategy that outperforms MaxEnt RL in tasks where it fails.

Paper