

Soft Actor-Critic

Maximum Entropy Reinforcement Learning

T. Haarnoja, et al., “Reinforcement Learning with Deep Energy-Based Policies”, ICML 2017

T. Haarnoja, et, al., “Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor”, ICML 2018

T. Haarnoja, et, al., “Soft Actor-Critic Algorithms and Applications”, arXiv preprint 2018

Presented by Dongmin Lee

October 27, 2019

Outline

1. Reinforcement Learning
 - Markov Decision Processes (MDPs)
 - Bellman Equation
2. Maximum Entropy Reinforcement Learning
 - Soft MDPs
 - Soft Bellman Equation
3. From Soft Policy Iteration to Soft Actor-Critic
 - Soft Policy Iteration
 - Soft Actor-Critic
4. Results

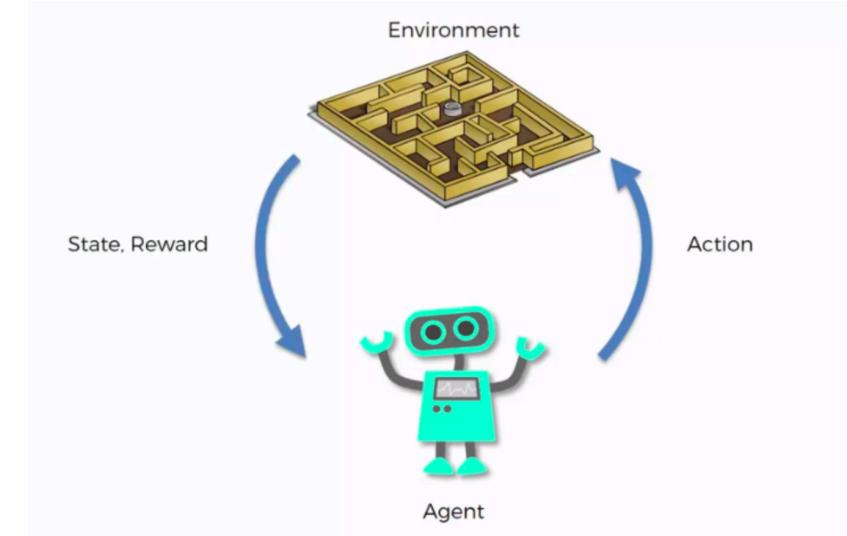
Reinforcement Learning

- State of the environment: $s_t \in S$
- Action determined by the agent: $a_t \in A$
- Reward evaluated the benefit of state and action: $r_t \in R$
- Policy (mapping from state to action): π
 - Deterministic

$$\pi(s_t) = a_t$$

- Stochastic (randomized)

$$\pi(a|s) = \text{Prob}(a_t = a | s_t = s)$$



→ Policy is what we want to optimize!

Reinforcement Learning

What are the challenges of RL?

- Huge # of samples: millions
- Fast, stable learning
- Hyperparameter tuning
- Exploration
- Sparse reward signals
- Safety / reliability
- Simulator

Markov Decision Processes (MDPs)

A Markov decision process (MDP) is a tuple $\langle S, A, p, r, \gamma \rangle$, consisting of

- S : set of **states** (state space)
e.g., $S = \{1, \dots, n\}$ (discrete), $S = \mathbb{R}^n$ (continuous)
- A : set of **actions** (action space)
e.g., $A = \{1, \dots, m\}$ (discrete), $A = \mathbb{R}^m$ (continuous)
- p : state transition probability
 $p(s'|s, a) \triangleq \text{Prob}(s_{t+1} = s' | s_t = s, a_t = a)$
- r : **reward** function
 $r(s_t, a_t) = r_t$
- $\gamma \in (0,1]$: discount factor

Markov Decision Processes (MDPs)

The MDP problem

- To find an optimal policy that maximizes the expected cumulative reward:

$$\max_{\pi \in \Pi} \mathbb{E}_{\pi} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \right]$$

Value function

- The state value function $V^{\pi}(s)$ of a policy π is the expected return starting from state s under executing π :

$$V^{\pi}(s) \triangleq \mathbb{E}_{\pi} \left[\sum_{\tau=t}^{\infty} \gamma^{\tau-t} r(s_{\tau}, a_{\tau}) \mid s_t = s \right]$$

Q-function

- The action value function $Q^{\pi}(s, a)$ of a policy π is the expected return starting from state s , taking action a , and then following π :

$$Q^{\pi}(s, a) \triangleq \mathbb{E}_{\pi} \left[\sum_{\tau=t}^{\infty} \gamma^{\tau-t} r(s_{\tau}, a_{\tau}) \mid s_t = s, a_t = a \right]$$

Markov Decision Processes (MDPs)

The MDP problem

- To find an optimal policy that maximizes the expected cumulative reward:

$$\max_{\pi \in \Pi} \mathbb{E}_{\pi} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \right]$$

Optimal value function

- The optimal value function $V^*(s)$ is the maximum value function over all policies:

$$V^*(s) \triangleq \max_{\pi \in \Pi} V^{\pi}(s) = \max_{\pi \in \Pi} \mathbb{E}_{\pi} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) | s_t = s \right]$$

Optimal Q-function

- The optimal Q-function $Q^*(s, a)$ is the maximum Q-function over all policies:

$$Q^*(s, a) \triangleq \max_{\pi \in \Pi} Q^{\pi}(s, a)$$

Bellman Equation

Bellman expectation equation for V^π

- The value function V^π is the unique solution to the following Bellman equation:

$$\begin{aligned} V^\pi(\textcolor{blue}{s}) &= \mathbb{E}[r(s_t, a_t) + \gamma V^\pi(s_{t+1}) | s_t = \textcolor{blue}{s}] \\ &= \sum_{\textcolor{red}{a} \in A} \pi(\textcolor{red}{a} | \textcolor{blue}{s}) \left(r(\textcolor{blue}{s}, \textcolor{red}{a}) + \gamma \sum_{s' \in S} p(s' | \textcolor{blue}{s}, \textcolor{red}{a}) V^\pi(s') \right) = \sum_{\textcolor{red}{a} \in A} \pi(\textcolor{red}{a} | \textcolor{blue}{s}) Q^\pi(\textcolor{blue}{s}, \textcolor{red}{a}) \end{aligned}$$

- In operator form

$$V^\pi = TV^\pi$$

Bellman expectation equation for Q^π

- The Q-function Q^π satisfies:

$$\begin{aligned} Q^\pi(\textcolor{blue}{s}, \textcolor{red}{a}) &= \mathbb{E}[r(s_t, a_t) + \gamma V^\pi(s_{t+1}) | s_t = \textcolor{blue}{s}, a_t = \textcolor{red}{a}] \\ &= r(\textcolor{blue}{s}, \textcolor{red}{a}) + \gamma \sum_{s' \in S} p(s' | \textcolor{blue}{s}, \textcolor{red}{a}) V^\pi(s') \\ &= r(\textcolor{blue}{s}, \textcolor{red}{a}) + \gamma \sum_{s' \in S} \left(p(s' | \textcolor{blue}{s}, \textcolor{red}{a}) \sum_{a' \in A} \pi(a' | s') Q^\pi(s', a') \right) \end{aligned}$$

Bellman Equation

Bellman optimality equation for V^*

- The optimal value function V^* must satisfies the self-consistency condition given by the Bellman equation for V^π :

$$\begin{aligned} V^*(\textcolor{blue}{s}) &= \max_{\textcolor{red}{a}_t \in A} \mathbb{E}[r(s_t, \textcolor{red}{a}_t) + \gamma V^*(s_{t+1}) | s_t = \textcolor{blue}{s}] \\ &= \max_{\textcolor{red}{a} \in A} \left(r(\textcolor{blue}{s}, \textcolor{red}{a}) + \gamma \sum_{s' \in S} p(s' | \textcolor{blue}{s}, \textcolor{red}{a}) V^*(s') \right) = \max_{\textcolor{red}{a} \in A} Q^*(\textcolor{blue}{s}, \textcolor{red}{a}) \end{aligned}$$

- In operator form

$$V^* = TV^*$$

Bellman optimality equation for Q^*

- The optimal Q-function Q^* satisfies:

$$\begin{aligned} Q^*(\textcolor{blue}{s}, \textcolor{red}{a}) &= \mathbb{E}[r(s_t, a_t) + \gamma V^*(s_{t+1}) | s_t = \textcolor{blue}{s}, a_t = \textcolor{red}{a}] \\ &= r(\textcolor{blue}{s}, \textcolor{red}{a}) + \gamma \sum_{s' \in S} p(s' | \textcolor{blue}{s}, \textcolor{red}{a}) V^*(s') \\ &= r(\textcolor{blue}{s}, \textcolor{red}{a}) + \gamma \sum_{s' \in S} p(s' | \textcolor{blue}{s}, \textcolor{red}{a}) \max_{a' \in A} Q^*(s', a') \end{aligned}$$

Bellman Equation

The MDP problem

- To find an optimal policy that maximizes the expected cumulative reward:

$$\max_{\pi \in \Pi} \mathbb{E}_{\pi} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \right]$$

Known model:

- Policy iteration
- Value iteration

Unknown model:

- Temporal-difference learning
- Q-learning

Maximum Entropy Reinforcement Learning

Common issue: exploration

- Is there a value of exploring unknown regions of the environment?
- Which action should we try to explore unknown regions of the environment?



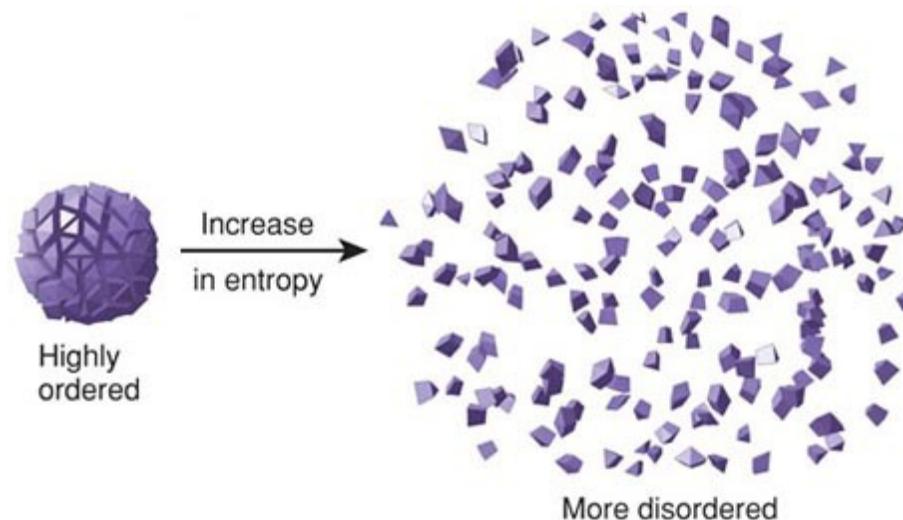
Maximum Entropy Reinforcement Learning

What is entropy?

- Average rate at which information is produced by a stochastic source of data
- The well-known standard Shannon-Gibbs entropy:

$$H(P) \triangleq \mathbb{E}_{X \sim P}[-\log(P(X))] = - \sum_{x \in X} P(x) \log P(x)$$

- More generally, entropy refers to **disorder**



Maximum Entropy Reinforcement Learning

What's the benefit of high entropy policy?

- Entropy of policy:

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

- Higher disorder in policy π
- Try new risky behaviors: Potentially explore unexplored regions

Soft MDPs

Standard MDP problem:

$$\max_{\pi} \mathbb{E}_{\pi} \left[\sum_t \gamma^t r(s_t, a_t) \right]$$

Maximum entropy MDP problem:

$$\max_{\pi} \mathbb{E}_{\pi} \left[\sum_t \gamma^t \left(r(s_t, a_t) + H(\pi_t(\cdot | s_t)) \right) \right]$$

Soft Bellman Equation

Theorem 1 (Soft value functions and Optimal policy)

Soft Q-function:

$$Q_{\text{soft}}^{\pi}(s_t, a_t) \triangleq r(s_t, a_t) + \mathbb{E}_{\pi} \left[\sum_{l=1}^{\infty} \gamma^l \left(r(s_{t+l}, a_{t+l}) + H(\pi(\cdot | s_{t+l})) \right) \right]$$

Soft value function:

$$V_{\text{soft}}^{\pi}(s_t) \triangleq \log \int \exp(Q_{\text{soft}}^{\pi}(s_t, a)) da$$

Optimal value functions:

$$Q_{\text{soft}}^{*}(s_t, a_t) \triangleq \max_{\pi} Q_{\text{soft}}^{\pi}(s_t, a_t)$$

$$V_{\text{soft}}^{*}(s_t) \triangleq \log \int \exp(Q_{\text{soft}}^{*}(s_t, a)) da$$

Soft Bellman Equation

Proof.

Maximum entropy MDP problem:

$$\max_{\pi} \mathbb{E}_{\pi} \left[\sum_t \gamma^t \left(r(s_t, a_t) + H(\pi_t(\cdot | s_t)) \right) \right]$$

Soft value function:

$$\begin{aligned} V_{\text{soft}}^{\pi}(s) &= \mathbb{E}_{\pi} \left[\sum_{\tau=t} \gamma^{\tau-t} (r(s_{\tau}, a_{\tau}) - \log \pi(a_{\tau} | s_{\tau})) \mid s_t = s \right] \\ &= \mathbb{E}_{\pi} [r(s_t, a_t) - \log \pi(a_t | s_t) + \gamma V_{\text{soft}}^{\pi}(s_{t+1}) \mid s_t = s] \end{aligned}$$

Soft Q-function:

$$\begin{aligned} Q_{\text{soft}}^{\pi}(s, a) &= \mathbb{E}_{\pi} \left[\sum_{\tau=t} \gamma^{\tau-t} (r(s_{\tau}, a_{\tau}) - \log \pi(a_{\tau} | s_{\tau})) \mid s_t = s, a_t = a \right] \\ &= \mathbb{E}_{\pi} [r(s_t, a_t) + \gamma V_{\text{soft}}^{\pi}(s_{t+1}) \mid s_t = s, a_t = a] \end{aligned}$$

note that $Q_{\text{soft}}^{\pi}(s, a)$ is the same with the original form

Soft Bellman Equation

Proof.

Soft value function:

$$\begin{aligned} V_{\text{soft}}^{\pi}(s) &= \mathbb{E}_{\pi} \left[\sum_{\tau=t} \gamma^{\tau-t} (r(s_{\tau}, a_{\tau}) - \log \pi(a_{\tau}|s_{\tau})) \mid s_t = s \right] \\ &= \mathbb{E}_{\pi} [r(s_t, a_t) - \log \pi(a_t|s_t) + \gamma V_{\text{soft}}^{\pi}(s_{t+1}) \mid s_t = s] \end{aligned}$$

Soft Q-function:

$$\begin{aligned} Q_{\text{soft}}^{\pi}(s, a) &= \mathbb{E}_{\pi} \left[\sum_{\tau=t} \gamma^{\tau-t} (r(s_{\tau}, a_{\tau}) - \log \pi(a_{\tau}|s_{\tau})) \mid s_t = s, a_t = a \right] \\ &= \mathbb{E}_{\pi} [r(s_t, a_t) + \gamma V_{\text{soft}}^{\pi}(s_{t+1}) \mid s_t = s, a_t = a] \end{aligned}$$

Soft value function (rewrite):

$$\begin{aligned} V_{\text{soft}}^{\pi}(s) &= \mathbb{E}_{\pi} [r(s_t, a_t) - \log \pi(a_t|s_t) + \gamma V_{\text{soft}}^{\pi}(s_{t+1}) \mid s_t = s] \\ &= \mathbb{E}_{\pi} [Q_{\text{soft}}^{\pi}(s_t, a_t) - \log \pi(a_t|s_t) \mid s_t = s] \end{aligned}$$

Soft Bellman Equation

Proof.

Given a policy π_{old} , define a new policy π_{new} as:

$$\begin{aligned}\pi_{\text{new}}(a|s) &\propto \exp(Q_{\text{soft}}^{\pi_{\text{old}}}(s, a)) \\ \pi_{\text{new}}(a|s) &= \frac{\exp(Q_{\text{soft}}^{\pi_{\text{old}}}(s, a))}{\int \exp(Q_{\text{soft}}^{\pi_{\text{old}}}(s, a')) da'} = \text{soft max}_{a \in A} Q_{\text{soft}}^{\pi_{\text{old}}}(s, a) \\ &= \exp(Q_{\text{soft}}^{\pi_{\text{old}}}(s, a) - V_{\text{soft}}^{\pi_{\text{old}}}(s)) = \exp(A_{\text{soft}}^{\pi_{\text{old}}}(s, a))\end{aligned}$$

Substitute $\pi_{\text{new}}(a|s)$ into $V_{\text{soft}}^{\pi}(s)$:

$$\begin{aligned}V_{\text{soft}}^{\pi}(s) &= \mathbb{E}_{a \sim \pi} \left[Q_{\text{soft}}^{\pi}(s, a) - \log \left(\frac{\exp(Q_{\text{soft}}^{\pi}(s, a))}{\int \exp(Q_{\text{soft}}^{\pi}(s, a')) da'} \right) \right] \\ &= \mathbb{E}_{a \sim \pi} \left[Q_{\text{soft}}^{\pi}(s, a) - \log \exp(Q_{\text{soft}}^{\pi}(s, a)) + \log \int \exp(Q_{\text{soft}}^{\pi}(s, a')) da' \right]\end{aligned}$$

Soft Bellman Equation

Proof.

Substitute $\pi_{\text{new}}(a|s)$ into $V_{\text{soft}}^{\pi}(s)$:

$$\begin{aligned} V_{\text{soft}}^{\pi}(s) &= \mathbb{E}_{a \sim \pi} \left[Q_{\text{soft}}^{\pi}(s, a) - \log \exp(Q_{\text{soft}}^{\pi}(s, a)) + \log \int \exp(Q_{\text{soft}}^{\pi}(s, a')) da' \right] \\ &= \mathbb{E}_{a \sim \pi} \left[\underbrace{\log \int \exp(Q_{\text{soft}}^{\pi}(s, a')) da'}_{\text{independent from } a} \right] \\ &= \log \int \exp(Q_{\text{soft}}^{\pi}(s, a')) da' \\ \therefore V_{\text{soft}}^{\pi}(s) &= \log \int \exp(Q_{\text{soft}}^{\pi}(s, a)) da \end{aligned}$$

Soft Bellman Equation

Theorem 1 (Soft value functions and Optimal policy)

Soft Q-function:

$$Q_{\text{soft}}^{\pi}(s_t, a_t) \triangleq r(s_t, a_t) + \mathbb{E}_{\pi} \left[\sum_{l=1}^{\infty} \gamma^l \left(r(s_{t+l}, a_{t+l}) + H(\pi(\cdot | s_{t+l})) \right) \right]$$

Soft value function:

$$V_{\text{soft}}^{\pi}(s_t) \triangleq \log \int \exp(Q_{\text{soft}}^{\pi}(s_t, a)) da$$

Optimal value functions:

$$Q_{\text{soft}}^{*}(s_t, a_t) \triangleq \max_{\pi} Q_{\text{soft}}^{\pi}(s_t, a_t)$$

$$V_{\text{soft}}^{*}(s_t) \triangleq \log \int \exp(Q_{\text{soft}}^{*}(s_t, a)) da$$

Soft Bellman Equation

Theorem 2 (Soft policy improvement theorem)

Given a policy π_{old} , define a new policy π_{new} as:

$$\pi_{\text{new}}(a|s) \propto \exp(Q_{\text{soft}}^{\pi_{\text{old}}}(s, a))$$

Assume that throughout our computation, Q is bounded and $\int \exp(Q(s, a)) da$ is bounded for any s (for both π_{old} and π_{new}).

Then,

$$Q_{\text{soft}}^{\pi_{\text{new}}}(s, a) \geq Q_{\text{soft}}^{\pi_{\text{old}}}(s, a) \quad \forall s, a$$

→ Monotonic policy improvement!

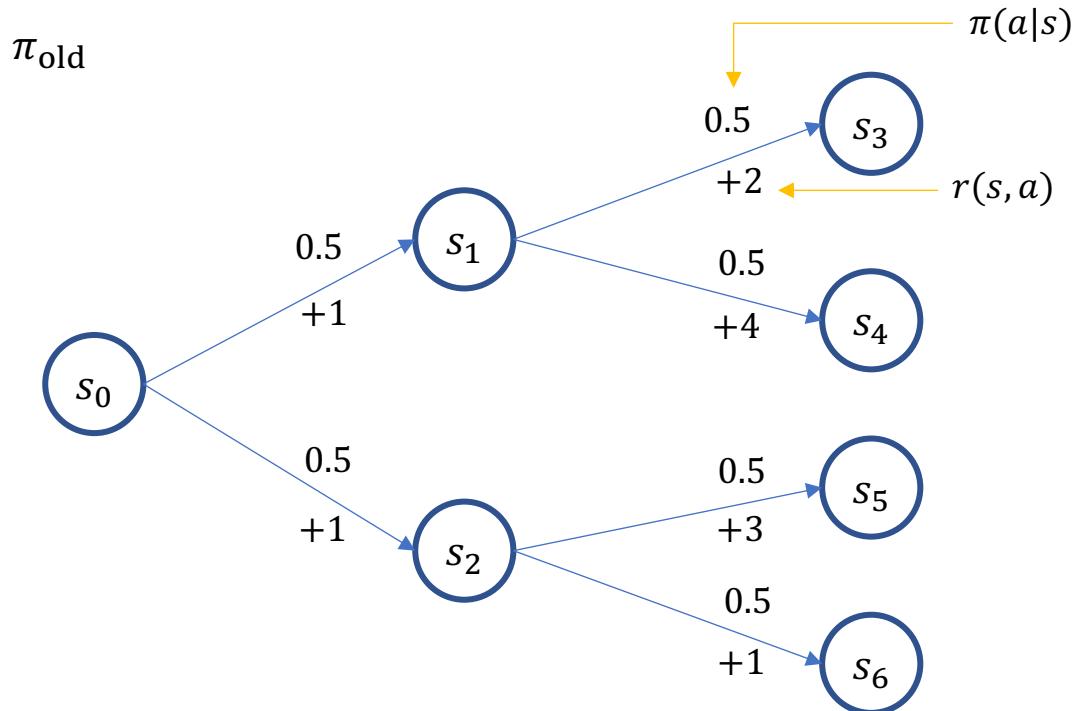
Soft Bellman Equation

Proof.

If one greedily maximize the sum of entropy and value with one-step look-ahead, then one obtains π_{new} from π_{old} :

$$H(\pi_{\text{old}}(\cdot | s)) + \mathbb{E}_{\pi_{\text{old}}}[Q_{\text{soft}}^{\pi_{\text{old}}}(s, a)] \leq H(\pi_{\text{new}}(\cdot | s)) + \mathbb{E}_{\pi_{\text{new}}}[Q_{\text{soft}}^{\pi_{\text{old}}}(s, a)]$$

Example



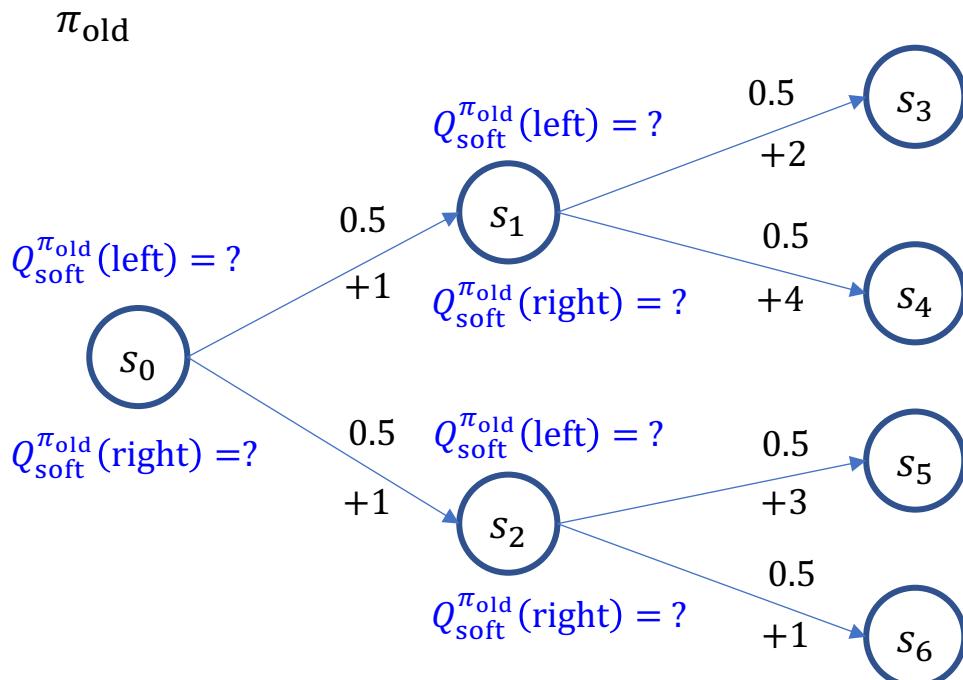
Soft Bellman Equation

Proof.

If one greedily maximize the sum of entropy and value with one-step look-ahead, then one obtains π_{new} from π_{old} :

$$H(\pi_{\text{old}}(\cdot|s)) + \mathbb{E}_{\pi_{\text{old}}}[Q_{\text{soft}}^{\pi_{\text{old}}}(s, a)] \leq H(\pi_{\text{new}}(\cdot|s)) + \mathbb{E}_{\pi_{\text{new}}}[Q_{\text{soft}}^{\pi_{\text{old}}}(s, a)]$$

Example



Soft Q-function:

$$\begin{aligned} Q_{\text{soft}}^\pi(s, a) &\triangleq r(s_t, a_t) + \mathbb{E}_\pi \left[\sum_{l=1}^{\infty} \gamma^l (r(s_{t+l}, a_{t+l}) + H(\pi(\cdot|s_{t+l}))) \right] \\ &= r(s, a) + \gamma p(s'|s, a) V_{\text{soft}}^\pi(s') \end{aligned}$$

Soft value function:

$$V_{\text{soft}}^\pi(s) = \sum_a \pi(a|s) (Q_{\text{soft}}^\pi(s, a) - \log \pi(a|s))$$

Entropy of policy:

$$H(\pi(\cdot|s)) \triangleq - \sum_a \pi(a|s) \log \pi(a|s)$$

Soft Bellman Equation

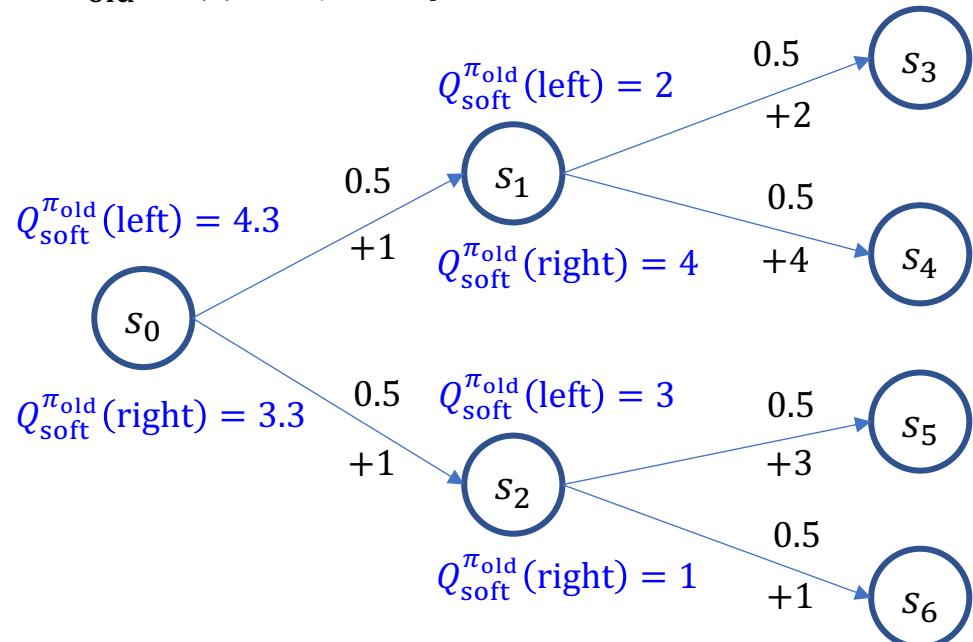
Proof.

If one greedily maximize the sum of entropy and value with one-step look-ahead, then one obtains π_{new} from π_{old} :

$$H(\pi_{\text{old}}(\cdot | s)) + \mathbb{E}_{\pi_{\text{old}}}[Q_{\text{soft}}^{\pi_{\text{old}}}(s, a)] \leq H(\pi_{\text{new}}(\cdot | s)) + \mathbb{E}_{\pi_{\text{new}}}[Q_{\text{soft}}^{\pi_{\text{old}}}(s, a)]$$

Example

π_{old} (suppose $\gamma = 1$, $p(s'|s, a) = 1$)



Soft Q-function:

$$\begin{aligned} Q_{\text{soft}}^\pi(s, a) &\triangleq r(s_t, a_t) + \mathbb{E}_\pi \left[\sum_{l=1}^{\infty} \gamma^l (r(s_{t+l}, a_{t+l}) + H(\pi(\cdot | s_{t+l}))) \right] \\ &= r(s, a) + \gamma p(s'|s, a) V_{\text{soft}}^\pi(s') \end{aligned}$$

Soft value function:

$$V_{\text{soft}}^\pi(s) = \sum_a \pi(a|s) (Q_{\text{soft}}^\pi(s, a) - \log \pi(a|s))$$

Entropy of policy:

$$H(\pi(\cdot | s)) \triangleq - \sum_a \pi(a|s) \log \pi(a|s)$$

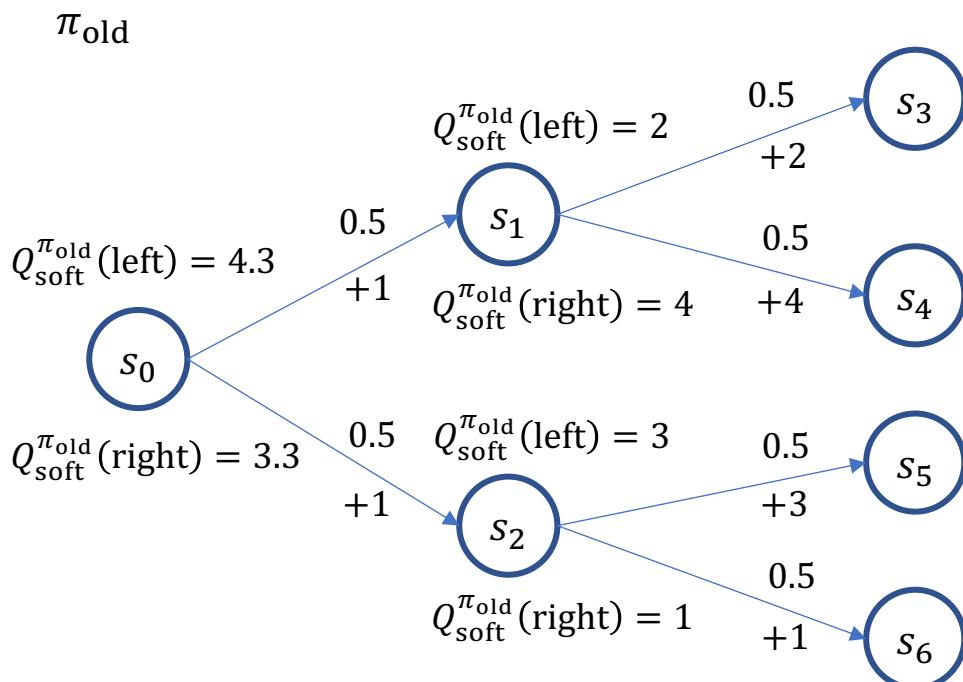
Soft Bellman Equation

Proof.

If one greedily maximize the sum of entropy and value with one-step look-ahead, then one obtains π_{new} from π_{old} :

$$H(\pi_{\text{old}}(\cdot|s)) + \mathbb{E}_{\pi_{\text{old}}}[Q_{\text{soft}}^{\pi_{\text{old}}}(s, a)] \leq H(\pi_{\text{new}}(\cdot|s)) + \mathbb{E}_{\pi_{\text{new}}}[Q_{\text{soft}}^{\pi_{\text{old}}}(s, a)]$$

Example



A new policy π_{new} :

$$\pi_{\text{new}}(a|s) \propto \exp(Q_{\text{soft}}^{\pi_{\text{old}}}(s, a))$$

$$\begin{aligned}\pi_{\text{new}}(a|s) &= \frac{\exp(Q_{\text{soft}}^{\pi_{\text{old}}}(s, a))}{\sum_{a'} \exp(Q_{\text{soft}}^{\pi_{\text{old}}}(s, a'))} = \underset{a \in A}{\text{softmax}} Q_{\text{soft}}^{\pi_{\text{old}}}(s, a) \\ &= \exp(Q_{\text{soft}}^{\pi_{\text{old}}}(s, a) - V_{\text{soft}}^{\pi_{\text{old}}}(s))\end{aligned}$$

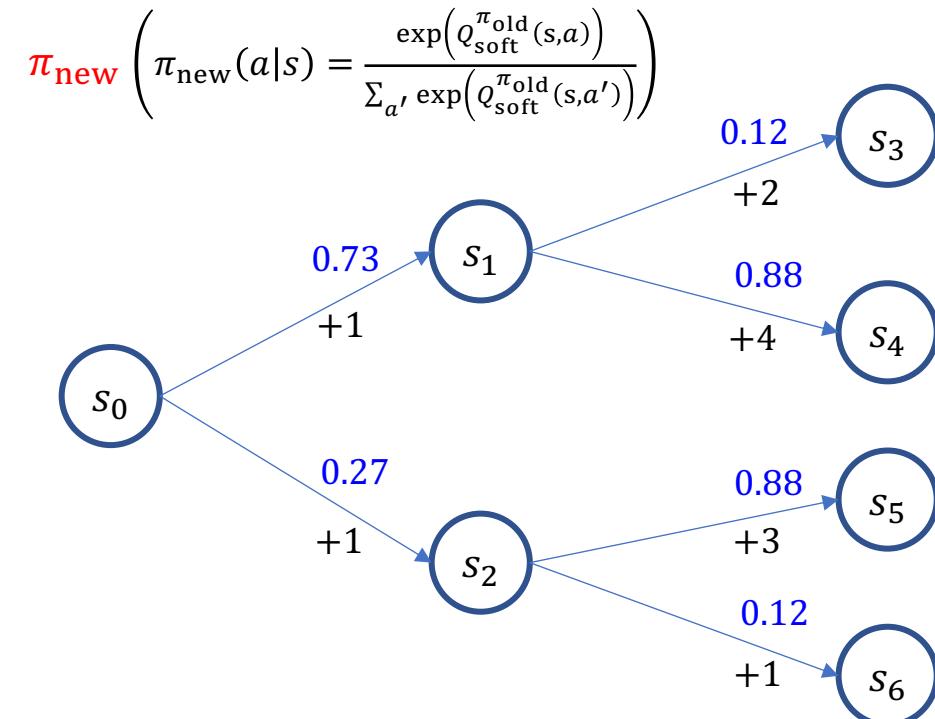
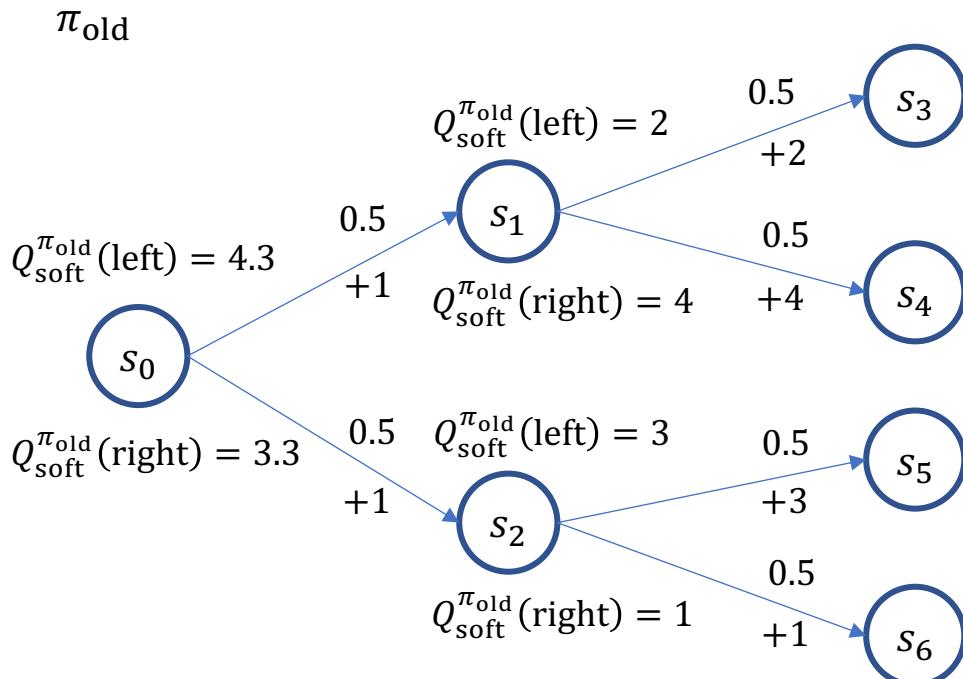
Soft Bellman Equation

Proof.

If one greedily maximize the sum of entropy and value with one-step look-ahead, then one obtains π_{new} from π_{old} :

$$H(\pi_{\text{old}}(\cdot | s)) + \mathbb{E}_{\pi_{\text{old}}}[Q_{\text{soft}}^{\pi_{\text{old}}}(s, a)] \leq H(\pi_{\text{new}}(\cdot | s)) + \mathbb{E}_{\pi_{\text{new}}}[Q_{\text{soft}}^{\pi_{\text{old}}}(s, a)]$$

Example



Soft Bellman Equation

Proof.

If one greedily maximize the sum of entropy and value with one-step look-ahead, then one obtains π_{new} from π_{old} :

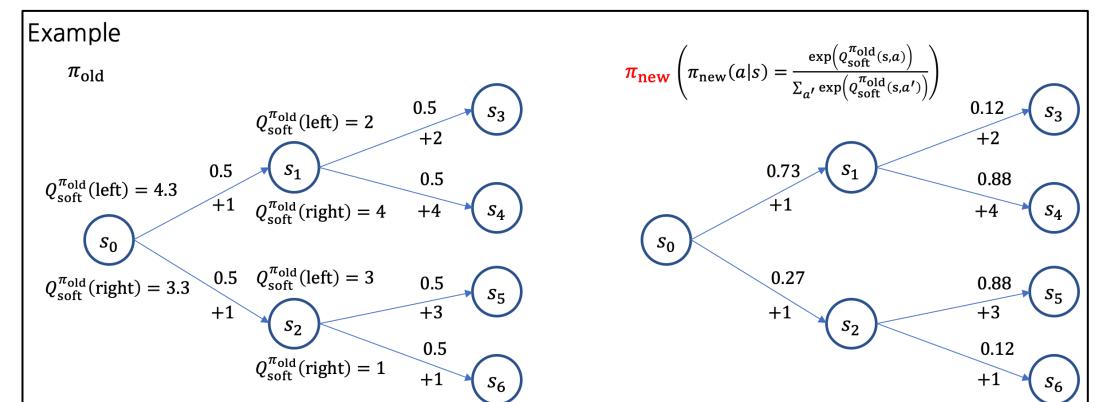
$$H(\pi_{\text{old}}(\cdot | s)) + \mathbb{E}_{\pi_{\text{old}}}[Q_{\text{soft}}^{\pi_{\text{old}}}(s, a)] \leq H(\pi_{\text{new}}(\cdot | s)) + \mathbb{E}_{\pi_{\text{new}}}[Q_{\text{soft}}^{\pi_{\text{old}}}(s, a)]$$

So, Substitute s_0, a_0 for s, a :

$$H(\pi_{\text{old}}(\cdot | s_0)) + \mathbb{E}_{\pi_{\text{old}}}[Q_{\text{soft}}^{\pi_{\text{old}}}(s_0, a_0)] \leq H(\pi_{\text{new}}(\cdot | s_0)) + \mathbb{E}_{\pi_{\text{new}}}[Q_{\text{soft}}^{\pi_{\text{old}}}(s_0, a_0)]$$

$$0.3 + 3.8 \leq 0.25 + 4.03$$

$$4.1 \leq 4.28$$



Soft Bellman Equation

Proof.

If one greedily maximize the sum of entropy and value with one-step look-ahead, then one obtains π_{new} from π_{old} :

$$H(\pi_{\text{old}}(\cdot | s)) + \mathbb{E}_{\pi_{\text{old}}} [Q_{\text{soft}}^{\pi_{\text{old}}}(s, a)] \leq H(\pi_{\text{new}}(\cdot | s)) + \mathbb{E}_{\pi_{\text{new}}} [Q_{\text{soft}}^{\pi_{\text{old}}}(s, a)]$$

Then, we can show that:

$$\begin{aligned} Q_{\text{soft}}^{\pi_{\text{old}}}(s, a) &= \mathbb{E}_{s_1} [r_0 + \gamma (H(\pi_{\text{old}}(\cdot | s_1)) + \mathbb{E}_{a_1 \sim \pi_{\text{old}}} [Q_{\text{soft}}^{\pi}(s_1, a_1)])] \\ &\leq \mathbb{E}_{s_1} [r_0 + \gamma (H(\pi_{\text{new}}(\cdot | s_1)) + \mathbb{E}_{a_1 \sim \pi_{\text{new}}} [Q_{\text{soft}}^{\pi}(s_1, a_1)])] \\ &= \mathbb{E}_{s_1} [r_0 + \gamma (H(\pi_{\text{new}}(\cdot | s_1)) + r_1)] + \gamma^2 \mathbb{E}_{s_2} [H(\pi_{\text{old}}(\cdot | s_2)) + \mathbb{E}_{a_2 \sim \pi_{\text{old}}} [Q_{\text{soft}}^{\pi}(s_2, a_2)]] \\ &\leq \mathbb{E}_{s_1} [r_0 + \gamma (H(\pi_{\text{new}}(\cdot | s_1)) + r_1)] + \gamma^2 \mathbb{E}_{s_2} [H(\pi_{\text{new}}(\cdot | s_2)) + \mathbb{E}_{a_2 \sim \pi_{\text{new}}} [Q_{\text{soft}}^{\pi}(s_2, a_2)]] \\ &= \mathbb{E}_{s_1, a_2 \sim \pi_{\text{new}}, s_2} [r_0 + \gamma (H(\pi_{\text{new}}(\cdot | s_1)) + r_1) + \gamma^2 H(\pi_{\text{new}}(\cdot | s_2)) + r_2] + \gamma^3 \mathbb{E}_{s_3} [H(\pi_{\text{new}}(\cdot | s_3)) + \mathbb{E}_{a_3 \sim \pi_{\text{new}}} [Q_{\text{soft}}^{\pi}(s_3, a_3)]] \\ &\vdots \\ &\leq \mathbb{E}_{\tau \sim \pi_{\text{new}}} [r_0 + \sum_{t=1}^{\infty} \gamma^t (H(\pi_{\text{new}}(\cdot | s_t)) + r_t)] \\ &= Q_{\text{soft}}^{\pi_{\text{new}}}(s, a) \end{aligned}$$

Soft Bellman Equation

Theorem 2 (Soft policy improvement theorem)

Given a policy π_{old} , define a new policy π_{new} as:

$$\pi_{\text{new}}(a|s) \propto \exp(Q_{\text{soft}}^{\pi_{\text{old}}}(s, a))$$

Assume that throughout our computation, Q is bounded and $\int \exp(Q(s, a)) da$ is bounded for any s (for both π_{old} and π_{new}).

Then,

$$Q_{\text{soft}}^{\pi_{\text{new}}}(s, a) \geq Q_{\text{soft}}^{\pi_{\text{old}}}(s, a) \quad \forall s, a$$

→ Monotonic policy improvement!

Soft Bellman Equation

Comparison to standard MDP: Bellman expectation equation

Standard:

$$\begin{aligned} Q^\pi(s, a) &= \mathbb{E}[r(s_t, a_t) + \gamma V^\pi(s_{t+1}) | s_t = s, a_t = a] \\ &= r(s, a) + \gamma \sum_{s' \in S} p(s' | s, a) V^\pi(s') \\ V^\pi(s) &= \sum_{a \in A} \pi(a | s) Q^\pi(s, a) \end{aligned}$$

Maximum entropy:

$$\begin{aligned} Q_{\text{soft}}^\pi(s, a) &= \mathbb{E}_\pi[r(s_t, a_t) + \gamma V_{\text{soft}}^\pi(s_{t+1}) | s_t = s, a_t = a] \\ &= r(s, a) + \gamma \sum_{s' \in S} p(s' | s, a) V_{\text{soft}}^\pi(s') \\ V_{\text{soft}}^\pi(s) &= \mathbb{E}_\pi[Q_{\text{soft}}^\pi(s_t, a_t) - \log \pi(a_t | s_t) | s_t = s] \\ &= \log \int \exp(Q_{\text{soft}}^\pi(s, a)) da \end{aligned}$$

Maximum entropy variant:

- Add explicit temperature parameter α

$$\begin{aligned} V_{\text{soft}}^\pi(s) &= \mathbb{E}_\pi[Q_{\text{soft}}^\pi(s_t, a_t) - \alpha \log \pi(a_t | s_t) | s_t = s] \\ &= \alpha \log \int \exp\left(\frac{1}{\alpha} Q_{\text{soft}}^\pi(s, a)\right) da \end{aligned}$$

Soft Bellman Equation

Comparison to standard MDP: Bellman optimality equation

Standard:

$$\begin{aligned} Q^*(s, a) &= \mathbb{E}[r(s_t, a_t) + \gamma V^*(s_{t+1}) | s_t = s, a_t = a] \\ &= r(s, a) + \gamma \sum_{s' \in S} p(s' | s, a) V^*(s') \end{aligned}$$

$$V^*(s) = \max_{a \in A} Q^*(s, a)$$

Maximum entropy variant:

$$\begin{aligned} Q_{\text{soft}}^*(s, a) &= \mathbb{E}_\pi[r(s_t, a_t) + \gamma V_{\text{soft}}^*(s_{t+1}) | s_t = s, a_t = a] \\ &= r(s, a) + \gamma \sum_{s' \in S} p(s' | s, a) V_{\text{soft}}^*(s') \end{aligned}$$

$$\begin{aligned} V_{\text{soft}}^*(s) &= \max_{a \in A} (Q_{\text{soft}}^*(s, a) - \alpha \log \pi(a | s)) \\ &= \alpha \log \int \exp\left(\frac{1}{\alpha} Q_{\text{soft}}^*(s, a)\right) da \end{aligned}$$

Connection:

- Note that as $\alpha \rightarrow 0$, $\exp\left(\frac{1}{\alpha} Q_{\text{soft}}^*(s, a)\right)$ emphasizes $\max_{a \in A} Q^*(s, a)$:

$$\alpha \log \int \exp\left(\frac{1}{\alpha} Q_{\text{soft}}^*(s, a)\right) da \rightarrow \max_{a \in A} Q^*(s, a) \quad \text{as } \alpha \rightarrow 0$$

Soft Bellman Equation

Comparison to standard MDP: Policy

Standard:

$$\pi_{\text{new}}(a|s) \in \arg \max_a Q^{\pi_{\text{old}}}(s, a)$$

→ Deterministic policy

Maximum Entropy:

$$\pi_{\text{new}}(a|s) = \frac{\exp\left(\frac{1}{\alpha} Q_{\text{soft}}^{\pi_{\text{old}}}(s, a)\right)}{\int \exp\left(\frac{1}{\alpha} Q_{\text{soft}}^{\pi_{\text{old}}}(s, a')\right) da'} = \exp\left(\frac{1}{\alpha} [Q_{\text{soft}}^{\pi_{\text{old}}}(s, a) - V_{\text{soft}}^{\pi_{\text{old}}}(s)]\right)$$

→ Stochastic policy

Soft Bellman Equation

So, what's the advantage of maximum entropy?

- Computation:
No maximization involved
- Exploration:
High entropy policy
- Structural similarity:
Can combine it with many RL methods for standard MDP

From Soft Policy Iteration to Soft Actor-Critic

Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor

Tuomas Haarnoja¹ Aurick Zhou¹ Pieter Abbeel¹ Sergey Levine¹

Version 1 (Jan 4 2018)

Soft Actor-Critic Algorithms and Applications

Tuomas Haarnoja^{*†‡} Aurick Zhou^{*†} Kristian Hartikainen^{*†} George Tucker[‡]
Sehoon Ha[‡] Jie Tan[‡] Vikash Kumar[‡] Henry Zhu[†] Abhishek Gupta[†]
Pieter Abbeel[†] Sergey Levine^{†‡}

Version 2 (Dec 13 2018)

Idea:

- Maximum entropy + off-policy actor-critic

Advantages:

- Exploration
- Sample efficiency
- Stable convergence
- Little hyperparameter tuning

Soft Policy Iteration

Soft policy evaluation

Modified Bellman operator:

$$T^\pi Q^\pi(s, a) \triangleq r(s, a) + \gamma \sum_{s' \in S} p(s'|s, a) V_{\text{soft}}^\pi(s')$$

where $V^\pi(s) = \mathbb{E}_\pi[Q_{\text{soft}}^\pi(s, a) - \alpha \log \pi(a|s)]$

Repeatedly apply T^π to Q_k :

$$Q_{k+1} \leftarrow T^\pi Q_k$$

- **Result: Q_k converges to Q^π !**

Soft Policy Iteration

Soft policy improvement

If no constraint on policies,

$$\pi_{\text{new}}(a|s) = \exp\left(\frac{1}{\alpha}[Q_{\text{soft}}^{\pi_{\text{old}}}(s, a) - V_{\text{soft}}^{\pi_{\text{old}}}(s)]\right)$$

When constraint need to be satisfied (i.e., $\pi \in \Pi$),

Information projection:

$$\pi_{\text{new}}(\cdot|s) \in \arg \min_{\pi'(\cdot|s) \in \Pi} D_{KL}\left(\pi'(\cdot|s) \parallel \exp\left(\frac{1}{\alpha}[Q_{\text{soft}}^{\pi_{\text{old}}}(s, \cdot) - V_{\text{soft}}^{\pi_{\text{old}}}(s)]\right)\right)$$

target policy

- Result: Monotonic improvement on policy! (π_{new} better than π_{old})

Soft Actor-Critic

Model-Free version of soft policy iteration: Soft actor-critic

Soft policy iteration: maximum entropy variant of policy iteration

Soft actor-critic (SAC): maximum entropy variant of actor-critic

- Soft critic:
evaluates soft Q-function Q_θ of policy π_ϕ
- Soft actor:
improves maximum entropy policy using critic's evaluation
- Temperature parameter alpha:
automatically adjusts entropy for maximum entropy policy using alpha α

Soft Actor-Critic

Soft critic: evaluates soft Q-function Q_θ of policy π_ϕ

Idea: DQN + Maximum entropy

- Training soft Q-function:

$$\min_{\theta} J_Q(\theta) \triangleq \mathbb{E}_{(s_t, a_t)} \left[\frac{1}{2} \left(Q_\theta(s_t, a_t) - \underbrace{\left[r(s_t, a_t) + \gamma \mathbb{E}_{s_{t+1}} [V_{\theta^-}(s_{t+1})] \right]}_{\text{target}} \right)^2 \right]$$

- Stochastic gradient:

$$\hat{\nabla}_\theta J_Q(\theta) = \nabla_\theta Q_\theta(s_t, a_t) \times \left[Q_\theta(s_t, a_t) - \underbrace{\left[r(s_t, a_t) + \gamma \left(Q_{\theta^-}(s_{t+1}, a_{t+1}) - \alpha \log \pi_\phi(a_{t+1} | s_{t+1}) \right) \right]}_{\text{sample estimate of } \mathbb{E}_{s_{t+1}} [V_{\theta^-}(s_{t+1})]} \right]$$

Soft Actor-Critic

Soft actor: updates policy using critic's evaluation

Idea: Policy gradient + Maximum entropy

- Minimizing the expected KL-divergence:

$$\min_{\phi} D_{KL} \left(\pi_{\phi}(\cdot | s_t) \parallel \exp \left(\frac{1}{\alpha} [Q_{\theta}(s_t, \cdot) - V_{\theta}(s_t)] \right) \right)$$


target policy

which is equivalent to

$$\min_{\phi} J_{\pi}(\phi) \triangleq \mathbb{E}_{s_t} \left[\mathbb{E}_{a_t \sim \pi_{\phi}(\cdot | s_t)} [\alpha \log \pi_{\phi}(a_t | s_t) - Q_{\theta}(s_t, a_t)] \right]$$

Soft Actor-Critic

Reparameterization trick

Reparameterize the policy as

$$a_t \triangleq f_\phi(\epsilon_t; s_t)$$

which ϵ_t is an input noise with some fixed distribution (e.g., Gaussian)

Benefit: lower variance

Rewrite the policy optimization problem as

$$\min_{\phi} J_\pi(\phi) \triangleq \mathbb{E}_{s_t, \epsilon_t} \left[\alpha \log \pi_\phi(f_\phi(\epsilon_t; s_t) | s_t) - Q_\theta(s_t, f_\phi(\epsilon_t; s_t)) \right]$$

Stochastic gradient:

$$\widehat{\nabla}_\phi J_\pi(\phi) = \nabla_\phi \alpha \log \pi_\phi(a_t | s_t) + [\nabla_{a_t} \alpha \log \pi_\phi(a_t | s_t) - \nabla_{a_t} Q_\theta(s_t, a_t)] \times \nabla_\phi f_\phi(\epsilon_t; s_t)$$

which a_t is evaluated at $f_\phi(\epsilon_t; s_t)$

Soft Actor-Critic

Automating entropy adjustment

Why do we have to do automatic temperature parameter α adjustment?

- Choosing the optimal temperature α_t^* is non-trivial, and the temperature needs to be tuned for each task.
- Since the entropy $H(\pi_t)$ can vary unpredictably both across tasks and during training as the policy becomes better, this makes the temperature adjustment particularly difficult.
- Thus, simply forcing the entropy to a fixed value is a poor solution, since the policy should be free to explore more in regions.

Soft Actor-Critic

Automating entropy adjustment

Constrained optimization problem (suppose $\gamma = 1$):

$$\max_{\pi_0, \dots, \pi_T} \mathbb{E} \left[\sum_{t=0}^T r(s_t, a_t) \right] \quad \text{s. t. } H(\pi_t) \geq H_0 \quad \forall t$$

where H_0 is a desired minimum expected entropy threshold

Rewrite the objective as an iterated maximization employing an (approximate) dynamic programming approach:

$$\max_{\pi_0} \left(\mathbb{E}[r(s_0, a_0)] + \max_{\pi_1} \left(\mathbb{E}[\dots] + \max_{\pi_T} \mathbb{E}[r(s_T, a_T)] \right) \right)$$

Subject to the constraint on entropy

Start from the last time step T :

$$\max_{\pi_T} \mathbb{E}_{(s_T, a_T) \sim \rho_\pi} [r(s_T, a_T)]$$

Subject to $\mathbb{E}_{(s_T, a_T) \sim \rho_\pi} [-\log \pi_T(a_T | s_T)] - H_0 \geq 0$

Soft Actor-Critic

Automating entropy adjustment

Change the constrained maximization to the dual problem using Lagrangian:

1. Create the following function:

$$f(\pi_T) = \begin{cases} \mathbb{E}_{(s_T, a_T) \sim \rho_\pi}[r(s_T, a_T)], & \text{s. t. } h(\pi_T) \geq 0 \\ -\infty, & \text{otherwise} \end{cases}$$

$$h(\pi_T) = H(\pi_T) - H_0 = \mathbb{E}_{(s_T, a_T) \sim \rho_\pi}[-\log \pi_T(a_T | s_T)] - H_0$$

2. So, rewrite the objective as

$$\max_{\pi_T} f(\pi_T) \quad \text{s. t. } h(\pi_T) \geq 0$$

3. Use Lagrange multiplier:

$$\min_{\alpha_T \geq 0} L(\pi_T, \alpha_T) = f(\pi_T) + \alpha_T h(\pi_T)$$

where α_T is the dual variable

4. Rewrite the objective using Lagrangian as

$$\max_{\pi_T} f(\pi_T) = \min_{\alpha_T \geq 0} \max_{\pi_T} L(\pi_T, \alpha_T) = \min_{\alpha_T \geq 0} \max_{\pi_T} f(\pi_T) + \alpha_T h(\pi_T)$$

Soft Actor-Critic

Automating entropy adjustment

Change the constrained maximization to the dual problem using Lagrangian:

$$\begin{aligned}\max_{\pi_T} \mathbb{E}_{(s_T, a_T) \sim \rho_\pi} [r(s_T, a_T)] &= \max_{\pi_T} f(\pi_T) \\&= \min_{\alpha_T \geq 0} \max_{\pi_T} L(\pi_T, \alpha_T) = \min_{\alpha_T \geq 0} \max_{\pi_T} f(\pi_T) + \alpha_T h(\pi_T) \\&= \min_{\alpha_T \geq 0} \max_{\pi_T} \mathbb{E}_{(s_T, a_T) \sim \rho_\pi} [r(s_T, a_T)] + \alpha_T (\mathbb{E}_{(s_T, a_T) \sim \rho_\pi} [-\log \pi_T(a_T | s_T)] - H_0) \\&= \min_{\alpha_T \geq 0} \max_{\pi_T} \mathbb{E}_{(s_T, a_T) \sim \rho_\pi} [r(s_T, a_T) - \alpha_T \log \pi_T(a_T | s_T)] - \alpha_T H_0 \\&= \min_{\alpha_T \geq 0} \max_{\pi_T} \mathbb{E}_{(s_T, a_T) \sim \rho_\pi} [r(s_T, a_T) + \alpha_T H(\pi_T) - \alpha_T H_0]\end{aligned}$$

We can solve for

- Optimal policy π_T^* as

$$\pi_T^* = \arg \max_{\pi_T} \mathbb{E}_{(s_T, a_T) \sim \rho_\pi} [r(s_T, a_T) + \alpha_T H(\pi_T) - \alpha_T H_0]$$

- Optimal dual variable α_T^* as

$$\alpha_T^* = \arg \min_{\alpha_T \geq 0} \mathbb{E}_{(s_T, a_T) \sim \rho_{\pi^*}} [\alpha_T H(\pi_T^*) - \alpha_T H_0]$$

Thus,

$$\max_{\pi_T} \mathbb{E}[r(s_T, a_T)] = \mathbb{E}_{(s_T, a_T) \sim \rho_{\pi^*}} [r(s_T, a_T) + \alpha_T^* H(\pi_T^*) - \alpha_T^* H_0]$$

Soft Actor-Critic

Automating entropy adjustment

Then, the soft Q-function can be computed as

$$\begin{aligned} Q_{T-1}(s_{T-1}, a_{T-1}) &= r(s_{T-1}, a_{T-1}) + \mathbb{E}[Q_T(s_T, a_T) - \alpha_T \log \pi_T(a_T | s_T)] \\ &= r(s_{T-1}, a_{T-1}) + \mathbb{E}[r(s_T, a_T)] + \alpha_T H(\pi_T) \end{aligned}$$

$$Q_{T-1}^*(s_{T-1}, a_{T-1}) = r(s_{T-1}, a_{T-1}) + \max_{\pi_T} \mathbb{E}[r(s_T, a_T)] + \alpha_T^* H(\pi_T^*)$$

Now, given the time step $T - 1$, we have:

$$\begin{aligned} &\max_{\pi_{T-1}} \left(\mathbb{E}[r(s_{T-1}, a_{T-1})] + \max_{\pi_T} \mathbb{E}[r(s_T, a_T)] \right) \\ &= \max_{\pi_{T-1}} \left(Q_{T-1}^*(s_{T-1}, a_{T-1}) - \alpha_T^* H(\pi_T^*) \right) \\ &= \min_{\alpha_{T-1} \geq 0} \max_{\pi_{T-1}} \left(Q_{T-1}^*(s_{T-1}, a_{T-1}) - \alpha_T^* H(\pi_T^*) + \alpha_{T-1} (H(\pi_{T-1}) - H_0) \right) \\ &= \min_{\alpha_{T-1} \geq 0} \max_{\pi_{T-1}} \left(Q_{T-1}^*(s_{T-1}, a_{T-1}) - \alpha_{T-1} H(\pi_{T-1}) - \alpha_{T-1} H_0 - \alpha_T^* H(\pi_T^*) \right) \end{aligned}$$

Soft Actor-Critic

Automating entropy adjustment

We can solve for

- Optimal policy π_{T-1}^* as

$$\pi_{T-1}^* = \arg \max_{\pi_{T-1}} \mathbb{E}_{(s_{T-1}, a_{T-1}) \sim \rho_\pi} [Q_{T-1}^*(s_{T-1}, a_{T-1}) + \alpha_{T-1} H(\pi_{T-1}) - \alpha_{T-1} H_0]$$

- Optimal dual variable α_{T-1}^* as

$$\alpha_{T-1}^* = \arg \min_{\alpha_{T-1} \geq 0} \mathbb{E}_{(s_{T-1}, a_{T-1}) \sim \rho_{\pi^*}} [\alpha_{T-1} H(\pi_{T-1}^*) - \alpha_{T-1} H_0]$$

In this way, we can proceed backwards in time and recursively optimize constrained optimization problem. Then, we can solve the optimal dual variable α_t^* after solving for Q_t^* and π_t^* :

$$\alpha_t^* = \arg \min_{\alpha_t} \mathbb{E}_{a_t \sim \pi_t^*} [-\alpha_t \log \pi_t^*(a_t | s_t) - \alpha_t H_0]$$

Soft Actor-Critic

Temperature parameter alpha: automatically adjusts entropy for maximum entropy policy using alpha α

Idea: Constrained optimization problem + Dual problem

- Minimizing the dual objective by approximating dual gradient descent:

$$\min_{\alpha} \mathbb{E}_{a_t \sim \pi_t} [-\alpha \log \pi_t(a_t | s_t) - \alpha H_0]$$

Soft Actor-Critic

Putting everything together: Soft Actor-Critic (SAC)

Algorithm 1 Soft Actor-Critic

Input: θ_1, θ_2, ϕ

$$\bar{\theta}_1 \leftarrow \theta_1, \bar{\theta}_2 \leftarrow \theta_2$$

$$\mathcal{D} \leftarrow \emptyset$$

for each iteration **do**

for each environment step **do**

$$\mathbf{a}_t \sim \pi_\phi(\mathbf{a}_t | \mathbf{s}_t)$$

$$\mathbf{s}_{t+1} \sim p(\mathbf{s}_{t+1} | \mathbf{s}_t, \mathbf{a}_t)$$

$$\mathcal{D} \leftarrow \mathcal{D} \cup \{(\mathbf{s}_t, \mathbf{a}_t, r(\mathbf{s}_t, \mathbf{a}_t), \mathbf{s}_{t+1})\}$$

 ▷ Initial parameters

 ▷ Initialize target network weights

 ▷ Initialize an empty replay pool

end for

for each gradient step **do**

$$\theta_i \leftarrow \theta_i - \lambda_Q \hat{\nabla}_{\theta_i} J_Q(\theta_i) \text{ for } i \in \{1, 2\}$$

 ▷ Sample action from the policy

 ▷ Sample transition from the environment

 ▷ Store the transition in the replay pool

$$\phi \leftarrow \phi - \lambda_\pi \hat{\nabla}_\phi J_\pi(\phi)$$

 ▷ Update the Q-function parameters

$$\alpha \leftarrow \alpha - \lambda \hat{\nabla}_\alpha J(\alpha)$$

 ▷ Update policy weights

$$\bar{\theta}_i \leftarrow \tau \theta_i + (1 - \tau) \bar{\theta}_i \text{ for } i \in \{1, 2\}$$

 ▷ Adjust temperature

end for

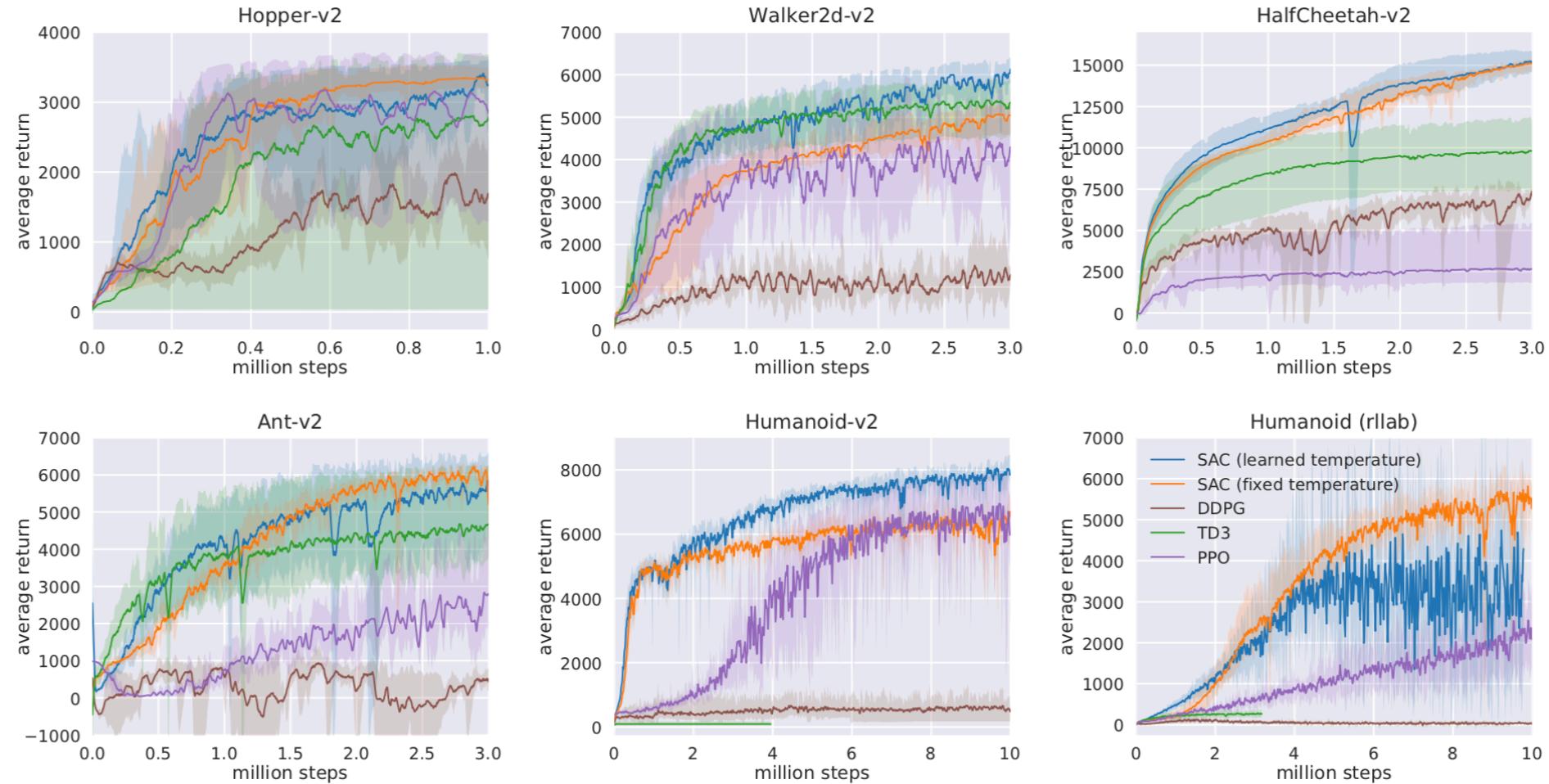
 ▷ Update target network weights

end for

Output: θ_1, θ_2, ϕ

 ▷ Optimized parameters

Results



- Soft actor-critic (blue and yellow) performs **consistently** across all tasks and **outperforming** both on-policy and off-policy methods in the most challenging tasks.

Results

Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor

Tuomas Haarnoja¹ Aurick Zhou¹ Pieter Abbeel¹ Sergey Levine¹

Version 1 (Jan 4 2018)

Idea:

- Maximum entropy + off-policy actor-critic

Advantages:

- Exploration
- Sample efficiency
- Stable convergence
- Little hyperparameter tuning

Soft Actor-Critic Algorithms and Applications

Tuomas Haarnoja^{*†‡} Aurick Zhou^{*†} Kristian Hartikainen^{*†} George Tucker[‡]
Sehoon Ha[‡] Jie Tan[‡] Vikash Kumar[‡] Henry Zhu[†] Abhishek Gupta[†]
Pieter Abbeel[†] Sergey Levine^{†‡}

Version 2 (Dec 13 2018)

Results

Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor

Tuomas Haarnoja¹ Aurick Zhou¹ Pieter Abbeel¹ Sergey Levine¹

Version 1 (Jan 4 2018)

Idea:

- Maximum entropy + off-policy actor-critic

Advantages:

- Exploration
- Sample efficiency
- Stable convergence
- Little hyperparameter tuning

Soft Actor-Critic Algorithms and Applications

Tuomas Haarnoja^{*†‡} Aurick Zhou^{*†} Kristian Hartikainen^{*†} George Tucker[‡]
Sehoon Ha[‡] Jie Tan[‡] Vikash Kumar[‡] Henry Zhu[†] Abhishek Gupta[†]
Pieter Abbeel[†] Sergey Levine^{†‡}

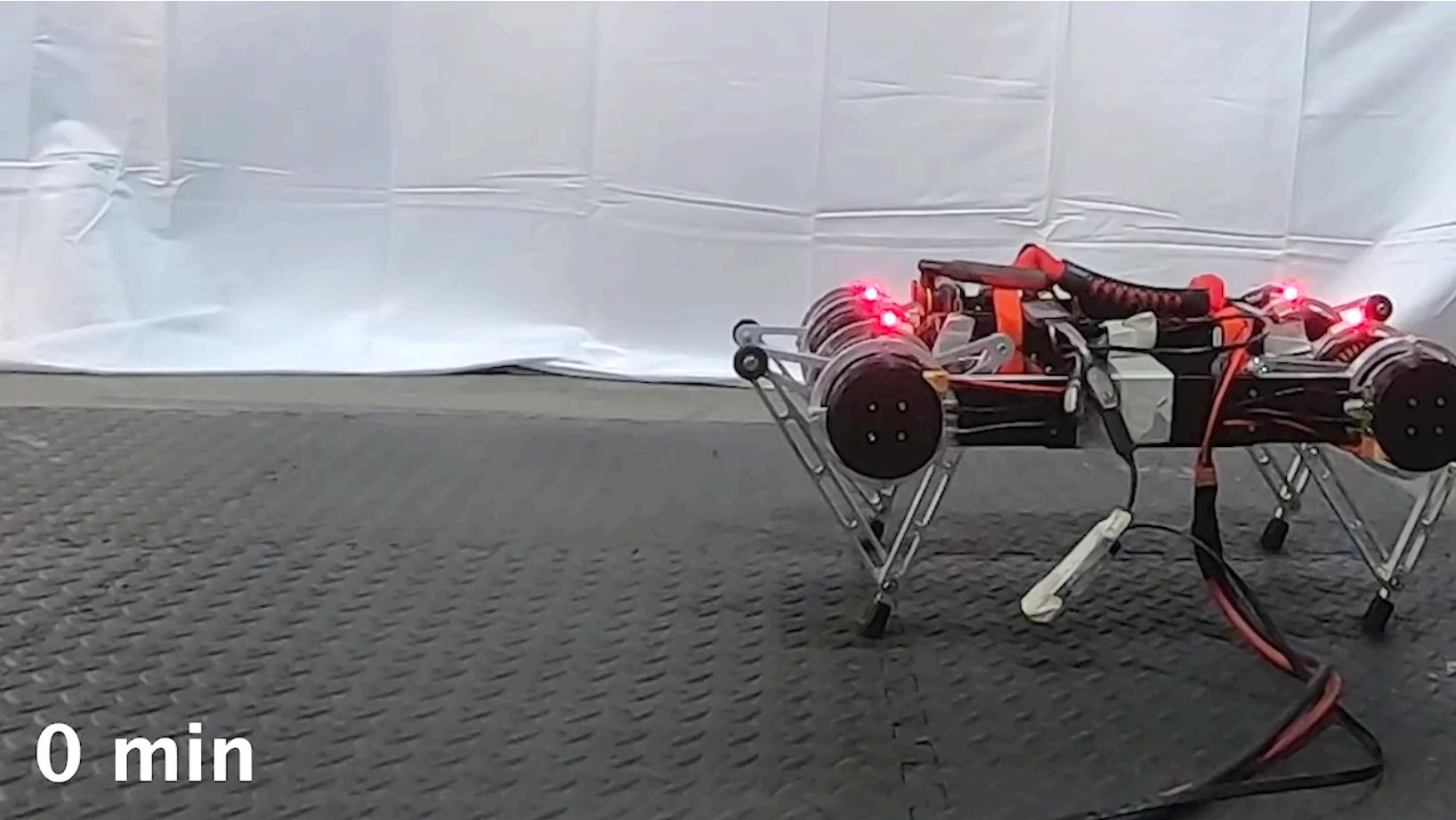
Version 2 (Dec 13 2018)

Disadvantages:

- Performance loss
- Coefficient of entropy term
- Different types of entropy

Results

- Soft Actor-Critic on Minitaur - Training



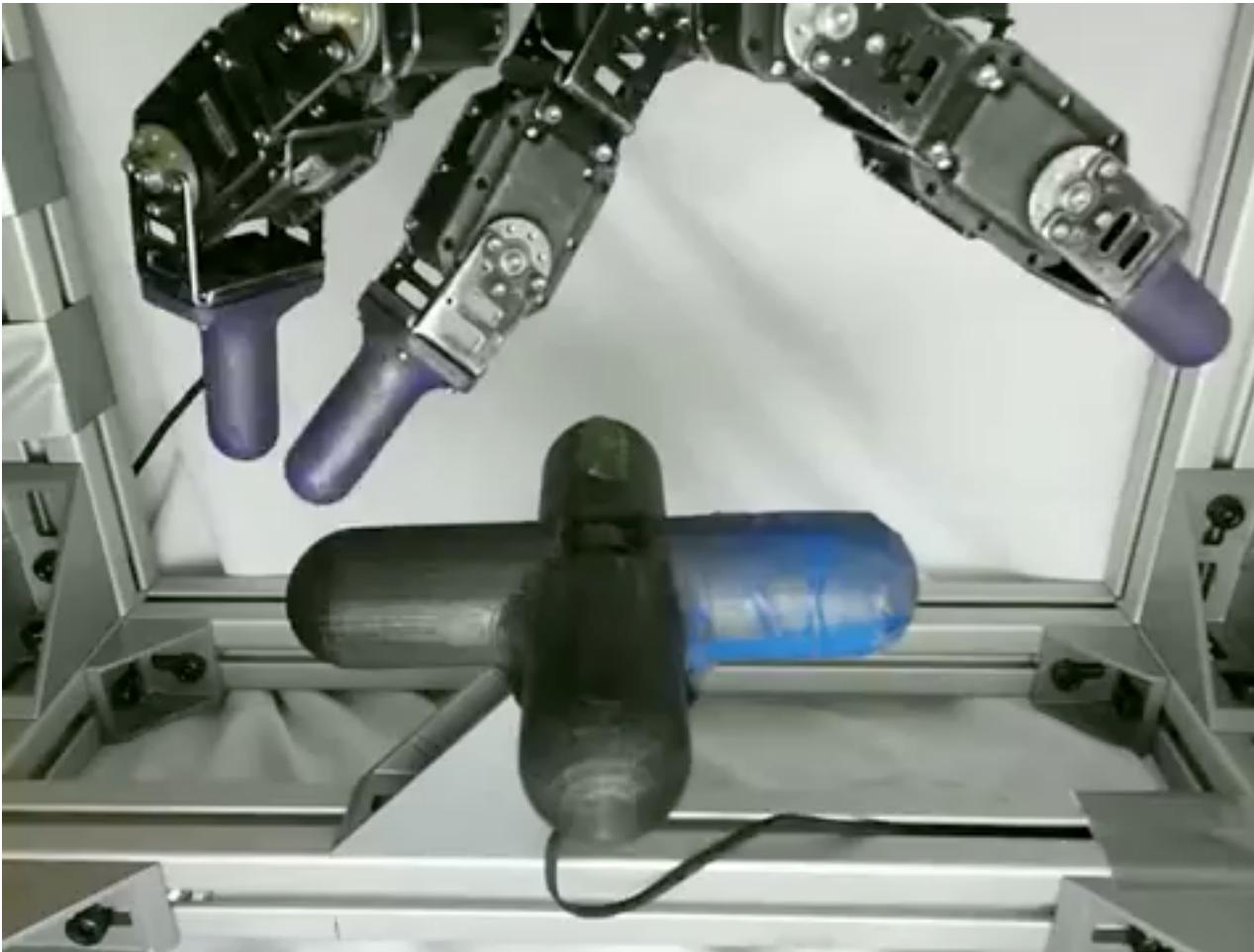
Results

- Soft Actor-Critic on Minitaur - Testing



Results

- Interfered rollouts from Soft Actor-Critic policy trained for Dynamixel Claw task from vision



References

Papers

- [T. Haarnoja, et al., “Reinforcement Learning with Deep Energy-Based Policies”, ICML 2017](#)
- [T. Haarnoja, et, al., “Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor”, ICML 2018](#)
- [T. Haarnoja, et, al., “Soft Actor-Critic Algorithms and Applications”, arXiv preprint 2018](#)

Theoretical analysis

- [“Induction of Q-learning, Soft Q-learning, and Sparse Q-learning” written by Sungjoon Choi](#)
- [“Reinforcement Learning with Deep Energy-Based Policies” reviewed by Kyungjae Lee](#)
- [“Entropy Regularization in Reinforcement Learning \(Stochastic Policy\)” reviewed by Kyungjae Lee](#)
- [Reframing Control as an Inference Problem, CS 294-112: Deep Reinforcement Learning](#)
- [Constrained optimization problem and dual problem for SAC with automatically adjusted temperature \(in Korean\)](#)

Thank You!
Any Questions?