



Trust Region Policy Optimization

Changhoon Jeong
DGU AI lab.
chjeong@dongguk.edu

Introduction

■ Last Seminar : Deep Reinforcement Learning

1. Introduction to Deep Reinforcement Learning
2. Value-based RL & Policy-based RL
3. Policy Gradient
4. Advantage Actor-Critic(A2C)
5. Asynchronous Advantage Actor-Critic(A3C)
6. ~~Trust Region Policy Optimization(TRPO)~~
7. ~~Proximal Policy Optimization(PPO)~~



*A little bit hard mathematics & statistics,
To be presented next!*

Introduction

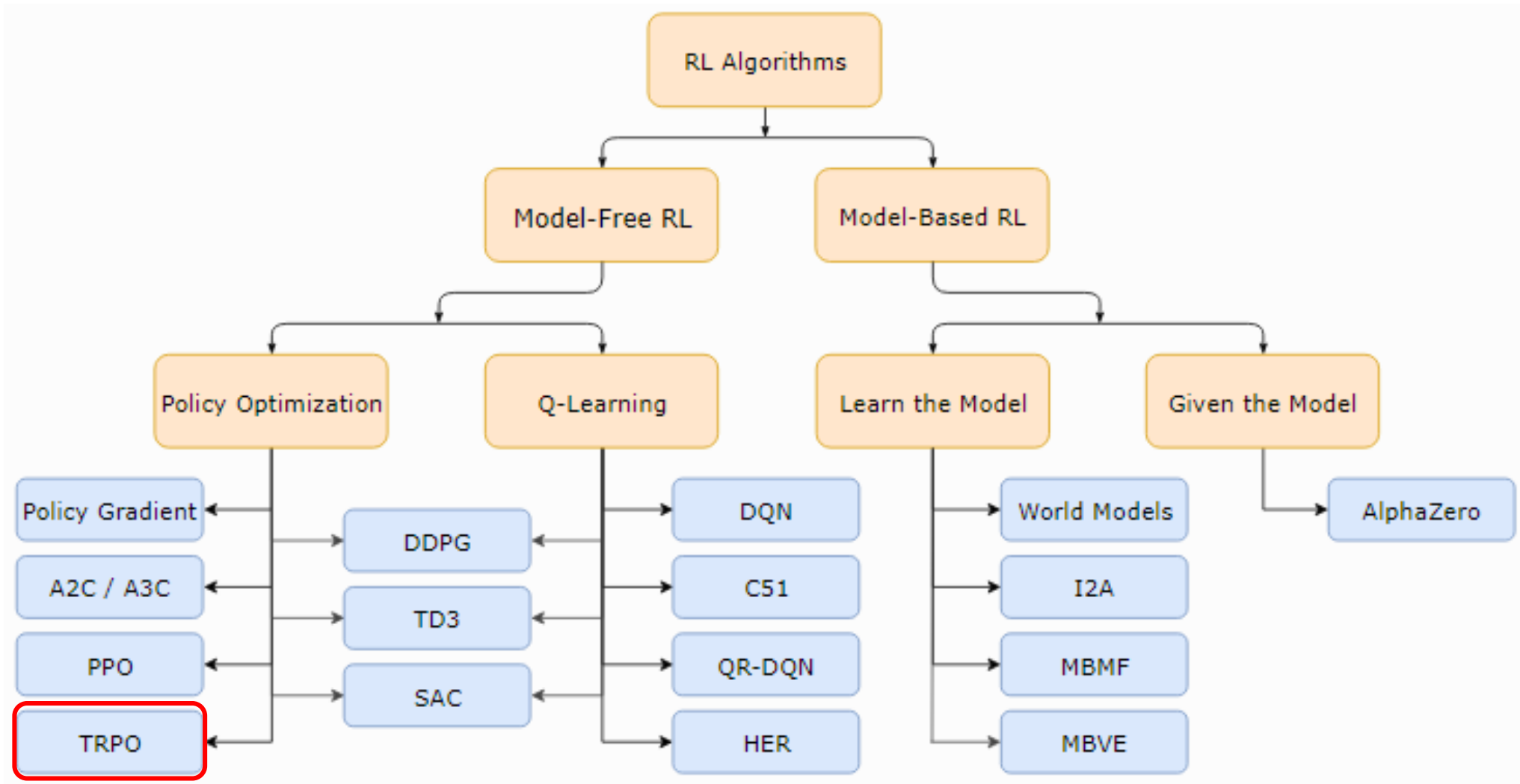
■ Today's Seminar : Deep Reinforcement Learning

1. Introduction to Deep Reinforcement Learning
2. Value-based RL & Policy-based RL
3. Policy Gradient
4. Advantage Actor-Critic(A2C)
5. Asynchronous Advantage Actor-Critic(A3C)
6. Trust Region Policy Optimization(TRPO)
7. Proximal Policy Optimization(PPO)



Hard mathematics!

■ A Taxonomy of RL Algorithms



Trust Region Policy Optimization

John Schulman
Sergey Levine
Philipp Moritz
Michael Jordan
Pieter Abbeel

University of California, Berkeley, Department of Electrical Engineering and Computer Sciences

JOSCHU@EECS.BERKELEY.EDU
SLEVINE@EECS.BERKELEY.EDU
PCMORITZ@EECS.BERKELEY.EDU
JORDAN@CS.BERKELEY.EDU
PABBEEL@CS.BERKELEY.EDU

Abstract

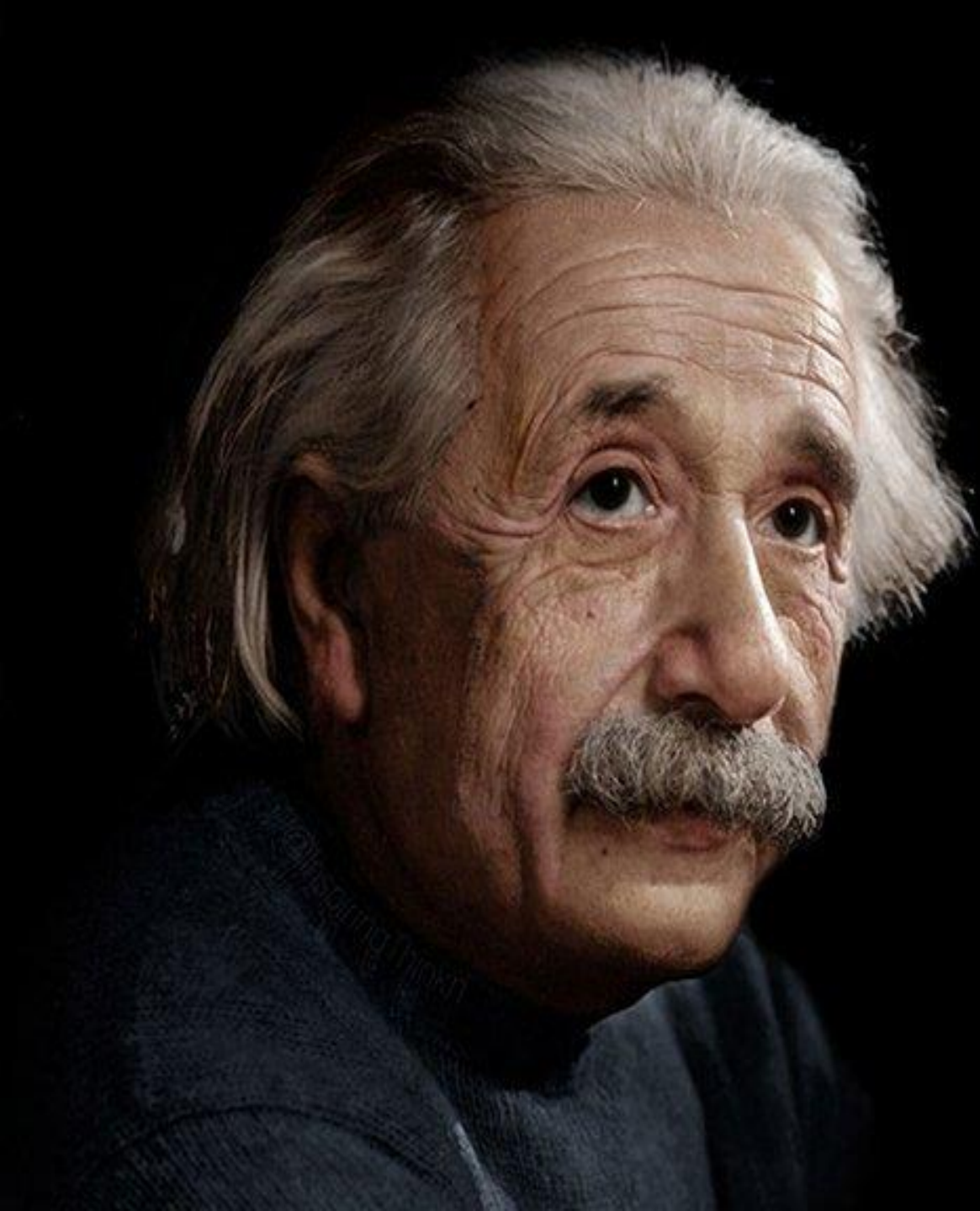
We describe an iterative procedure for optimizing policies, with guaranteed monotonic improvement. By making several approximations to the theoretically-justified procedure, we develop a practical algorithm, called Trust Region Policy Optimization (TRPO). This algorithm is similar to natural policy gradient methods and is effective for optimizing large nonlinear policies such as neural networks. Our experiments demonstrate its robust performance on a wide variety of tasks: learning simulated robotic swimming, hopping, and walking gaits; and playing Atari

Tetris is a classic benchmark problem for approximate dynamic programming (ADP) methods, stochastic optimization methods are difficult to beat on this task (Gabillon et al., 2013). For continuous control problems, methods like CMA have been successful at learning control policies for challenging tasks like locomotion when provided with hand-engineered policy classes with low-dimensional parameterizations (Wampler & Popovic, 2009). The inability of ADP and gradient-based methods to consistently beat gradient-free random search is unsatisfying, since gradient-based optimization algorithms enjoy much better sample complexity guarantees than gradient-free methods (Nemirovski, 2005). Continuous gradient-based optimization has been very successful at learning function approxi-



OpenAI

Schulman, John, et al. "Trust region policy optimization." *International Conference on Machine Learning*. 2015. (cited 1113)



If you can't explain
it simply, you
don't understand it
well enough.

Albert Einstein / @InspiringThinkn

WARNING

This paper has a very theoretical approach and is difficult!
First, I would like to apologize to the junior researchers☹

Later, as you study Reinforcement Learning,
look back at this material when you meet this paper!

Agenda

1. Preliminaries
2. Monotonic Improvement Guarantee for General Stochastic Policies
3. Optimization of Parameterized Policies
4. Sample-Based Estimation of the Objective and Constraint
5. Experiment and Result

1. Preliminaries

- Markov Decision Processes tuple : $(\mathcal{S}, \mathcal{A}, P, r, \rho_0, \gamma)$
 - \mathcal{S} : A finite set of states
 - \mathcal{A} : A finite set of actions
 - P : The transition probability distribution($P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow \mathbb{R}$)
 - r : The reward function($r : \mathcal{S} \rightarrow \mathbb{R}$)
 - ρ_0 : The distribution of the initial state s_0 ($\rho_0 : \mathcal{S} \rightarrow \mathbb{R}$)
 - γ : The discount factor($\gamma \in (0,1)$)
- Policy
 - π : A stochastic policy($\pi : \mathcal{S} \times \mathcal{A} \rightarrow [0,1]$)

Preliminaries

- Expected cumulative discounted reward

$$\eta(\pi) = \mathbb{E}_{s_0, a_0, \dots} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t) \right],$$

where $s_0 \sim \rho_0(s_0), a_t \sim \pi(a_t | s_t), s_{t+1} \sim P(s_{t+1} | s_t, a_t)$

- State-action value function Q_π

$$Q_\pi(s_t, a_t) = \mathbb{E}_{s_{t+1}, a_{t+1}, \dots} \left[\sum_{l=0}^{\infty} \gamma^l r(s_{t+l}) \right]$$

- State value function V_π

$$V_\pi(s_t) = \mathbb{E}_{a_t, s_{t+1}, \dots} \left[\sum_{l=0}^{\infty} \gamma^l r(s_{t+l}) \right]$$

- Advantage function A_π

$$A_\pi(s, a) = Q_\pi(s, a) - V_\pi(s),$$

where $a_t \sim \pi(a_t | s_t), s_{t+1} \sim P(s_{t+1} | s_t, a_t)$ for $t \geq 0$

- Useful identity, Kakade & Langford(2002), Appendix A

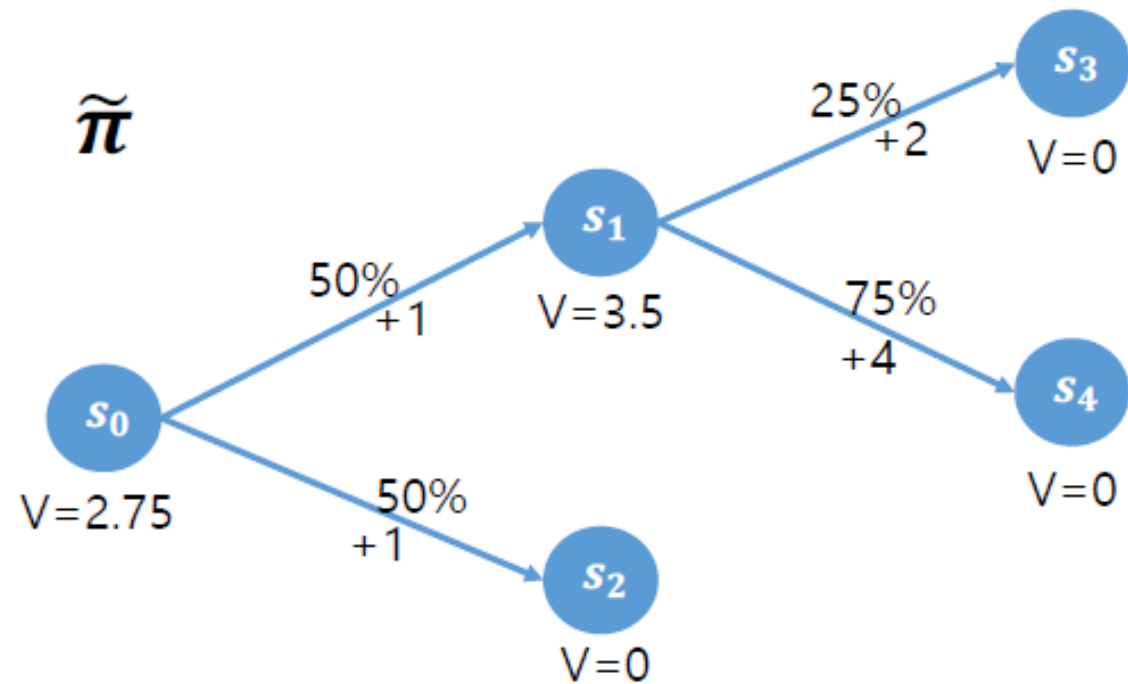
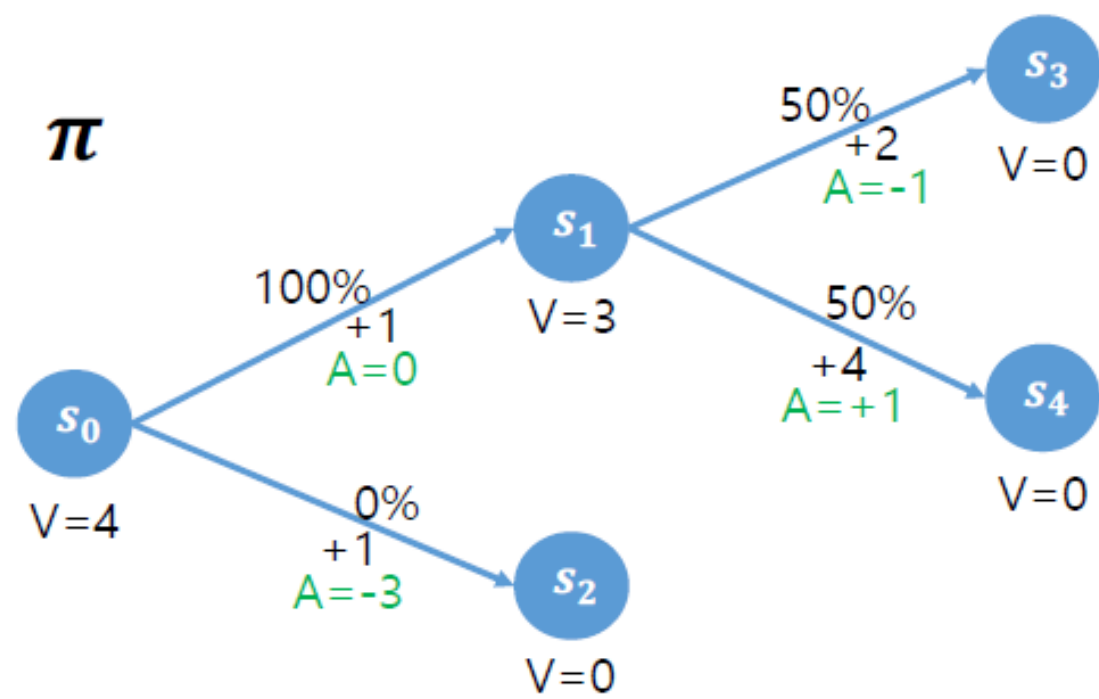
$$\eta(\tilde{\pi}) = \eta(\pi) + \mathbb{E}_{s_0, a_0, \dots \sim \tilde{\pi}} \left[\sum_{t=0}^{\infty} \gamma^t A_{\pi}(s_t, a_t) \right],$$

where $\mathbb{E}_{s_0, a_0, \dots \sim \tilde{\pi}}[\dots]$ indicates that actions are sampled $a_t \sim \tilde{\pi}(\cdot | s_t)$

- (Unnormalized) Discounted visitation frequencies

$$\rho_{\pi}(s) = P(s_0 = s) + \gamma P(s_1 = s) + \gamma^2 P(s_2 = s) + \dots,$$

where $s_0 \sim \rho_0$ and the actions are chosen according to π



$$\eta(\tilde{\pi}) = \eta(\pi) + \mathbb{E}_{s_0, a_0, \dots \sim \tilde{\pi}} \left[\sum_{t=0}^{\infty} \gamma^t A_{\pi}(s_t, a_t) \right]$$


좌변 = 2.75

우변 = $4 + 0.5 * -3 + 0.5 * (0 + 0.25 * -1 + 0.75 * 1)$
 $= 4 - 1.5 + 0.5 * 0.5$
 $= 2.5 + 0.25$
 $= 2.75$

- We can rewrite Equation (1) with **Sum over states** instead **timestep**

$$\eta(\tilde{\pi}) = \eta(\pi) + \mathbb{E}_{s_0, a_0, \dots \sim \tilde{\pi}} \left[\sum_{t=0}^{\infty} \gamma^t A_{\pi}(s_t, a_t) \right]$$

Sum over timestep


$$\eta(\tilde{\pi}) = \eta(\pi) + \sum_{t=0}^{\infty} \sum_s P(s_t = s | \tilde{\pi}) \sum_a \tilde{\pi}(a|s) \gamma^t A_{\pi}(s, a)$$

$$= \eta(\pi) + \sum_s \sum_{t=0}^{\infty} \gamma^t P(s_t = s | \tilde{\pi}) \sum_a \tilde{\pi}(a|s) A_{\pi}(s, a)$$

$$= \eta(\pi) + \sum_s \rho_{\tilde{\pi}}(s) \sum_a \tilde{\pi}(a|s) A_{\pi}(s, a)$$

Sum over states

- So, what does that mean?

$$\eta(\tilde{\pi}) = \eta(\pi) + \sum_s \rho_{\tilde{\pi}}(s) \sum_a \tilde{\pi}(a|s) A_{\pi}(s, a)$$

- Any policy update $\pi \rightarrow \tilde{\pi}$ that has a nonnegative expected advantage at every state s , i.e. $\sum_a \tilde{\pi}(a|s) A_{\pi}(s, a) \geq 0$, is **guaranteed to increase policy performance η** , or leave it constant in the case that the expected advantage is zero everywhere
- e.g. policy iteration


$$\tilde{\pi}(s) = \operatorname{argmax}_a A_{\pi}(s, a)$$

- But, in the approximate setting, there are estimation and approximation error, some states s have negative expected advantage

$$\sum_a \tilde{\pi}(a|s) A_{\pi}(s, a) < 0$$

- The complex dependency of $\rho_{\tilde{\pi}}(s)$ on $\tilde{\pi}$ makes Equation (2) difficult to optimize directly. Instead, we introduce the following local approximation to η :

$$\eta(\tilde{\pi}) = \eta(\pi) + \sum_s \rho_{\tilde{\pi}}(s) \sum_a \tilde{\pi}(a|s) A_{\pi}(s, a)$$


$$L_{\pi}(\tilde{\pi}) = \eta(\pi) + \sum_s \rho_{\pi}(s) \sum_a \tilde{\pi}(a|s) A_{\pi}(s, a)$$

- L_{π} uses the visitation frequency $\rho_{\pi}(s)$ rather than $\rho_{\tilde{\pi}}(s)$, ignoring changes in state visitation density due to changes in the policy
- However, if we have parameterized policy π_{θ} (differentiable), then L_{π} matched η to first order (Kakade & Langford (2002))
- That is, for any parameter value θ_0 ,

$$\begin{aligned} L_{\pi_{\theta_0}}(\pi_{\theta_0}) &= \eta(\pi_{\theta_0}), \\ \nabla_{\theta} L_{\pi_{\theta_0}}(\pi_{\theta})|_{\theta=\theta_0} &= \nabla_{\theta} \eta(\pi_{\theta})|_{\theta=\theta_0} \end{aligned}$$



Danger
Zone



Tribes

Starting
Point



Thorny Patch



Snake's
Den



Abandoned Forest



Wild Bufaloes



Destination



Waterfall

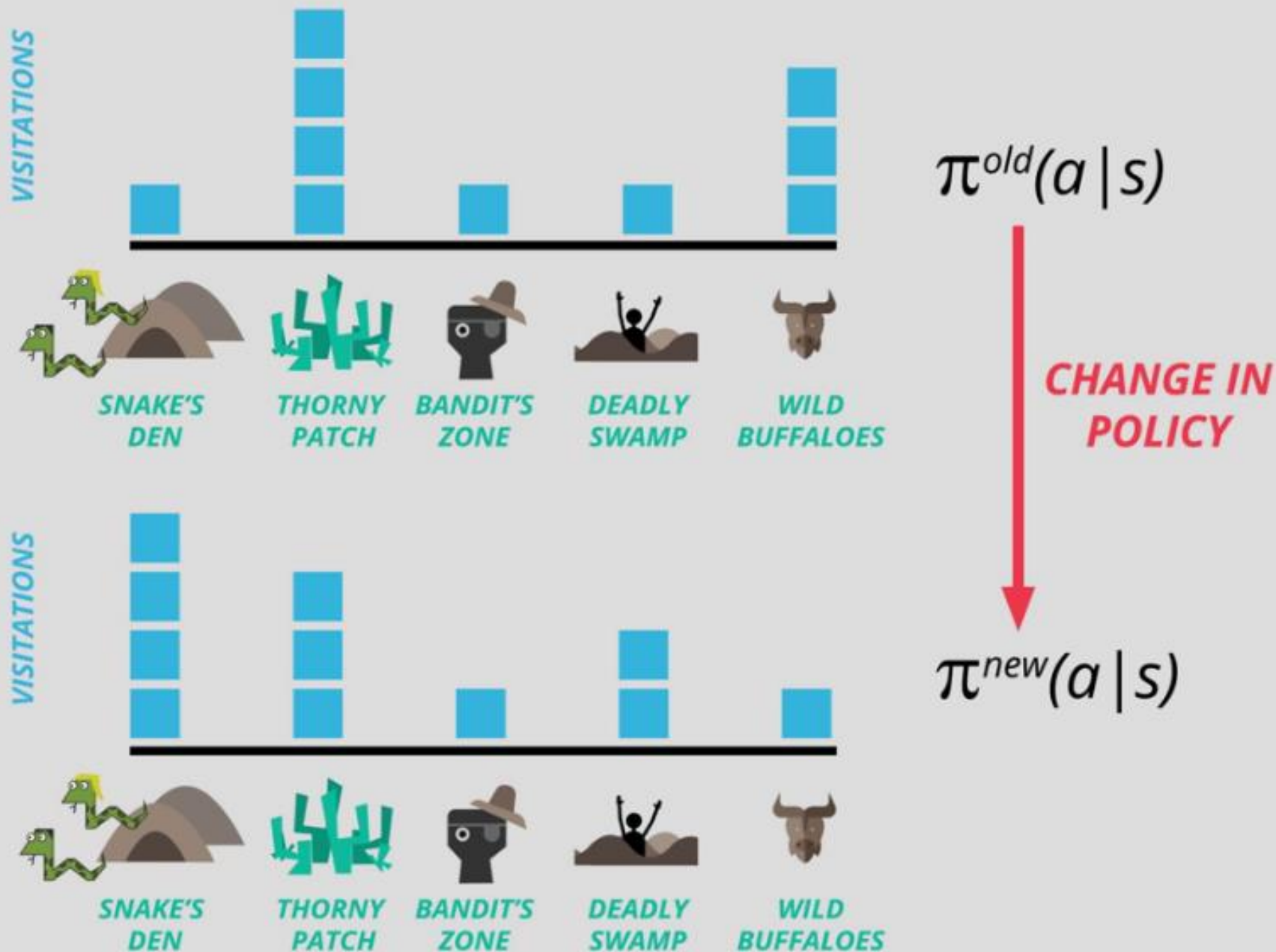


Sea
Beyond

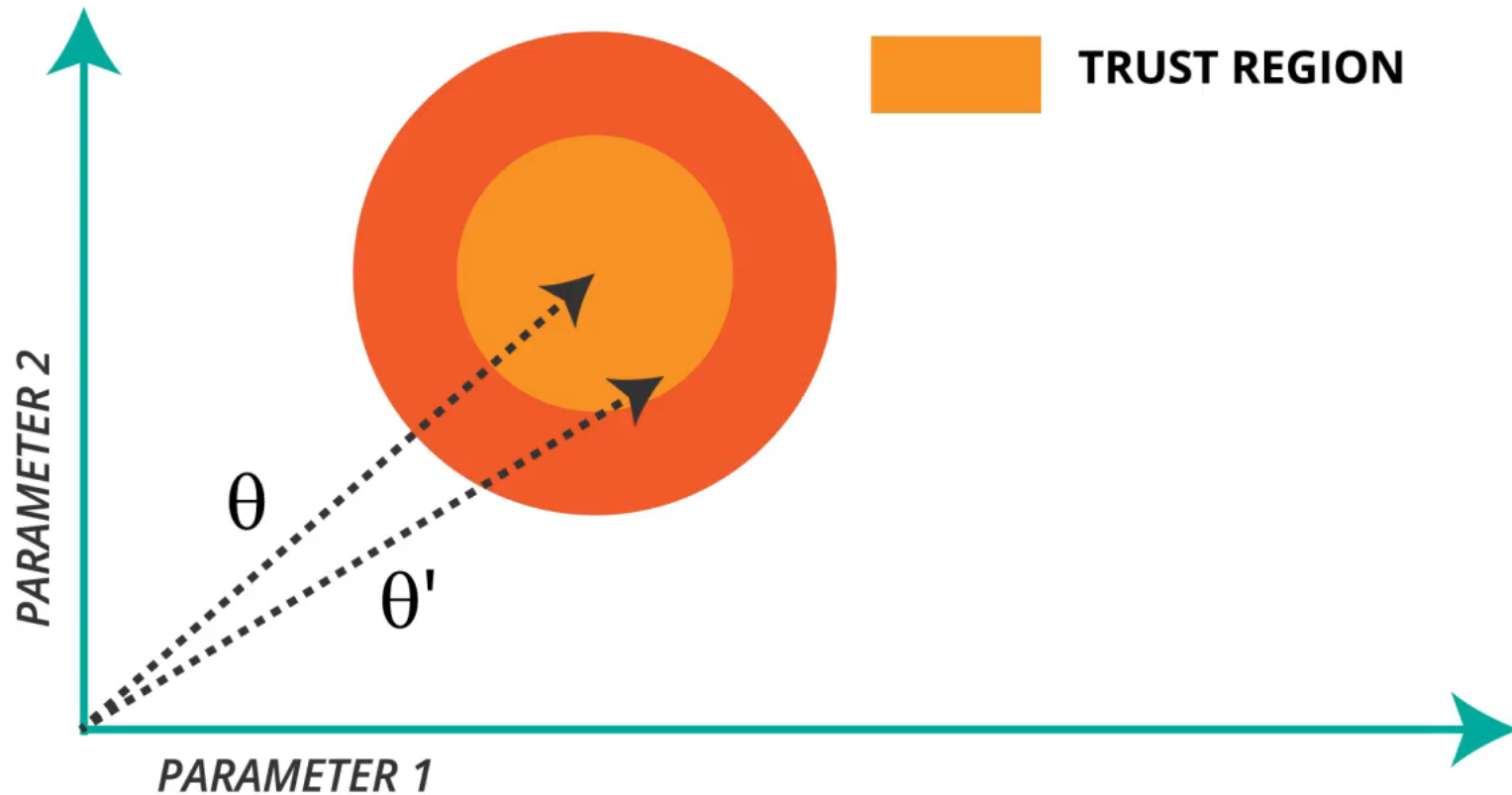


Deadly
Swamp

STATE VISITATION FREQUENCY



THAT'S WHY - TRUST REGIONS



$\pi_{\theta'}(s | a)$ DOESN'T CHANGE DRAMATICALLY

- So, what does that mean?

$$L_{\pi_{\theta_0}}(\pi_{\theta_0}) = \eta \pi_{\theta_0}$$
$$\nabla_{\theta} L_{\pi_{\theta_0}}(\pi_{\theta})|_{\theta=\theta_0} = \nabla_{\theta} \eta(\pi_{\theta})|_{\theta=\theta_0}$$

- A sufficiently small step $\pi_{\theta_0} \rightarrow \tilde{\pi}$ that improves $L_{\pi_{\theta_{old}}}$ will also improve η
- But, it does not give us any guidance on how big of a step to take...
- To address this issue, Kakade & Langford(2002) again...!

- Conservative policy iteration

$$\pi_{new}(a|s) = (1 - \alpha)\pi_{old}(a|s) + \alpha\pi^*(a|s)$$

$$\pi_{old} : \text{current policy}$$
$$\pi^* = \operatorname{argmax}_{\pi} L_{\pi_{old}}(\pi)$$

- Conservative policy iteration

$$\pi_{new}(a|s) = (1 - \alpha)\pi_{old}(a|s) + \alpha\pi'(a|s)$$

- Kakade and Langford derived the following lower bound:

$$\eta(\pi_{new}) \geq L_{\pi_{old}}(\pi_{new}) - \frac{2\epsilon\gamma}{(1-\gamma)^2} \alpha^2$$

where $\epsilon = \max_s |\mathbb{E}_{a \sim \pi'}(A_{\pi}(s, a))|$.

- But, this bound only applies to mixture policies
- It is desirable for a practical policy update scheme to be applicable to all general stochastic policy classes! (Finally, we are ready to see the TRPO...!)

2. Monotonic Improvement Guarantee for General Stochastic Policies

Monotonic Improvement Guarantee for General Stochastic Policies

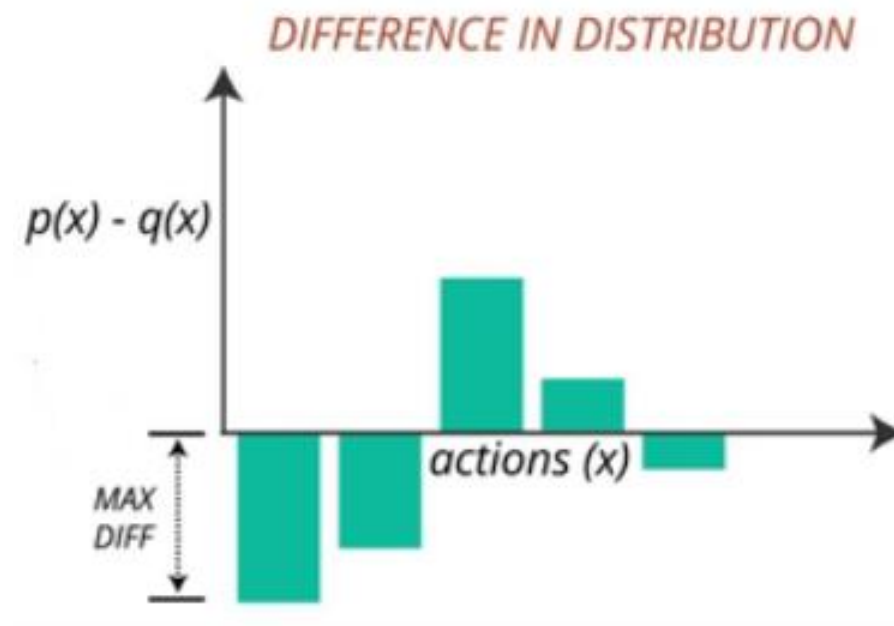
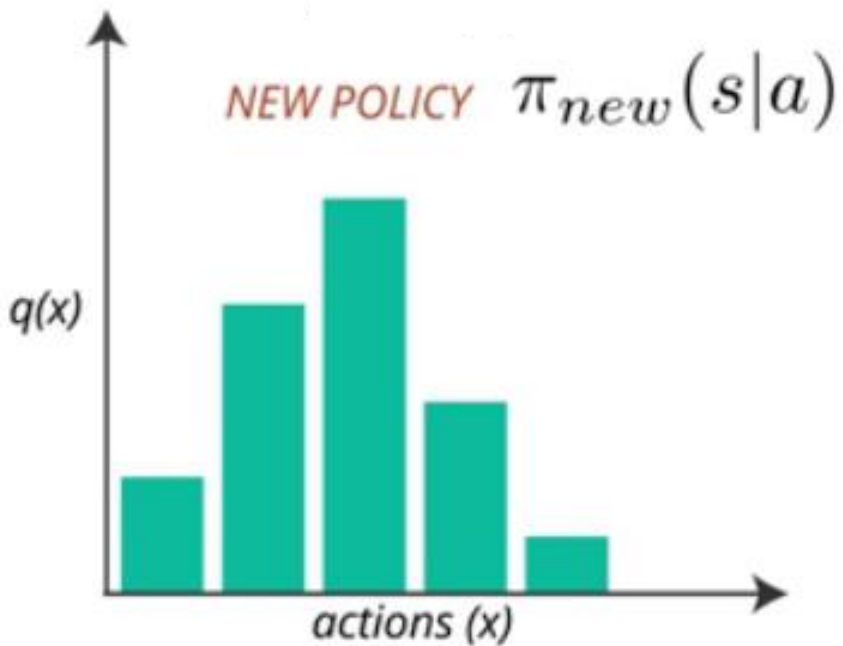
- By replacing α with distance measure between π and $\tilde{\pi}$, and changing the ϵ appropriately, we can extend Equation(6) to general stochastic policies
 - Total Variation Divergence

$$D_{TV}(p||q) = \frac{1}{2} \sum_i |p_i - q_i|$$
$$D_{TV}^{max}(\pi, \tilde{\pi}) = \max_s D_{TV}(\pi(\cdot | s) || \tilde{\pi}(\cdot | s))$$

Theorem 1. *Let $\alpha = D_{TV}^{max}(\pi_{old}, \pi_{new})$. Then the following bound holds:*

$$\eta(\pi_{new}) \geq L_{\pi_{old}}(\pi_{new}) - \frac{4\epsilon\gamma}{(1-\gamma)^2} \alpha^2,$$

where $\epsilon = \max_{s,a} |A_{\pi}(s, a)|$




$$D_{TV}^{max}(\pi, \tilde{\pi}) = \max_s D_{TV}(\pi(\cdot | s) || \tilde{\pi}(\cdot | s))$$

Monotonic Improvement Guarantee for General Stochastic Policies

- Kakade & Langford → John Schulman

$$\eta(\pi_{new}) \geq L_{\pi_{old}}(\pi_{new}) - \frac{2\epsilon\gamma}{(1-\gamma)^2} \alpha^2,$$

where $\epsilon = \max_s |\mathbb{E}_{a \sim \pi} (a|s) [A_{\pi}(s, a)]|$.


$$\eta(\pi_{new}) \geq L_{\pi_{old}}(\pi_{new}) - \frac{4\epsilon\gamma}{(1-\gamma)^2} \alpha^2,$$

where $\epsilon = \max_{s,a} |A_{\pi}(s, a)|$

- Additionally, Pollard(2000)

$$D_{TV}(p||q)^2 \leq D_{KL}(p||q)$$

$$\text{Let } D_{KL}^{max}(\pi, \tilde{\pi}) = \max_s D_{KL}(\pi(\cdot | s) || \tilde{\pi}(\cdot | s))$$

So, we get :

$$\eta(\tilde{\pi}) \geq L_{\pi}(\tilde{\pi}) - C D_{KL}^{max}(\pi, \tilde{\pi}),$$

$$\text{where } C = \frac{4\epsilon\gamma}{(1-\gamma)^2}$$

Monotonic Improvement Guarantee for General Stochastic Policies

- Policy iteration algorithm guaranteeing non-decreasing expected return η

Algorithm 1 Policy iteration algorithm guaranteeing non-decreasing expected return η

Initialize π_0 .

for $i = 0, 1, 2, \dots$ until convergence **do**

 Compute all advantage values $A_{\pi_i}(s, a)$.

 Solve the constrained optimization problem

$$\pi_{i+1} = \arg \max_{\pi} [L_{\pi_i}(\pi) - CD_{\text{KL}}^{\max}(\pi_i, \pi)]$$

$$\text{where } C = 4\epsilon\gamma/(1 - \gamma)^2$$

$$\text{and } L_{\pi_i}(\pi) = \eta(\pi_i) + \sum_s \rho_{\pi_i}(s) \sum_a \pi(a|s) A_{\pi_i}(s, a)$$

end for

- The Algorithm uses a constraint on the KL divergence rather than a penalty to robustly allow large updates

Monotonic Improvement Guarantee for General Stochastic Policies

- Does it guarantee to generate a monotonically improving sequence of policies $\eta(\pi_0) \leq \eta(\pi_1) \leq \eta(\pi_2) \leq \dots$? **Yes!**

- To see this, let $M_i(\pi) = L_{\pi_i}(\pi) - CD_{KL}^{max}(\pi_i, \pi)$: *surrogate function*, Then,

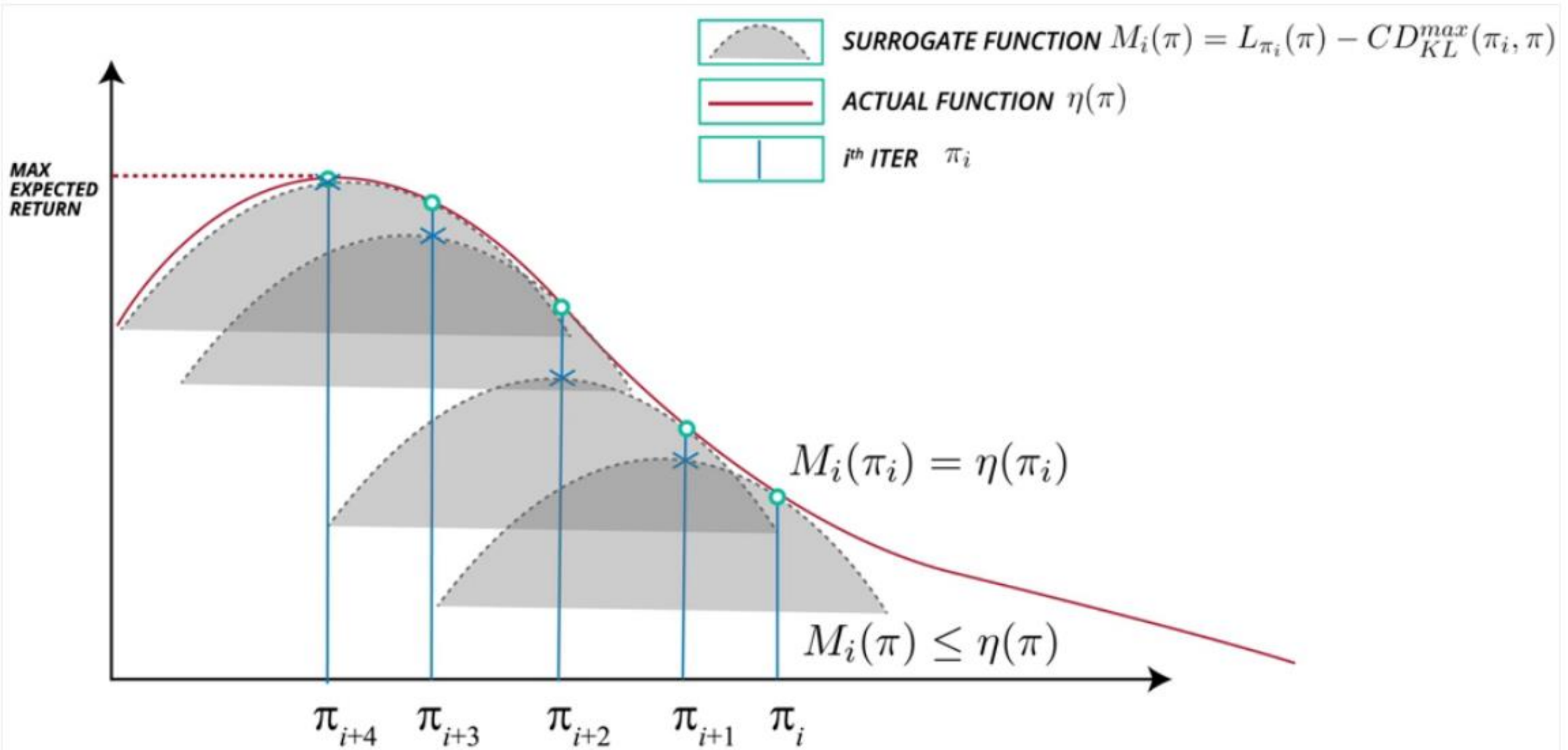
$$\eta(\pi_{i+1}) \geq M_i(\pi_{i+1}) \text{ by Equation(9)}$$

$$\eta(\pi_i) = M_i(\pi_i), \text{ therefore,}$$

$$\eta(\pi_{i+1}) - \eta(\pi_i) \geq M_i(\pi_{i+1}) - M_i(\pi_i)$$

- Thus, by maximizing M_i , at each iteration, we guarantee that the true objective η is non-decreasing

Algorithm 1



3. Optimization of Parameterized Policies

Optimization of Parameterized Policies

- Consider parameterized policies $\pi_{\theta}(a|s)$
 - So change the notations

$$\eta(\theta) := \eta(\pi_{\theta})$$

$$L_{\theta}(\tilde{\theta}) := L_{\pi_{\theta}}(\pi_{\tilde{\theta}})$$

$$D_{KL}(\theta || \tilde{\theta}) := D_{KL}(\pi_{\theta} || \pi_{\tilde{\theta}})$$

- True objective η that we are guaranteed to improve :

$$\underset{\theta}{\text{maximize}} [L_{\theta_{old}}(\theta) - c D_{KL}^{max}(\theta_{old}, \theta)]$$

Optimization of Parameterized Policies

- We want to take larger steps in robust way, but there is a problem

$$\underset{\theta}{\text{maximize}} [L_{\theta_{old}}(\theta) - C D_{KL}^{max}(\theta_{old}, \theta)], \text{ where } C = \frac{4\epsilon\gamma}{(1-\gamma)^2}$$

- C is very large number (consider when $\gamma = 0.99$)
- So step size should be smaller...
- One way to take larger steps is to use a **constraint** on the KL divergence between the new policy and the old policy, i.e., a trust region constraint :

$$\begin{aligned} & \underset{\theta}{\text{maximize}} L_{\theta_{old}}(\theta) \\ & \text{subject to } D_{KL}^{max}(\theta_{old}, \theta) \leq \delta \end{aligned}$$

Optimization of Parameterized Policies

- But... we have another problem

$$\begin{aligned} & \underset{\theta}{\text{maximize}} \quad L_{\theta_{old}}(\theta) \\ & \text{subject to} \quad D_{KL}^{max}(\theta_{old}, \theta) \leq \delta \end{aligned}$$

- It imposes a constraint that the **KL divergence is bounded at every point** in the state space
- Motivated by the theory, but impractical!
- Instead, we can use a heuristic approximation : the average KL divergence

$$\bar{D}_{KL}^{\rho}(\theta_1, \theta_2) := \mathbb{E}_{s \sim \rho} [D_{KL}(\pi_{\theta_1}(\cdot | s) || \pi_{\theta_2}(\cdot | s))].$$


$$\begin{aligned} & \underset{\theta}{\text{maximize}} \quad L_{\theta_{old}}(\theta) \\ & \text{subject to} \quad \bar{D}_{KL}^{\rho_{\pi_{old}}}(\theta_{old}, \theta_{new}) \leq \delta \end{aligned}$$

4. Sample-Based Estimation of the Objective and Constraint

Sample-Based Estimation of the Objective and Constraint

- Ok, How to change the Objective and Constraint to sample-based monte-carlo estimation?

$$\begin{aligned} & \text{maximize}_{\theta} L_{\theta_{old}}(\theta) \\ & \text{subject to } \bar{D}_{KL}^{\rho\pi_{old}}(\theta_{old}, \theta_{new}) \leq \delta \end{aligned}$$


$$\begin{aligned} & \text{maximize}_{\theta} \sum_s \rho_{\theta_{old}}(s) \sum_a \pi_{\theta}(a|s) A_{\theta_{old}}(s, a) \\ & \text{subject to } \bar{D}_{KL}^{\rho\pi_{old}}(\theta_{old}, \theta_{new}) \leq \delta \end{aligned}$$

- Let's take some useful steps :

- Replace $\sum_s \rho_{\pi_{old}}(s) [\dots]$ by the expectation $\frac{1}{1-\gamma} \mathbb{E}_{s \sim \rho_{\pi_{old}}} [\dots]$
- Replace advantage values $A_{\theta_{old}}$ by the Q-values $Q_{\theta_{old}}$
- Replace the sum over the actions by an **importance sampling** estimator

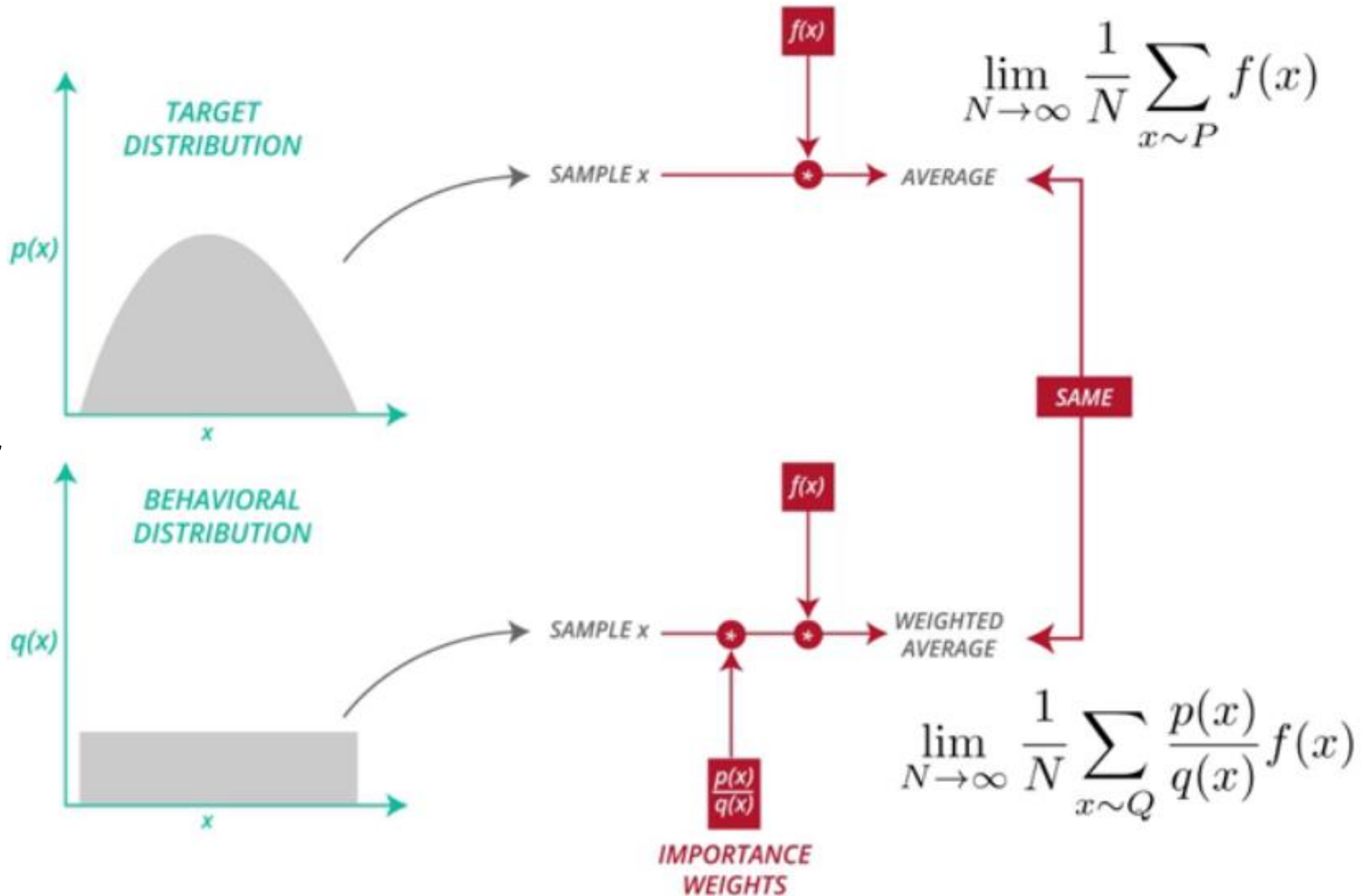
IMPORTANCE SAMPLING

$$\mathbb{E}_{x \sim p}[f(x)]$$

$$= \int p(x) f(x) dx$$

$$= \int \frac{p(x)}{q(x)} q(x) f(x) dx$$

$$= \mathbb{E}_{x \sim q} \left[\frac{p(x)}{q(x)} f(x) \right]$$



Sample-Based Estimation of the Objective and Constraint

- Let's take some useful steps :

- Replace $\sum_s \rho_{\pi_{old}}(s) [\dots]$ by the expectation $\frac{1}{1-\gamma} \mathbb{E}_{s \sim \rho_{\pi_{old}}} [\dots]$
- Replace advantage values $A_{\theta_{old}}$ by the Q-values $Q_{\theta_{old}}$
- Replace the sum over the actions by an **importance sampling** estimator

$$\sum_a \pi_{\theta}(a|s_n) A_{\theta_{old}}(s_n, a) = \mathbb{E}_{a \sim q} \left[\frac{\pi_{\theta}(a|s_n)}{q(a|s_n)} A_{\theta_{old}}(s_n, a) \right]$$

$q(\text{old policy})$: sampling distribution

- So we can write the formula in terms of expectations :

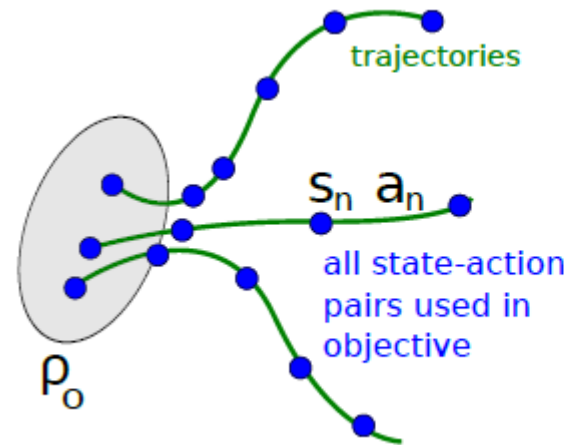
$$\begin{aligned} & \underset{\theta}{\text{maximize}} \mathbb{E}_{s \sim \rho_{\pi_{old}}, a \sim q} \left[\frac{\pi_{\theta}(a|s)}{q(a|s)} Q_{\theta_{old}}(s, a) \right] \\ & \text{subject to } \mathbb{E}_{s \sim \rho_{\pi_{old}}} [D_{KL}(\pi_{\theta_{old}}(\cdot | s) || \pi_{\theta}(\cdot | s))] \leq \delta \end{aligned}$$

Sample-Based Estimation of the Objective and Constraint

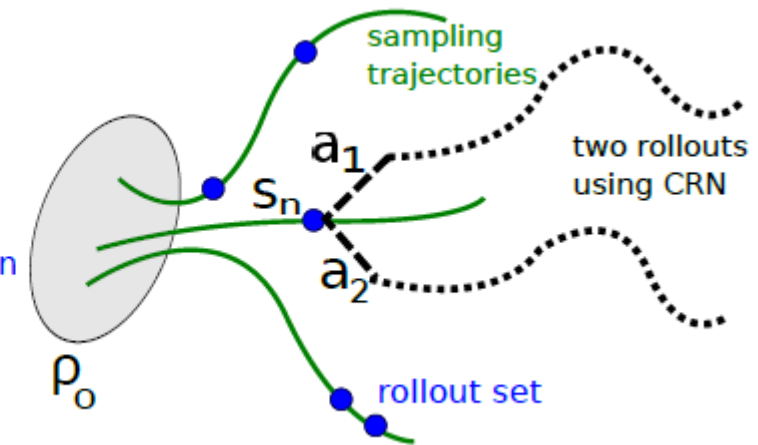
■ Single Path

- Collect a sequence of states by sampling $s_0 \sim \rho_0$
- Generate some number of timesteps' trajectory using $\pi_{\theta_{old}} = q$
 $s_0, a_0, s_1, a_1, \dots, s_{T-1}, a_{T-1}, s_T$
- $Q_{\theta_{old}}(s, a)$ is computed at each state-action pair (s_t, a_t) by taking the discounted sum of future rewards

$$\begin{aligned} & \underset{\theta}{\text{maximize}} \mathbb{E}_{s \sim \rho_{\pi_{old}}, a \sim q} \left[\frac{\pi_{\theta}(a|s)}{q(a|s)} Q_{\theta_{old}}(s, a) \right] \\ & \text{subject to } \mathbb{E}_{s \sim \rho_{\pi_{old}}} [D_{KL}(\pi_{\theta_{old}}(\cdot|s) || \pi_{\theta}(\cdot|s))] \leq \delta \end{aligned}$$



Single Path

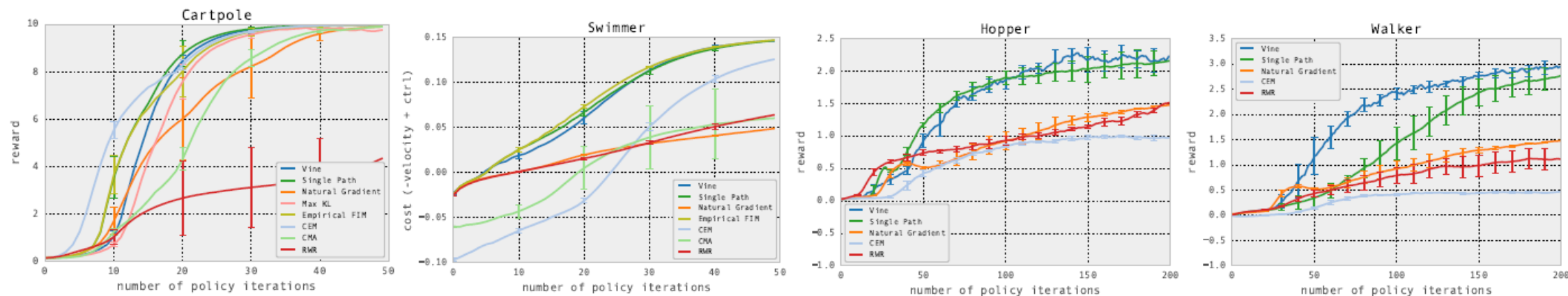


Vine

5. Experiment and Result

Experiment and Result

■ Learning curves for locomotion tasks

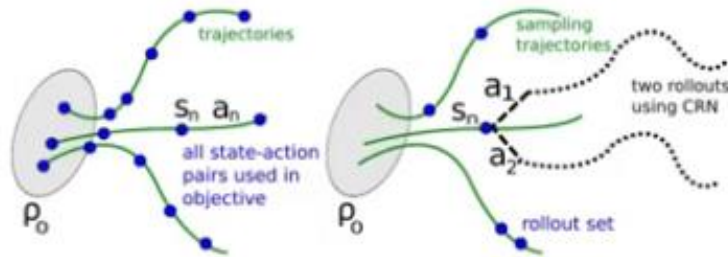


■ Vision-based RL algorithms on the Atari domain

	<i>B. Rider</i>	<i>Breakout</i>	<i>Enduro</i>	<i>Pong</i>	<i>Q*bert</i>	<i>Seaquest</i>	<i>S. Invaders</i>
Random	354	1.2	0	-20.4	157	110	179
Human (Mnih et al., 2013)	7456	31.0	368	-3.0	18900	28010	3690
Deep Q Learning (Mnih et al., 2013)	4092	168.0	470	20.0	1952	1705	581
UCC-I (Guo et al., 2014)	5702	380	741	21	20025	2995	692
TRPO - single path	1425.2	10.8	534.6	20.9	1973.5	1908.6	568.4
TRPO - vine	859.5	34.2	430.8	20.9	7732.5	788.4	450.2

Summary

1. Use the *single path* or *vine* procedures to collect a set of state-action pairs along with Monte Carlo estimates of their Q -values.



2. By averaging over samples, construct the estimated objective and constraint in Equation (14).

$$\begin{aligned} & \underset{\theta}{\text{maximize}} \mathbb{E}_{s \sim \rho_{\theta_{\text{old}}}, a \sim q} \left[\frac{\pi_{\theta}(a|s)}{q(a|s)} Q_{\theta_{\text{old}}}(s, a) \right] \\ & \text{subject to } \mathbb{E}_{s \sim \rho_{\theta_{\text{old}}}} [D_{\text{KL}}(\pi_{\theta_{\text{old}}}(\cdot|s) \parallel \pi_{\theta}(\cdot|s))] \leq \delta. \end{aligned} \quad (14)$$

3. Approximately solve this constrained optimization problem to update the policy's parameter vector θ . We use the conjugate gradient algorithm followed by a line search, which is altogether only slightly more expensive than computing the gradient itself. See Appendix C for details.

$$\frac{1}{N} \sum_{n=1}^N \frac{\partial^2}{\partial \theta_i \partial \theta_j} D_{\text{KL}}(\pi_{\theta_{\text{old}}}(\cdot|s_n) \parallel \pi_{\theta}(\cdot|s_n)).$$

Find Monte Carlo Estimates of
 Q values for (s,a) samples

Plug the calculated Q values
+
Plug old action prob for KL Div

Policy update directions are
conjugate w.r.t F.I.M
(Fisher Information Matrix)

Reference

- <https://www.youtube.com/watch?v=CKaN5PgkSBc&t=90s>
- <https://www.youtube.com/watch?v=XBO4oPChMfl>
- <https://www.youtube.com/watch?v=CKaN5PgkSBc&feature=youtu.be&t=4m35s>
- https://reinforcement-learning-kr.github.io/2018/06/24/5_trpo/
- <https://arxiv.org/abs/1502.05477>
- <http://www.cs.toronto.edu/~tingwuwang/trpo.pdf>
- <http://rll.berkeley.edu/deeprlcourse/docs/lec5.pdf>



Thank you!

“Question everything generally thought to be obvious”

- Dieter Rams, 1932 -