



**POLITECNICO**  
MILANO 1863

# Retrace( $\lambda$ )

Temporal Credit Assignment in Off-Policy Reinforcement Learning

Matteo Papini

28th November 2017



- 1 Introduction
- 2 Eligibility Traces
- 3 Off-policy Credit Assignment
- 4 Retrace( $\lambda$ )
- 5 Experiments

## 1 Introduction

## 2 Eligibility Traces

## 3 Off-policy Credit Assignment

## 4 Retrace( $\lambda$ )

## 5 Experiments



- **Prediction:** measure the performance of a given policy  $\pi$
- **Control:** find the optimal policy  $\pi^*$

- **Prediction:** measure the performance of a given policy  $\pi$
- **Control:** find the optimal policy  $\pi^*$

Bellman expectation equation:

$$Q^\pi(s, a) = R(s, a) + \gamma \underset{\substack{s' \sim \mathcal{P} \\ a' \sim \pi}}{E} [Q^\pi(s', a')]$$

Bellman expectation **operator**:

$$\mathcal{T}^\pi Q = R(s, a) + \gamma \underset{\substack{s' \sim \mathcal{P} \\ a' \sim \pi}}{E} [Q(s', a')]$$

- $Q^\pi$  is the unique fixed point of  $\mathcal{T}^\pi$
- Contraction property:  $\|\mathcal{T}^\pi Q - Q^\pi\|_\infty \leq \gamma \|Q - Q^\pi\|_\infty$



Bellman expectation equation:

$$Q^\pi(s, a) = R(s, a) + \gamma \mathop{E}_{\substack{s' \sim \mathcal{P} \\ a' \sim \pi}} [Q^\pi(s', a')]$$

Bellman expectation **operator**:

$$\mathcal{T}^\pi Q = R(s, a) + \gamma \mathop{E}_{\substack{s' \sim \mathcal{P} \\ a' \sim \pi}} [Q(s', a')]$$

- $Q^\pi$  is the unique fixed point of  $\mathcal{T}^\pi$
- Contraction property:  $\|\mathcal{T}^\pi Q - Q^\pi\|_\infty \leq \gamma \|Q - Q^\pi\|_\infty$

Bellman expectation equation:

$$Q^\pi(s, a) = R(s, a) + \gamma \underset{\substack{s' \sim \mathcal{P} \\ a' \sim \pi}}{E} [Q^\pi(s', a')]$$

Bellman expectation **operator**:

$$\mathcal{T}^\pi Q = R(s, a) + \gamma \underset{\substack{s' \sim \mathcal{P} \\ a' \sim \pi}}{E} [Q(s', a')]$$

- $Q^\pi$  is the unique fixed point of  $\mathcal{T}^\pi$
- Contraction property:  $\|\mathcal{T}^\pi Q - Q^\pi\|_\infty \leq \gamma \|Q - Q^\pi\|_\infty$

Bellman expectation equation:

$$Q^\pi(s, a) = R(s, a) + \gamma \underset{\substack{s' \sim \mathcal{P} \\ a' \sim \pi}}{E} [Q^\pi(s', a')]$$

Bellman expectation **operator**:

$$\mathcal{T}^\pi Q = R(s, a) + \gamma \underset{\substack{s' \sim \mathcal{P} \\ a' \sim \pi}}{E} [Q(s', a')]$$

- $Q^\pi$  is the unique fixed point of  $\mathcal{T}^\pi$
- Contraction property:  $\|\mathcal{T}^\pi Q - Q^\pi\|_\infty \leq \gamma \|Q - Q^\pi\|_\infty$

Bellman optimality equation:

$$Q^*(s, a) = R(s, a) + \gamma \mathop{E}_{s' \sim \mathcal{P}} \left[ \max_{a' \in \mathcal{A}} Q^*(s', a') \right]$$

Bellman optimality **operator**:

$$\mathcal{T}^* Q = R(s, a) + \gamma \mathop{E}_{s' \sim \mathcal{P}} \left[ \max_{a' \in \mathcal{A}} Q^*(s', a') \right]$$

- $Q^*$  is the unique fixed point of  $\mathcal{T}^*$
- Contraction property:  $\|\mathcal{T}^* Q - Q^*\|_\infty \leq \gamma \|Q - Q^*\|_\infty$

- Optimal **deterministic** policy:

$$\pi^*(s) = \pi_{\text{greedy}(Q^*)}(s) \doteq \arg \max_{a' \in \mathcal{A}} Q^*(s, a')$$

- Maximization as a special case of expectation:

$$\max_{a' \in \mathcal{A}} Q(s, a') = \underset{a' \sim}{E} [Q(s, a')]$$

- Optimal **deterministic** policy:

$$\pi^*(s) = \pi_{\text{greedy}(Q^*)}(s) \doteq \arg \max_{a' \in \mathcal{A}} Q^*(s, a')$$

- Maximization as a special case of expectation:

$$\max_{a' \in \mathcal{A}} Q^*(s, a') = E_{a' \sim \pi^*} [Q^*(s, a')]$$

- Optimal **deterministic** policy:

$$\pi^*(s) = \pi_{\text{greedy}(Q^*)}(s) \doteq \arg \max_{a' \in \mathcal{A}} Q^*(s, a')$$

- Maximization as a special case of expectation:

$$\max_{a' \in \mathcal{A}} Q(s, a') = E_{a' \sim \pi_{\text{greedy}(Q)}} [Q(s, a')]$$

- **Prediction:** given policy  $\pi$ , compute  $Q^\pi$
- **Control:** find  $\pi^*$  such that  $Q^{\pi^*} = Q^*$



- Idea: repeatedly apply  $\mathcal{T}$  to  $Q$
- Approximation: update  $Q$  towards a target
- Prediction: fixed  $\pi$ , keep updating  $Q$
- Control: alternate value updates and policy updates

$$\pi_{k+1} = \pi_{\text{greedy}}(Q_k)$$

- Idea: repeatedly apply  $\mathcal{T}$  to  $Q$
- Approximation: update  $Q$  towards a target

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha [G_t - Q(s_t, a_t)]$$

- Prediction: fixed  $\pi$ , keep updating  $Q$
- Control: alternate value updates and policy updates

$$\pi_{k+1} = \pi_{\text{greedy}}(Q_k)$$

- Idea: repeatedly apply  $\mathcal{T}$  to  $Q$
- Approximation: update  $Q$  towards a target

$$\Delta Q(s_t, a_t) = \alpha [G_t - Q(s_t, a_t)]$$

- Prediction: fixed  $\pi$ , keep updating  $Q$
- Control: alternate value updates and policy updates

$$\pi_{k+1} = \pi_{\text{greedy}}(Q_k)$$

- Idea: repeatedly apply  $\mathcal{T}$  to  $Q$
- Approximation: update  $Q$  towards a target

$$\Delta Q(s_t, a_t) = \alpha [G_t - Q(s_t, a_t)]$$

- Prediction: fixed  $\pi$ , keep updating  $Q$
- Control: alternate value updates and policy updates

$$\pi_{k+1} = \pi_{\text{greedy}}(Q_k)$$

- Idea: repeatedly apply  $\mathcal{T}$  to  $Q$
- Approximation: update  $Q$  towards a target

$$\Delta Q(s_t, a_t) = \alpha [G_t - Q(s_t, a_t)]$$

- Prediction: fixed  $\pi$ , keep updating  $Q$
- Control: alternate value updates and policy updates

$$\pi_{k+1} = \pi_{\text{greedy}}(Q_k)$$

- Idea: repeatedly apply  $\mathcal{T}$  to  $Q$
- Approximation: update  $Q$  towards a target

$$\Delta Q(s_t, a_t) = \alpha [G_t - Q(s_t, a_t)]$$

- Prediction: fixed  $\pi$ , keep updating  $Q$
- Control: alternate value updates and policy updates

$$\pi_{k+1} = (1 - \epsilon)\pi_{\text{greedy}(Q_k)} + \epsilon\pi_{\text{any}}$$

**Increasingly greedy policy:**  $\epsilon \rightarrow 0$  as  $t \rightarrow \infty$

- **Forward view:** look one step forward to compute the target

$$\Delta Q(s_t, a_t) = \alpha \left[ r_{t+1} + \gamma \underset{a \sim \pi}{E} [Q(s_{t+1}, a)] - Q(s_t, a_t) \right]$$

- **Backward view:** wait one step to update  $Q(s_t, a_t)$

$$\Delta Q(s_{t-1}, a_{t-1}) = \alpha \left[ r_t + \gamma \underset{a \sim \pi}{E} [Q(s_t, a)] - Q(s_{t-1}, a_{t-1}) \right]$$

- **Forward view:** look one step forward to compute the target

$$\Delta Q(s_t, a_t) = \alpha \left[ r_{t+1} + \gamma \underset{a \sim \pi}{E} [Q(s_{t+1}, a)] - Q(s_t, a_t) \right]$$

- **Backward view:** wait one step to update  $Q(s_t, a_t)$

$$\Delta Q(s_{t-1}, a_{t-1}) = \alpha \left[ r_t + \gamma \underset{a \sim \pi}{E} [Q(s_t, a)] - Q(s_{t-1}, a_{t-1}) \right]$$



- Short answer: useful later
- Longer answer: Van Seijen et al. [2]



Look far (n steps) in the future:

$$\begin{aligned} G_t^{(n)} &\doteq r_{t+1} + \gamma r_{t+2} + \cdots + \gamma^{n-1} r_{t+n} + \gamma^n E_{a \sim \pi} [Q(s_{t+n}, a)] \\ &= \sum_{k=1}^n \gamma^{k-1} r_{t+k} + \gamma^n E_{a \sim \pi} [Q(s_{t+n}, a)] \end{aligned}$$

- $G_t^{(1)}$  is a TD target: high bias, low variance
- $G_t^{(T-t)}$  is a Monte Carlo target: no bias, high variance

Look far (n steps) in the future:

$$\begin{aligned} G_t^{(n)} &\doteq r_{t+1} + \gamma r_{t+2} + \cdots + \gamma^{n-1} r_{t+n} + \gamma^n \underset{a \sim \pi}{E} [Q(s_{t+n}, a)] \\ &= r_{t+1} + \gamma \underset{a \sim \pi}{E} [Q(s_{t+n}, a)] \end{aligned}$$

- $G_t^{(1)}$  is a TD target: high bias, low variance
- $G_t^{(T-t)}$  is a Monte Carlo target: no bias, high variance

Look far (n steps) in the future:

$$\begin{aligned} G_t^{(n)} &\doteq r_{t+1} + \gamma r_{t+2} + \cdots + \gamma^{n-1} r_{t+n} + \gamma^n E_{a \sim \pi} [Q(s_{t+n}, a)] \\ &= \sum_{k=1}^{T-t} \gamma^{k-1} r_{t+k} + \gamma^n E_{a \sim \pi} [Q(s_{t+n}, a)] \end{aligned}$$

- $G_t^{(1)}$  is a TD target: high bias, low variance
- $G_t^{(T-t)}$  is a Monte Carlo target: no bias, high variance

Look far (n steps) in the future:

$$G_t^{(n)} \doteq r_{t+1} + \gamma r_{t+2} + \cdots + \gamma^{n-1} r_{t+n} + \gamma^n \underset{a \sim \pi}{E} [Q(s_{t+n}, a)]$$
$$=$$

- $G_t^{(1)}$  is a TD target: high bias, low variance
- $G_t^{(T-t)}$  is a Monte Carlo target: no bias, high variance

**How can we use all future returns while keeping the variance low?**

1 Introduction

2 Eligibility Traces

3 Off-policy Credit Assignment

4 Retrace( $\lambda$ )

5 Experiments

Average **all** n-step targets:

$$G_t^\lambda \doteq (1 - \lambda) \sum_{n=1}^{\infty} \lambda^{n-1} G_t^{(n)}$$

- $\lambda = 0$  gives  $G_t^{(1)}$ , the TD target
- $\lambda = 1$  gives  $G_t^{(T-t)}$ , the Monte Carlo target



Average **all** n-step targets:

$$G_t^\lambda = (1 - \lambda) \sum_{n=1}^{T-t-1} \lambda^{n-1} G_t^{(n)} + \lambda^{T-t-1} G_t^{(T-t)}$$

- $\lambda = 0$  gives  $G_t^{(1)}$ , the TD target
- $\lambda = 1$  gives  $G_t^{(T-t)}$ , the Monte Carlo target

Average **all** n-step targets:

$$G_t^\lambda = (1 - 0)0^0 G_t^{(1)} + (1 - 0) \sum_{n=2}^{T-t-1} 0^{n-1} G_t^{(n)} + 0^{T-t-1} G_t^{(T-t)}$$

- $\lambda = 0$  gives  $G_t^{(1)}$ , the TD target
- $\lambda = 1$  gives  $G_t^{(T-t)}$ , the Monte Carlo target

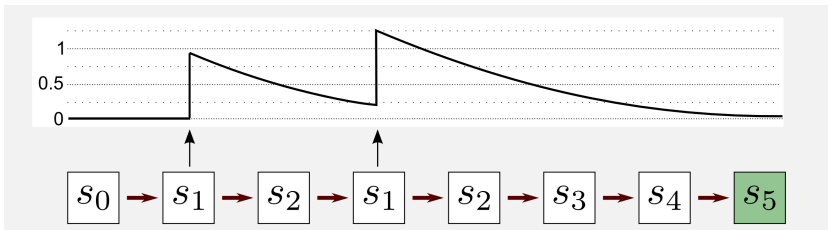
Average **all** n-step targets:

$$G_t^\lambda = (1 - \lambda) \sum_{n=1}^{T-t-1} \lambda^{n-1} G_t^{(n)} + \lambda^{T-t-1} G_t^{(T-t)}$$

- $\lambda = 0$  gives  $G_t^{(1)}$ , the TD target
- $\lambda = 1$  gives  $G_t^{(T-t)}$ , the Monte Carlo target

A 10x10 grid with a star at (5, 5) and an arrow pointing up at (5, 4).

- Assign a "trace" to each state-action pair
- Set the trace to one when visited
- All traces fade with time
- TD update proportional to trace



Initialization:

$$e(s, a) \leftarrow 0 \text{ for all } s, a$$

Update **all states** at each step  $t$ :

$$e(s, a) \leftarrow \mathbb{1}\{s = s_t, a = a_t\} + (1 - \mathbb{1}\{s = s_t, a = a_t\})\gamma\lambda e(s, a)$$

$$\Delta Q(s, a) = \alpha e(s, a)(G_t^{(1)} - Q(s, a))$$

Initialization:

$$e(s, a) \leftarrow 0 \text{ for all } s, a$$

Update **all states** at each step  $t$ :

$$e \leftarrow \mathbb{1}_t + (1 - \mathbb{1}_t)\gamma\lambda e(s, a)$$

$$\Delta Q = \alpha e \delta_t$$

$$\Delta Q(s_t, a_t) = \alpha_k \sum_{k \geq t} \left[ (G_k^{(1)} - Q(s_k, a_k)) \sum_{j=t}^k (\gamma \lambda)^{k-j} \mathbb{1} \{s_j, a_j = s_t, a_t\} \right]$$



- 1 Introduction
- 2 Eligibility Traces
- 3 Off-policy Credit Assignment
- 4 Retrace( $\lambda$ )
- 5 Experiments

Two policies:

- **Target** policy  $\pi$ : the one that is evaluated/improved
- **Behavioral** policy  $\mu$ : the one that is used to interact with the environment

Potentially, both can change!

Advantages:

- Separate exploration from evaluation
- Reuse past experience

Correct the update with likelihood ratios

$$\Delta Q(s_t, a_t) = \alpha \prod_{k \geq t} \frac{\pi(a_k | s_k)}{\mu(a_k | s_k)} \left[ G_t^{(n)} - Q(s_t, a_t) \right]$$

Correct the update with likelihood ratios

$$\Delta Q(s_t, a_t) = \alpha \prod_{k \geq t} \frac{\pi(a_k | s_k)}{\mu(a_k | s_k)} \left[ G_t^{(n)} - Q(s_t, a_t) \right]$$

**Issue:** high variance!

Approximate  $\mathcal{T}^*$  directly

$$\Delta Q(s_t, a_t) = \alpha \left[ r_{t+1} + \gamma \max_{a \in \mathcal{A}} Q(s_{t+1}, a) - Q(s_t, a_t) \right]$$

Approximate  $\mathcal{T}^*$  directly

$$\Delta Q(s_t, a_t) = \alpha \left[ r_{t+1} + \gamma \max_{a \in \mathcal{A}} Q(s_{t+1}, a) - Q(s_t, a_t) \right]$$

Implicit target policy:  $\pi_{\text{greedy}}(Q)$

Approximate  $\mathcal{T}^*$  directly

$$\Delta Q(s_t, a_t) = \alpha [G_t^* - Q(s_t, a_t)]$$

Implicit target policy:  $\pi_{\text{greedy}(Q)}$

Maximization as a special case of expectation

$\implies G_t^* = G_t^{(1)}$  when  $\pi$  is a greedy policy

$\implies$  Q-learning is a special case of Expected Sarsa!





**Cut** the eligibility trace each time  $\mu$  performs a non-greedy action

For all  $s, a$

$$e \leftarrow \gamma \lambda e + \mathbb{1}_t$$

$$\Delta Q(s, a) = \alpha e(s, a) [G_t^* - Q(s, a)]$$

$$\text{If } a_t \neq \arg \max_{a' \in \mathcal{A}} Q(s_t, a')$$

$$e(s, a) \leftarrow 0$$

Credit is assigned **up to the last greedy action**

**Issues:**

- Traces are cut too often
- Convergence was an open problem since 1989!

Back to the forward view

$$\Delta Q(s_t, a_t) = \alpha_k \sum_{k \geq t} \left[ (G_k^{(1)} - Q(s_k, a_k)) \sum_{j=t}^k (\gamma \lambda)^{k-j} \mathbb{1}\{s_j, a_j = s_t, a_t\} \right]$$

Back to the forward view

$$\Delta Q(s_t, a_t) = \alpha_k \sum_{k \geq t} \left[ (G_k^{(1)} - Q(s_k, a_k)) \sum_{j=t}^k \gamma^{k-j} \lambda^{k-j} \mathbb{1}\{s_j, a_j = s_t, a_t\} \right]$$

Back to the forward view

$$\Delta Q(s_t, a_t) = \alpha_k \sum_{k \geq t} \left[ (G_k^{(1)} - Q(s_k, a_k)) \sum_{j=t}^k \gamma^{k-j} \left( \prod_{i=j+1}^k c_i \right) \mathbb{1} \{s_j, a_j = s_t, a_t\} \right]$$

Back to the forward view

$$\Delta Q(s_t, a_t) = \alpha_k \sum_{k \geq t} \left[ (G_k^{(1)} - Q(s_k, a_k)) \sum_{j=t}^k \gamma^{k-j} \left( \prod_{i=j+1}^k \mathbf{c}_i \right) \mathbb{1}_{jt} \right]$$

In this formulation we call the  $\mathbf{c}_i$  "traces".

$$\Delta Q(s_t, a_t) = \alpha_k \sum_{k \geq t} \left[ (G_k^{(1)} - Q(s_k, a_k)) \sum_{j=t}^k \gamma^{k-j} \left( \prod_{i=j+1}^k \mathbf{c}_i \right) \mathbb{1}_{jt} \right]$$

- $c_i = \frac{\pi(a_i | s_i)}{\mu(a_i | s_i)}$
- $c_i = \lambda$  [citation here]
- $c_i = \lambda \pi(a_i | s_i)$  [citation here]

$$\Delta Q(s_t, a_t) = \alpha_k \sum_{k \geq t} \left[ (G_k^{(1)} - Q(s_k, a_k)) \sum_{j=t}^k \gamma^{k-j} \left( \prod_{i=j+1}^k \mathbf{c}_i \right) \mathbb{1}_{jt} \right]$$

- $c_i = \frac{\pi(a_i | s_i)}{\mu(a_i | s_i)} \implies$  Importance Sampling!
- $c_i = \lambda$  [citation here]
- $c_i = \lambda \pi(a_i | s_i)$  [citation here]

$$\Delta Q(s_t, a_t) = \alpha_k \sum_{k \geq t} \left[ (G_k^{(1)} - Q(s_k, a_k)) \sum_{j=t}^k \gamma^{k-j} \left( \prod_{i=j+1}^k \mathbf{c}_i \right) \mathbb{1}_{jt} \right]$$

- $c_i = \frac{\pi(a_i | s_i)}{\mu(a_i | s_i)} \implies$  Importance Sampling!
- $c_i = \lambda \implies Q^\pi(\lambda)$  [citation here]
- $c_i = \lambda \pi(a_i | s_i)$  [citation here]



$$\Delta Q(s_t, a_t) = \alpha_k \sum_{k \geq t} \left[ (G_k^{(1)} - Q(s_k, a_k)) \sum_{j=t}^k \gamma^{k-j} \left( \prod_{i=j+1}^k \mathbf{c}_i \right) \mathbb{1}_{jt} \right]$$

- $c_i = \frac{\pi(a_i | s_i)}{\mu(a_i | s_i)} \implies$  Importance Sampling!
- $c_i = \lambda \implies Q^\pi(\lambda)$  [citation here]
- $c_i = \lambda \pi(a_i | s_i) \implies TB(\lambda)$  [citation here]

- Traces:  $c_i = \lambda$

- Idea:

- Low variance

- **Issue:** not safe

Convergence only if  $\pi \simeq \mu$ :

$$\|\pi - \mu\|_1 \leq \frac{1 - \gamma}{\lambda \gamma}$$

- Traces:  $c_i = \lambda \pi(a_i \mid s_i)$
- Idea:
- Convergence for any  $\pi, \mu$
- **Issue:** not efficient

Traces are cut unnecessarily when almost on-policy

$$\Delta Q(s_t, a_t) = \alpha_k \sum_{k \geq t} \left[ (G_k^{(1)} - Q(s_k, a_k)) \sum_{j=t}^k \gamma^{k-j} \left( \prod_{i=j+1}^k \mathbf{c}_i \right) \mathbb{1}_{jt} \right]$$

Algorithm	Trace $c_i$	Issue
IS	$\frac{\pi(a_i   s_i)}{\mu(a_i   s_i)}$	High variance
$Q^\pi(\lambda)$	$\lambda$	Not safe off-policy
$TB(\lambda)$	$\lambda \pi(a_i   s_i)$	Not efficient on-policy

We want an algorithm that has **low variance**, is **safe** and **efficient**

### Theorem (Off-policy prediction)

*For any  $\pi$  and  $\mu$ , assuming the state space is finite and all states are visited infinitely often:*

$$\text{If } 0 \leq c_i \leq \frac{\pi(a_i | s_i)}{\mu(a_i | s_i)} \quad \text{then} \quad Q \rightarrow Q^\pi$$

### Theorem (Off-policy prediction)

*For any  $\pi$  and  $\mu$ , assuming the state space is finite and all states are visited infinitely often:*

$$\text{If } 0 \leq c_i \leq \frac{\pi(a_i | s_i)}{\mu(a_i | s_i)} \quad \text{then} \quad Q \rightarrow Q^\pi$$

*Proof sketch:* Define the off-policy operator  $\mathcal{R}$ :

$$\mathcal{R}Q(s, a) \doteq Q(s, a) + \mathbb{E}_{a_t \sim \mu} \left[ \sum_{t \geq 0} \gamma^t \left( \prod_{i=1}^t c_i \right) (G_t^{(1)} - Q(s_t, a_t)) \right]$$

Show that  $Q^\pi$  is the unique fixed point of  $\mathcal{R}$

Show that  $\|\mathcal{R}Q - Q^\pi\|_\infty \leq \gamma \|Q - Q^\pi\|_\infty$

### Theorem (Off-policy prediction)

*For any  $\pi$  and  $\mu$ , assuming the state space is finite and all states are visited infinitely often:*

$$\text{If } 0 \leq c_i \leq \frac{\pi(a_i | s_i)}{\mu(a_i | s_i)} \quad \text{then} \quad Q \rightarrow Q^\pi$$

*Proof sketch:* Define the off-policy operator  $\mathcal{R}$ :

$$\mathcal{R}Q(s, a) \doteq Q(s, a) + E_{a_t \sim \mu} \left[ \sum_{t \geq 0} \gamma^t \left( \prod_{i=1}^t c_i \right) (G_t^{(1)} - Q(s_t, a_t)) \right]$$

Show that  $Q^\pi$  is the unique fixed point of  $\mathcal{R}$

Show that  $\|\mathcal{R}Q - Q^\pi\|_\infty \leq \gamma \|Q - Q^\pi\|_\infty$

### Theorem (Off-policy prediction)

*For any  $\pi$  and  $\mu$ , assuming the state space is finite and all states are visited infinitely often:*

$$\text{If } 0 \leq c_i \leq \frac{\pi(a_i | s_i)}{\mu(a_i | s_i)} \quad \text{then} \quad Q \rightarrow Q^\pi$$

*Proof sketch:* Define the off-policy operator  $\mathcal{R}$ :

$$\mathcal{R}Q(s, a) \doteq Q(s, a) + E_{a_t \sim \mu} \left[ \sum_{t \geq 0} \gamma^t \left( \prod_{i=1}^t c_i \right) (G_t^{(1)} - Q(s_t, a_t)) \right]$$

Show that  $Q^\pi$  is the unique fixed point of  $\mathcal{R}$

Show that  $\|\mathcal{R}Q - Q^\pi\|_\infty \leq \gamma \|Q - Q^\pi\|_\infty$



## Theorem (Off-policy prediction)

*For any  $\pi$  and  $\mu$ , assuming the state space is finite and all states are visited infinitely often:*

$$\text{If } 0 \leq c_i \leq \frac{\pi(a_i | s_i)}{\mu(a_i | s_i)} \quad \text{then} \quad Q \rightarrow Q^\pi$$

*Proof sketch:* Define the off-policy operator  $\mathcal{R}$ :

$$\mathcal{R}Q(s, a) \doteq Q(s, a) + E_{a_t \sim \mu} \left[ \sum_{t \geq 0} \gamma^t \left( \prod_{i=1}^t c_i \right) (G_t^{(1)} - Q(s_t, a_t)) \right]$$

Show that  $Q^\pi$  is the unique fixed point of  $\mathcal{R}$

Show that  $\|\mathcal{R}Q - Q^\pi\|_\infty \leq \gamma \|Q - Q^\pi\|_\infty$

### Theorem (Off-policy prediction)

*For any  $\pi$  and  $\mu$ , assuming the state space is finite and all states are visited infinitely often:*

$$\text{If } 0 \leq c_i \leq \frac{\pi(a_i | s_i)}{\mu(a_i | s_i)} \quad \text{then} \quad Q \rightarrow Q^\pi$$

- **Safety:** ensured
- **Variance:** maximal when  $c_i = \frac{\pi(a_i | s_i)}{\mu(a_i | s_i)}$
- **Efficiency:** minimal when  $c_i = 0$

- 1 Introduction
- 2 Eligibility Traces
- 3 Off-policy Credit Assignment
- 4 Retrace( $\lambda$ )**
- 5 Experiments

$$a_t \sim \mu(\cdot \mid s_t)$$

$$\Delta Q(s_t, a_t) = \alpha_k \sum_{k \geq t} \left[ (G_k^{(1)} - Q(s_k, a_k)) \sum_{j=t}^k \gamma^{k-j} \left( \prod_{i=j+1}^k \mathbf{c}_i \right) \mathbb{1}_{jt} \right]$$

$$c_i = \lambda \min \left\{ 1, \frac{\pi(a_i \mid s_i)}{\mu(a_i \mid s_i)} \right\}$$

$$a_t \sim \mu(\cdot \mid s_t)$$

$$\Delta Q(s_t, a_t) = \alpha_k \sum_{k \geq t} \left[ (G_k^{(1)} - Q(s_k, a_k)) \sum_{j=t}^k \gamma^{k-j} \left( \prod_{i=j+1}^k c_i \right) \mathbb{1}_{jt} \right]$$

$$c_i = \lambda \min \left\{ 1, \frac{\pi(a_i \mid s_i)}{\mu(a_i \mid s_i)} \right\}$$

- **Safe** since  $0 \leq c_i \leq \frac{\pi(a_i \mid s_i)}{\mu(a_i \mid s_i)}$
- **Low variance** since  $c_i \leq 1$
- **Efficient on-policy** since  $c_i \rightarrow 1$  as  $\mu \rightarrow \pi$

$$a_t \sim \mu(\cdot \mid s_t)$$

$$\Delta Q(s_t, a_t) = \alpha_k \sum_{k \geq t} \left[ (G_k^{(1)} - Q(s_k, a_k)) \sum_{j=t}^k \gamma^{k-j} \left( \prod_{i=j+1}^k c_i \right) \mathbb{1}_{jt} \right]$$

$$c_i = \lambda \min \left\{ 1, \frac{\pi(a_i \mid s_i)}{\mu(a_i \mid s_i)} \right\}$$

- **Safe** since  $0 \leq c_i \leq \frac{\pi(a_i \mid s_i)}{\mu(a_i \mid s_i)}$
- **Low variance** since  $c_i \leq 1$
- **Efficient on-policy** since  $c_i \rightarrow 1$  as  $\mu \rightarrow \pi$

$$a_t \sim \mu(\cdot \mid s_t)$$

$$\Delta Q(s_t, a_t) = \alpha_k \sum_{k \geq t} \left[ (G_k^{(1)} - Q(s_k, a_k)) \sum_{j=t}^k \gamma^{k-j} \left( \prod_{i=j+1}^k c_i \right) \mathbb{1}_{jt} \right]$$

$$c_i = \lambda \min \left\{ 1, \frac{\pi(a_i \mid s_i)}{\mu(a_i \mid s_i)} \right\}$$

- **Safe** since  $0 \leq c_i \leq \frac{\pi(a_i \mid s_i)}{\mu(a_i \mid s_i)}$
- **Low variance** since  $c_i \leq 1$
- **Efficient on-policy** since  $c_i \rightarrow 1$  as  $\mu \rightarrow \pi$

## Theorem (Off-policy control)

For any  $\mu_k$ , if  $\pi_k$  is **increasingly greedy** w.r.t to  $Q_k$ :

$$\text{If } 0 \leq c_i \leq \frac{\pi_k(a_i | s_i)}{\mu_k(a_i | s_i)} \quad \text{then} \quad Q_k \rightarrow Q^*$$



## Theorem (Off-policy control)

For any  $\mu_k$ , if  $\pi_k$  is **increasingly greedy** w.r.t to  $Q_k$ :

$$\text{If } 0 \leq c_i \leq \frac{\pi_k(a_i | s_i)}{\mu_k(a_i | s_i)} \quad \text{then} \quad Q_k \rightarrow Q^*$$

*Proof sketch:* Show that

$$\|\mathcal{R}Q_k - Q^*\|_\infty \leq \gamma \|Q_k - Q^*\|_\infty + \epsilon_k \|Q_k\|_\infty$$

Then  $Q_k \rightarrow Q^*$  as  $\epsilon_k \rightarrow 0$

## Theorem (Off-policy control)

For any  $\mu_k$ , if  $\pi_k$  is **increasingly greedy** w.r.t to  $Q_k$ :

$$\text{If } 0 \leq c_i \leq \frac{\pi_k(a_i | s_i)}{\mu_k(a_i | s_i)} \quad \text{then} \quad Q_k \rightarrow Q^*$$

*Proof sketch:* Show that

$$\|\mathcal{R}Q_k - Q^*\|_\infty \leq \gamma \|Q_k - Q^*\|_\infty + \epsilon_k \|Q_k\|_\infty$$

Then  $Q_k \rightarrow Q^*$  as  $\epsilon_k \rightarrow 0$

## Theorem (Off-policy control)

For any  $\mu_k$ , if  $\pi_k$  is **increasingly greedy** w.r.t to  $Q_k$ :

$$\text{If } 0 \leq c_i \leq \frac{\pi_k(a_i | s_i)}{\mu_k(a_i | s_i)} \quad \text{then} \quad Q_k \rightarrow Q^*$$

*Proof sketch:* Show that

$$\|\mathcal{R}Q_k - Q^*\|_\infty \leq \gamma \|Q_k - Q^*\|_\infty + \epsilon_k \|Q_k\|_\infty$$

Then  $Q_k \rightarrow Q^*$  as  $\epsilon_k \rightarrow 0$

## Theorem (Off-policy control)

For any  $\mu_k$ , if  $\pi_k$  is **increasingly greedy** w.r.t to  $Q_k$ :

$$\text{If } 0 \leq c_i \leq \frac{\pi_k(a_i | s_i)}{\mu_k(a_i | s_i)} \quad \text{then} \quad Q_k \rightarrow Q^*$$

*Proof sketch:* Show that

$$\|\mathcal{R}Q_k - Q^*\|_\infty \leq \gamma \|Q_k - Q^*\|_\infty + \epsilon_k \|Q_k\|_\infty$$

Then  $Q_k \rightarrow Q^*$  as  $\epsilon_k \rightarrow 0$

## Theorem (Off-policy control)

For any  $\mu_k$ , if  $\pi_k$  is **increasingly greedy** w.r.t to  $Q_k$ :

$$\text{If } 0 \leq c_i \leq \frac{\pi_k(a_i | s_i)}{\mu_k(a_i | s_i)} \quad \text{then } Q_k \rightarrow Q^*$$

## Remarks

- No GLIE assumption on  $\mu$
- Extends to continuous action spaces
- $c_i$  must be Markovian

$$\alpha_k \sum_{k \geq t} \left[ (G_k^{(1)} - Q(s_k, a_k)) \sum_{j=t}^k \gamma^{k-j} \left( \prod_{i=j+1}^k \lambda \min \left\{ 1, \frac{\pi(a_i | s_i)}{\mu(a_i | s_i)} \right\} \right) \mathbb{1}_{jt} \right]$$

$$\alpha_k \sum_{k \geq t} \left[ (G_k^{(1)} - Q(s_k, a_k)) \sum_{j=t}^k \gamma^{k-j} \left( \prod_{i=j+1}^k \lambda \min \left\{ 1, \frac{\pi(a_i | s_i)}{\mu(a_i | s_i)} \right\} \right) \mathbb{1}_{jt} \right]$$

- when  $\pi$  is a greedy policy,  $G_t^{(1)}$  is just  $G_t^*$
- when  $\pi$  is a greedy policy,  $c_i = \lambda \mathbb{1} \{ \mu_k(s_i) = \pi_{\text{greedy}}(s_i) \}$

$$\alpha_k \sum_{k \geq t} \left[ (G_k^{(1)} - Q(s_k, a_k)) \sum_{j=t}^k \gamma^{k-j} \left( \prod_{i=j+1}^k \lambda \min \left\{ 1, \frac{\pi(a_i | s_i)}{\mu(a_i | s_i)} \right\} \right) \mathbb{1}_{jt} \right]$$

- when  $\pi$  is a greedy policy,  $G_t^{(1)}$  is just  $G_t^*$
- when  $\pi$  is a greedy policy,  $c_i = \lambda \mathbb{1} \{ \mu_k(s_i) = \pi_{\text{greedy}}(s_i) \}$



$$\alpha_k \sum_{k \geq t} \left[ (G_k^{(1)} - Q(s_k, a_k)) \sum_{j=t}^k \gamma^{k-j} \left( \prod_{i=j+1}^k \lambda \min \left\{ 1, \frac{\pi(a_i | s_i)}{\mu(a_i | s_i)} \right\} \right) \mathbb{1}_{jt} \right]$$

- when  $\pi$  is a greedy policy,  $G_t^{(1)}$  is just  $G_t^*$
- when  $\pi$  is a greedy policy,  $c_i = \lambda \mathbb{1} \{ \mu_k(s_i) = \pi_{\text{greedy}}(s_i) \}$

Watkins'  $Q(\lambda)$  is a special case of  $\text{Retrace}(\lambda)$

Convergence of Watkins'  $Q(\lambda)$  proved after 27 years!

Retrace( $\lambda$ ) is more general than Watkins'  $Q(\lambda)$

- $\pi_k$  and  $\mu_k$  can both be increasingly greedy policies, with  $\pi_k$  converging faster than  $\mu_k \implies$  A3C [citation here]
- $\mu$  can be a snapshot of  $\pi \implies$  DQN with memory replay [citation here]

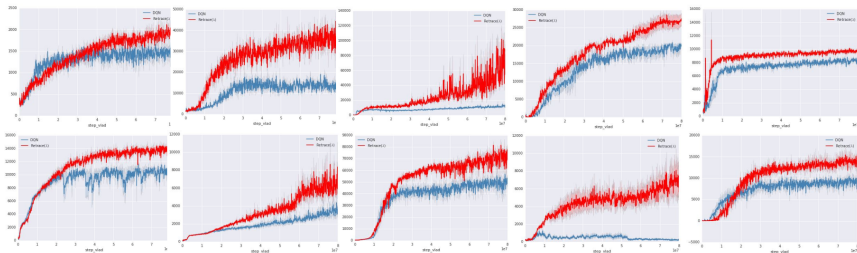
Retrace( $\lambda$ ) is more general than Watkins'  $Q(\lambda)$

- $\pi_k$  and  $\mu_k$  can both be increasingly greedy policies, with  $\pi_k$  converging faster than  $\mu_k \implies$  A3C [citation here]
- $\mu$  can be a snapshot of  $\pi \implies$  DQN with memory replay [citation here]

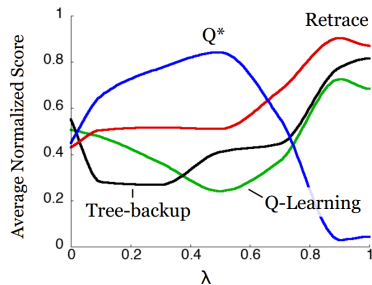
Retrace( $\lambda$ ) is more general than Watkins'  $Q(\lambda)$

- $\pi_k$  and  $\mu_k$  can both be increasingly greedy policies, with  $\pi_k$  converging faster than  $\mu_k \implies$  A3C [citation here]
- $\mu$  can be a snapshot of  $\pi \implies$  DQN with memory replay [citation here]

- 1 Introduction
- 2 Eligibility Traces
- 3 Off-policy Credit Assignment
- 4 Retrace( $\lambda$ )
- 5 Experiments



Asteroids, Defender, Demon Attack, Hero, Krull, River Raid, Space Invaders, Star Gunner, Wizard of Wor, Zaxxon



Algorithm	Times Best
DQN	12
$Q^*(\lambda)$	2
TB( $\lambda$ )	16
Retrace( $\lambda$ )	<b>30</b>

Table :  $\lambda = 1$



Harm Seijen and Rich Sutton.

True online td ( $\lambda$ ).

In *International Conference on Machine Learning*, pages 692–700, 2014.



Harm Van Seijen, Hado Van Hasselt, Shimon Whiteson, and Marco Wiering.

A theoretical and empirical analysis of expected sarsa.

In *Adaptive Dynamic Programming and Reinforcement Learning, 2009. ADPRL'09. IEEE Symposium on*, pages 177–184. IEEE, 2009.



Update **all states** at each step  $t$ :

$$e(s, a) \leftarrow \gamma \lambda e(s, a) + \mathbb{1}\{s = s_t, a = a_t\}$$

$$\Delta Q(s, a) = \alpha e(s, a) (G_t^{(1)} - Q(s, a))$$

Update **all states** at each step  $t$ :

$$e \leftarrow \gamma \lambda e + \mathbb{1}_t$$

$$\Delta Q = \alpha e \delta_t$$

Update at step  $t$ :

$$e \leftarrow (1 - \alpha \mathbb{1}_t) \gamma \lambda e + \mathbb{1}_t$$
$$\Delta Q = \alpha e \delta_t$$

Seijen and Sutton, 2014 [1]

