Safe and Efficient Off-Policy Reinforcement Learning

NIPS 2016

Yasuhiro Fujita

Preferred Networks Inc.

January 11, 2017

[Munos et al. 2016]

Safe and efficient off-policy reinforcement learning

Rémi Munos
munos@google.com
Google DeepMind

Tom Stepleton
stepleton@google.com
Google DeepMind

Anna Harutyunyan
anna.harutyunyan@vub.ac.be
Vrije Universiteit Brussel

Marc G. Bellemare
bellemare@google.com
Google DeepMind

- Proposes a new off-policy multi-step RL method: Retrace(λ)
  - Good theoretical properties: low-variance, safe and efficient
  - It outperforms one-step Q-learning and existing multi-step variants
- Proves the convergence of Watkins's Q(λ) for the first time

Multi-step methods

- Multi-step methods have some advantages over single-step methods
  - They can balance bias and variance
  - They can propagate values more quickly
- Example: SARSA(λ)
  - n-step return

$$\mathcal{R}_s^{(n)} = \sum_{t=s}^{s+n} \gamma^{t-s} r_t + \gamma^{n+1} Q(x_{s+n+1}, a_{s+n+1})$$

  - λ-return based update rule

$$\Delta Q(x_s, a_s) = \sum_{t \geq s} (\lambda\gamma)^{t-s} \delta_t$$

$$\delta_t = r_t + \gamma Q(x_{t+1}, a_{t+1}) - Q(x_t, a_t)$$

Multi-step methods in off-policy settings

- Can we apply multi-step methods to off-policy cases?
  - Policy evaluation: estimate Q^π from samples collected by μ ($\pi \neq \mu$)
  - Control: estimate Q^* from samples collected by μ
- In "Algorithms for Reinforcement Learning", p. 57

There exist multi-step versions of Q-learning (e.g., Sutton and Barto, 1998, Section 7.6). However, these are not as appealing (and straightforward) as the multi-step extensions of TD(0) since Q-learning is an inherently off-policy algorithm: the temporal differences underlying Q-learning do not telescope even when $X_{t+1} = Y_{t+1}$.

Watkins's Q(λ) [Watkins 1989]

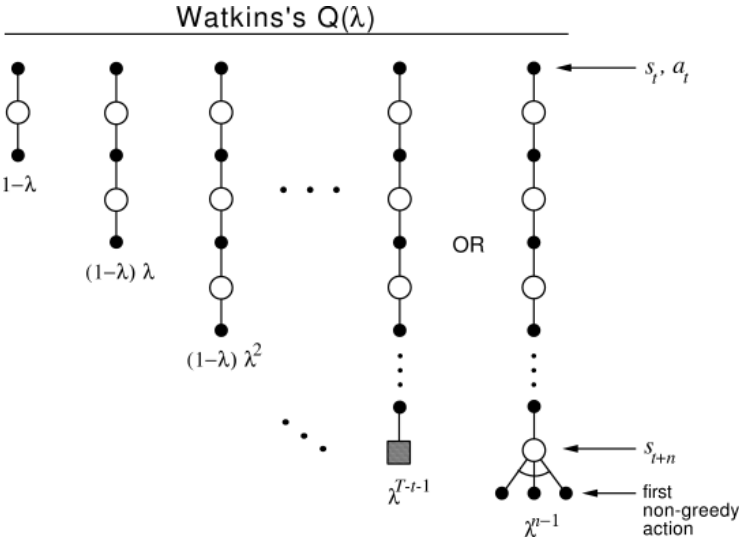
- Classic multi-step algorithm for off-policy control
- Cut off traces whenever a non-greedy action is taken

$$\mathcal{R}_s^{(n)} = \sum_{t=s}^{s+n} \gamma^{t-s} r_t + \gamma^{n+1} \max_a Q(x_{s+n+1}, a)$$

$$\text{(for any } n < \tau = \arg\min_{u \geq 1} \mathbb{I}\{\pi_{s+u} \neq \mu_{s+u}\})$$

- Converges to Q^* under a mild assumption (proved in this paper)
- Only little faster than one-step Q-learning if non-greedy actions are frequent (i.e. not "efficient")

Backup diagram of Q(λ)



General operator $\mathcal{R}$

- To compare off-policy multistep methods, consider the general operator $\mathcal{R}$:

$$\mathcal{R}Q(x,a) := Q(x,a) + \mathbb{E}_\mu\Big[\sum_{t\geq 0}\gamma^t\Big(\prod_{s=1}^{t} c_s\Big)\big(r_t + \gamma\mathbb{E}_\pi Q(x_{t+1},\cdot) - Q(x_t,a_t)\big)\Big], \qquad (3)$$

- Different non-negative coefficients c_s (traces) result in different methods
- (Is this equation correct?)

Desired properties

- Low variance
  - Variance of the online estimate is small
  - $\approx \mathbb{V}(c_1 \cdots c_t)$ is small
  - $\approx \mathbb{V}(c)$ is small
- Safe
  - Convergence to Q^π (policy evaluation) or Q^* (control) is guaranteed
- Efficient
  - Traces are not unnecessarily cut if π and μ are close

Comparison of properties

	Definition of c_s	Estimation variance	Guaranteed convergence†	Use full returns (near on-policy)
Importance sampling	$\frac{\pi(a_s\|x_s)}{\mu(a_s\|x_s)}$	High	for any π, μ	yes
$Q^\pi(\lambda)$	λ	Low	for π close to μ	yes
TB(λ)	$\lambda\pi(a_s\|x_s)$	Low	for any π, μ	no
Retrace(λ)	$\lambda \min\left(1, \frac{\pi(a_s\|x_s)}{\mu(a_s\|x_s)}\right)$	Low	for any π, μ	yes

Table 1: Properties of several algorithms defined in terms of the general operator given in (3). †Guaranteed convergence of the expected operator $\mathcal{R}$.

- Retrace(λ) is low-variance, safe and efficient
- Note that Watkins's Q(λ) $\neq$ Q$^\pi$(λ)

Importance Sampling (IS) [Precup et al. 2000]

$$\mathcal{R}Q(x,a) := Q(x,a) + \mathbb{E}_\mu \Big[\sum_{t \geq 0} \gamma^t \Big(\prod_{s=1}^{t} c_s \Big) \big(r_t + \gamma \mathbb{E}_\pi Q(x_{t+1}, \cdot) - Q(x_t, a_t) \big) \Big], \qquad (3)$$

$$c_s = \frac{\pi(a_s|x_s)}{\mu(a_s|x_s)}$$

- $\mathcal{R}Q = Q^\pi$ for any Q in this case
  - If $Q = 0$, it just becomes the basic IS estimate $\sum_{t \geq 0} \gamma^t (\prod_{s=1}^{t} c_s) r_t$
- High variance, mainly due to the variance of the product

$$\frac{\pi(a_1|x_1)}{\mu(a_1|x_1)} \cdots \frac{\pi(a_t|x_t)}{\mu(a_t|x_t)}$$

Off-policy $Q^\pi(\lambda)$ and $Q^*(\lambda)$ [Harutyunyan et al. 2016]

$$\mathcal{R}Q(x,a) := Q(x,a) + \mathbb{E}_\mu\Big[\sum_{t\geq 0}\gamma^t\Big(\prod_{s=1}^{t} c_s\Big)\big(r_t + \gamma\mathbb{E}_\pi Q(x_{t+1},\cdot) - Q(x_t,a_t)\big)\Big], \qquad (3)$$

$$c_s = \lambda$$

- A very recently proposed alternative
  - $Q^\pi(\lambda)$ for policy evaluation, Q^* for control
- To guarantee convergence, π and μ must be sufficiently close:
  - In policy evaluation, $\lambda < \frac{1-\gamma}{\gamma\epsilon}$, where $\epsilon := max_x \|\pi(\cdot|x) - \mu(\cdot|x)\|_1$
  - In control, $\lambda < \frac{1-\gamma}{2\gamma}$
- Available even if μ is unknown and/or non-Markovian

Tree Backup (TB) [Precup et al. 2000]

$$\mathcal{R}Q(x,a) := Q(x,a) + \mathbb{E}_\mu\Big[\sum_{t\geq 0}\gamma^t\Big(\prod_{s=1}^{t} c_s\Big)\big(r_t + \gamma\mathbb{E}_\pi Q(x_{t+1},\cdot) - Q(x_t,a_t)\big)\Big], \qquad (3)$$

$$c_s = \lambda\pi(a_s|x_s)$$

- The operator defines a contraction, thus is safe
- Not efficient because it cuts traces even if $\pi = \mu$
- Available even if μ is unknown and/or non-Markovian

Useful table [Harutyunyan et al. 2016]

Algorithm	***n*-step return**	**Update rule for the λ-return**	**FP**
TD(λ) (on-policy)	$\sum_{t=s}^{s+n} \gamma^{t-s} r_t + \gamma^{n+1} V(x_{s+n+1})$	$\sum_{t \geq s} (\lambda\gamma)^{t-s} \delta_t$ $\delta_t = r_t + \gamma V(x_{t+1}) - V(x_t)$	V^μ
SARSA(λ) (on-policy)	$\sum_{t=s}^{s+n} \gamma^{t-s} r_t + \gamma^{n+1} Q(x_{s+n+1}, a_{s+n+1})$	$\sum_{t \geq s} (\lambda\gamma)^{t-s} \delta_t$ $\delta_t = r_t + \gamma Q(x_{t+1}, a_{t+1}) - Q(x_t, a_t)$	Q^μ
$\mathbb{E}$ SARSA(λ) (on-policy)	$\sum_{t=s}^{s+n} \gamma^{t-s} r_t + \gamma^{n+1} \mathbb{E}_\mu Q(x_{s+n+1}, \cdot)$	$\sum_{t \geq s} (\lambda\gamma)^{t-s} \delta_t + \mathbb{E}_\mu Q(x_s, \cdot) - Q(x_s, a_s)$ $\delta_t = r_t + \gamma \mathbb{E}_\mu Q(x_{t+1}, \cdot) - \mathbb{E}_\mu Q(x_t, \cdot)$	Q^μ
General Q(λ) (off-policy)	$\sum_{t=s}^{s+n} \gamma^{t-s} r_t + \gamma^{n+1} \mathbb{E}_\pi Q(x_{s+n+1}, \cdot)$	$\sum_{t \geq s} (\lambda\gamma)^{t-s} \delta_t + \mathbb{E}_\pi Q(x_s, \cdot) - Q(x_s, a_s)$ $\delta_t = r_t + \gamma \mathbb{E}_\pi Q(x_{t+1}, \cdot) - \mathbb{E}_\pi Q(x_t, \cdot)$	$Q^{\mu,\pi}$
PDIS(λ) (off-policy)	$\sum_{t=s}^{s+n} \gamma^{t-s} r_t \prod_{i=s+1}^{t} \rho_i$ $+\gamma^{n+1} Q(x_{s+n+1}, a_{s+n+1}) \prod_{i=s}^{s+n} \rho_i$	$\sum_{t \geq s} (\lambda\gamma)^{t-s} \delta_t \prod_{i=s+1}^{t} \rho_i$ $\delta_t = r_t + \gamma \rho_{t+1} Q(x_{t+1}, a_{t+1}) - Q(x_t, a_t)$	Q^π
TB(λ) (off-policy)	$\sum_{t=s}^{s+n} \gamma^{t-s} \prod_{i=s+1}^{t} \pi_i \left[r_t + \gamma \mathbb{E}_\pi^{a \neq a_{t+1}} Q(x_{t+1}, \cdot)\right]$ $+\gamma^{n+1} \prod_{i=s+1}^{s+n+1} \pi_i Q(x_{s+n+1}, a_{s+n+1})$	$\sum_{t \geq s} (\lambda\gamma)^{t-s} \delta_t \prod_{i=s+1}^{t} \pi_i$ $\delta_t = r_t + \gamma \mathbb{E}_\pi Q(x_{t+1}, \cdot) - Q(x_t, a_t)$	Q^π
$\mathbf{Q}^\pi(\lambda)$ (on/off-policy)	$\sum_{t=s}^{s+n} \gamma^{t-s} \left[r_t + \mathbb{E}_\pi Q(x_t, \cdot) - Q(x_t, a_t)\right]$ $+\gamma^{n+1} \mathbb{E}_\pi Q(x_{s+n+1}, \cdot)$	$\sum_{t \geq s} (\lambda\gamma)^{t-s} \delta_t$ $\delta_t = r_t + \gamma \mathbb{E}_\pi Q(x_{t+1}, \cdot) - Q(x_t, a_t)$	Q^π
Q(λ) (Watkins's)	$\sum_{t=s}^{s+n} \gamma^{t-s} r_t + \gamma^{n+1} \max_a Q(x_{s+n+1}, a)$ (for any $n < \tau = \arg\min_{u \geq 1} \mathbb{I}\{\pi_{s+u} \neq \mu_{s+u}\}$)	$\sum_{t=s}^{s+\tau} (\lambda\gamma)^{t-s} \delta_t \prod_{i=s+1}^{t}$ $\delta_t = r_t + \gamma \max_a Q(x_{t+1}, a) - Q(x_t, a_t)$	Q^*
Q(λ) (P & W's)	$\sum_{t=s}^{s+n} \gamma^{t-s} r_t + \gamma^{n+1} \max_a Q(x_{s+n+1}, a)$	$\sum_{t=s}^{s+n} (\lambda\gamma)^{t-s} \delta_t + \max_a Q(x_s, a) - Q(x_s, a_s)$ $\delta_t = r_t + \gamma \max_a Q(x_{t+1}, a) - \max_a Q(x_t, a)$	$Q^{\mu,*}$
$\mathbf{Q}^*(\lambda)$	$\sum_{t=s}^{s+n} \gamma^{t-s} \left[r_t + \max_a Q(x_t, a) - Q(x_t, a_t)\right]$ $+\gamma^{n+1} \max_a Q(x_{s+n+1}, a)$	$\sum_{t \geq s} (\lambda\gamma)^{t-s} \delta_t$ $\delta_t = r_t + \gamma \max_a Q(x_{t+1}, a) - Q(x_t, a_t)$	Q^*

Retrace(λ)

$$\mathcal{R}Q(x,a) := Q(x,a) + \mathbb{E}_\mu\Big[\sum_{t\geq 0}\gamma^t\Big(\prod_{s=1}^{t} c_s\Big)\big(r_t + \gamma\mathbb{E}_\pi Q(x_{t+1},\cdot) - Q(x_t,a_t)\big)\Big], \qquad (3)$$

$$c_s = \lambda \min\left(1, \frac{\pi(a_s|x_s)}{\mu(a_s|x_s)}\right)$$

- Proposed by this paper
- IS ratio truncated at 1
- If π is close to μ, c_s is close to 1, avoid unnecessarily cutting traces

Experiments on Atari 2600

- Trained asynchrounously with 16 threads
- Each thread has private replay memory holding 62,500 transitions
- Q-learning uses a minibatch of 64 transitions
- Retrace, TB and Q* use four 16-step sequences

Performance comparison

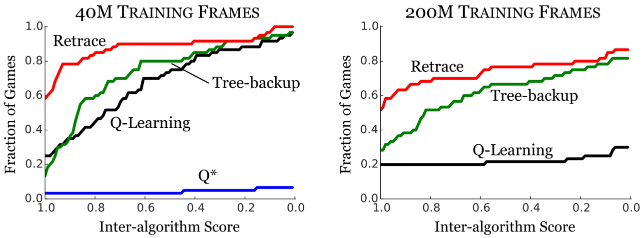


Figure 1: Inter-algorithm score distribution for λ-return ($\lambda = 1$) variants and Q-Learning ($\lambda = 0$).

- 0 and 1 of inter-algorithm scores respectively correspond to the worst and best scores for a particular game
- Retrace(λ) performs best on 30 out of 60 games

Sensitivity to λ

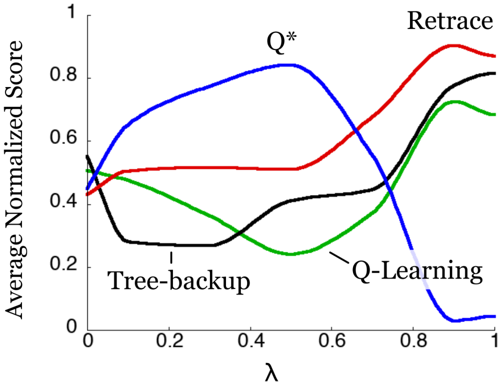


Figure 2: Average inter-algorithm scores for each value of λ. The DQN scores are fixed across different λ, but the corresponding inter-algorithm scores varies depending on the worst and best performer within each λ. **Note that average scores are not directly comparable across different values of λ.**

- Note that the Q-learning scores are fixed across different λ
- Q* performs best for small values of λ

Conclusions

- Retrace(λ)
  - is low-variance, safe and efficient
  - outperforms one-step Q-learning and existing multi-step variants on Atari 2600
  - (is already applied to A3C in another paper [Wang et al. 2016])
- Watkins's Q(λ) now has a convergence guarantee

Future work

- Estimate μ if it is unknown
- Relaxing the Markov assumption in the control case to allow $c_s > 1$:

$$c_s = \lambda \min\left(\frac{1}{c_1 \cdots c_{s-1}}, \frac{\pi(a_s|x_s)}{\mu(a_s|x_s)}\right)$$

Theorem 1

Theorem 1. *The operator $\mathcal{R}$ defined by (3) has a unique fixed point Q^π. Furthermore, if for each $a_s \in \mathcal{A}$ and each history $\mathcal{F}_s$ we have $c_s = c_s(a_s, \mathcal{F}_s) \in \left[0, \frac{\pi(a_s|x_s)}{\mu(a_s|x_s)}\right]$, then for any Q-function Q*

$$\|\mathcal{R}Q - Q^\pi\| \leq \gamma \|Q - Q^\pi\|.$$

- π and μ are stationary
- c_s can be non-Markovian

Theorem 2

Definition 1. *We say that a sequence of policies* $(\pi_k : k \in \mathbb{N})$ *is* increasingly greedy *w.r.t. a sequence* $(Q_k : k \in \mathbb{N})$ *of Q-functions if the following property holds for all k:* $P^{\pi_{k+1}} Q_{k+1} \geq P^{\pi_k} Q_{k+1}$.

Theorem 2. *Consider an arbitrary sequence of behaviour policies* (μ_k) *(which may depend on* (Q_k)*) and a sequence of target policies* (π_k) *that are increasingly greedy w.r.t. the sequence* (Q_k)*:*

$$Q_{k+1} = \mathcal{R}_k Q_k,$$

where the return operator $\mathcal{R}_k$ *is defined by (3) for* π_k *and* μ_k *and a Markovian* $c_s = c(a_s, x_s) \in [0, \frac{\pi_k(a_s|x_s)}{\mu_k(a_s|x_s)}]$*. Assume the target policies* π_k *are* ε_k*-away from the greedy policies w.r.t.* Q_k*, in the sense that* $\mathcal{T}^{\pi_k} Q_k \geq \mathcal{T} Q_k - \varepsilon_k \|Q_k\| e$*, where* e *is the vector with 1-components. Further suppose that* $\mathcal{T}^{\pi_0} Q_0 \geq Q_0$*. Then for any* $k \geq 0$*,*

$$\|Q_{k+1} - Q^*\| \leq \gamma \|Q_k - Q^*\| + \varepsilon_k \|Q_k\|.$$

In consequence, if $\varepsilon_k \to 0$ *then* $Q_k \to Q^*$.

- π is not stationary
- c_s must be Markovian

Theorem 3

Theorem 3. *Consider a sequence of sample trajectories, with the k^{th} trajectory $x_0, a_0, r_0, x_1, a_1, r_1, \ldots$ generated by following μ_k: $a_t \sim \mu_k(\cdot|x_t)$. For each (x, a) along this trajectory, with s being the time of first occurrence of (x, a), update*

$$Q_{k+1}(x,a) \leftarrow Q_k(x,a) + \alpha_k \sum_{t \geq s} \delta_t^{\pi_k} \sum_{j=s}^{t} \gamma^{t-j} \Big(\prod_{i=j+1}^{t} c_i \Big) \mathbb{I}\{x_j, a_j = x, a\}, \tag{7}$$

where $\delta_t^{\pi_k} := r_t + \gamma \mathbb{E}_{\pi_k} Q_k(x_{t+1}, \cdot) - Q_k(x_t, a_t)$, $\alpha_k = \alpha_k(x_s, a_s)$. We consider the Retrace(λ) algorithm where $c_i = \lambda \min\big(1, \frac{\pi(a_i|x_i)}{\mu(a_i|x_i)}\big)$. Assume that (π_k) are increasingly greedy w.r.t. (Q_k) and are each ε_k-away from the greedy policies (π_{Q_k}), i.e. $\max_x \|\pi_k(\cdot|x) - \pi_{Q_k}(\cdot|x)\|_1 \leq \varepsilon_k$, with $\varepsilon_k \to 0$. Assume that P^{π_k} and $P^{\pi_k \wedge \mu_k}$ asymptotically commute: $\lim_k \|P^{\pi_k} P^{\pi_k \wedge \mu_k} - P^{\pi_k \wedge \mu_k} P^{\pi_k}\| = 0$. Assume further that (1) all states and actions are visited infinitely often: $\sum_{t \geq 0} \mathbb{P}\{x_t, a_t = x, a\} \geq D > 0$, (2) the sample trajectories are finite in terms of the second moment of their lengths T_k: $\mathbb{E}_{\mu_k} T_k^2 < \infty$, (3) the stepsizes obey the usual Robbins-Munro conditions. Then $Q_k \to Q^*$ a.s.

- Convergence of sample-based online algorithm
- As a corollary, Watkins's Q(λ) converges to Q^*
  - Only c_s is different

参考文献 I

[1] Anna Harutyunyan et al. "Q(λ) with Off-Policy Corrections". In: *Proceedings of Algorithmic Learning Theory (ALT)*. 2016. arXiv: 1602.04951.

[2] Remi Munos et al. "Safe and Efficient Off-Policy Reinforcement Learning". In: *Proceedings of Neural Information Processing Systems (NIPS)*. 2016. arXiv: 1606.02647.

[3] Doina Precup, Richard S Sutton, and Satinder P Singh. "Eligibility Traces for Off-Policy Policy Evaluation". In: *ICML '00: Proceedings of the Seventeenth International Conference on Machine Learning* (2000), pp. 759–766.

[4] Ziyu Wang et al. "Sample Efficient Actor-Critic with Experience Replay". In: *arXiv* (2016), pp. 1–20. arXiv: 1611.01224.

[5] Christopher John Cornish Hellaby Watkins. "Learning from delayed rewards". PhD thesis. Cambridge University, 1989.