Safe and Efficient Off-Policy Reinforcement Learning

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

Abstract

In this work, we take a fresh look at some old and new algorithms for off-policy, return-based reinforcement learning. Expressing these in a common form, we derive a novel algorithm, Retrace(λ), with three desired properties: (1) *low variance*; (2) *safety*, as it safely uses samples collected from *any* behaviour policy, whatever its degree of "off-policyness"; and (3) *efficiency*, as it makes the best use of samples collected from near on-policy behaviour policies. We analyse the contractive nature of the related operator under both off-policy policy evaluation and control settings and derive online sample-based algorithms. To our knowledge, this is the first return-based off-policy control algorithm converging a.s. to Q^* without the GLIE assumption (Greedy in the Limit with Infinite Exploration). As a corollary, we prove the convergence of Watkins' Q(λ), which was still an open problem. We illustrate the benefits of Retrace(λ) on a standard suite of Atari 2600 games.

One fundamental trade-off in reinforcement learning lies in the definition of the update target: should one estimate Monte Carlo returns or bootstrap from an existing Q-function? Return-based methods (where *return* refers to the sum of discounted rewards $\sum_t \gamma^t r_t$) offer some advantages over value bootstrap methods: they are better behaved when combined with function approximation, and quickly propagate the fruits of exploration (Sutton, 1996). On the other hand, value bootstrap methods are more readily applied to off-policy data, a common use case. In this paper we show that *learning from returns need not be at cross-purposes with off-policy learning.*

We start from the recent work of Harutyunyan et al. (2016), who show that naive off-policy policy evaluation, without correcting for the "off-policyness" of a trajectory, still converges to the desired Q^π value function provided the behavior μ and target π policies are not too far apart (the maximum allowed distance depends on the λ parameter). Their $Q^\pi(\lambda)$ algorithm learns from trajectories generated by μ simply by summing discounted off-policy corrected rewards at each time step. Unfortunately, the assumption that μ and π are close is restrictive, as well as difficult to uphold in the control case, where the target policy is always greedy with respect to the current Q-function. In that sense this algorithm is not *safe*: it does not handle the case of arbitrary "off-policyness".

Alternatively, the Tree-backup (TB) (λ) algorithm (Precup et al., 2000) tolerates arbitrary target/behavior discrepancies by scaling information (here called *traces*) from future temporal differences by the product of target policy probabilities. TB(λ) is not *efficient* in the "near on-policy" case (similar μ and π), though, as traces may be cut prematurely, blocking learning from full returns.

In this work, we express several off-policy, return-based algorithms in a common form. From this we derive an improved algorithm, Retrace(λ), which is both *safe* and *efficient*, enjoying convergence guarantees for off-policy policy evaluation and – more importantly – for the control setting.

Retrace(λ) can learn from full returns retrieved from past policy data, as in the context of experience replay (Lin, 1993), which has returned to favour with advances in deep reinforcement learning (Mnih et al., 2015; Schaul et al., 2016). Off-policy learning is also desirable for exploration, since it allows the agent to deviate from the target policy currently under evaluation.

To the best of our knowledge, this is the first online return-based off-policy control algorithm which does not require the GLIE (Greedy in the Limit with Infinite Exploration) assumption (Singh et al., 2000). In addition, we provide as a corollary the first proof of convergence of Watkins' Q(λ) (see, e.g., Watkins, 1989; Sutton and Barto, 1998).

Finally, we illustrate the significance of Retrace(λ) in a deep learning setting by applying it to the suite of Atari 2600 games provided by the Arcade Learning Environment (Bellemare et al., 2013).

1 Notation

We consider an agent interacting with a Markov Decision Process $(\mathcal{X}, \mathcal{A}, \gamma, P, r)$. $\mathcal{X}$ is a finite state space, $\mathcal{A}$ the action space, $\gamma \in [0, 1)$ the discount factor, P the transition function mapping state-action pairs $(x, a) \in \mathcal{X} \times \mathcal{A}$ to distributions over $\mathcal{X}$, and $r : \mathcal{X} \times \mathcal{A} \to [-R_{\text{MAX}}, R_{\text{MAX}}]$ is the reward function. For notational simplicity we will consider a finite action space, but the case of infinite – possibly continuous – action space can be handled by the Retrace(λ) algorithm as well. A policy π is a mapping from $\mathcal{X}$ to a distribution over $\mathcal{A}$. A Q-function Q maps each state-action pair (x, a) to a value in $\mathbb{R}$; in particular, the reward r is a Q-function. For a policy π we define the operator P^π:

$$(P^\pi Q)(x, a) := \sum_{x' \in \mathcal{X}} \sum_{a' \in \mathcal{A}} P(x' \mid x, a)\pi(a' \mid x')Q(x', a').$$

The value function for a policy π, Q^π, describes the expected discounted sum of rewards associated with following π from a given state-action pair. Using operator notation, we write this as

$$Q^\pi := \sum_{t \geq 0} \gamma^t (P^\pi)^t r. \tag{1}$$

The *Bellman operator* $\mathcal{T}^\pi$ is

$$\mathcal{T}^\pi Q := r + \gamma P^\pi Q \tag{2}$$

and its fixed point is Q^π, i.e. $\mathcal{T}^\pi Q^\pi = Q^\pi = (I - \gamma P^\pi)^{-1} r$. The *Bellman optimality operator* introduces a maximization over the set of policies:

$$\mathcal{T} Q := r + \gamma \max_\pi P^\pi Q. \tag{3}$$

Its fixed point is Q^*, the unique *optimal value function* (Puterman, 1994). It is this quantity that we will seek to obtain when we talk about the "control setting".

Return-based Operators: The λ-return extension (Sutton, 1988) of both (2) and (3) considers exponentially weighted sums of n-step returns:

$$\mathcal{T}^\pi_\lambda Q := (1 - \lambda) \sum_{n \geq 0} \lambda^n \left[(\mathcal{T}^\pi)^n Q\right] = Q + (I - \lambda\gamma P^\pi)^{-1}(\mathcal{T}^\pi Q - Q),$$

where $\mathcal{T}^\pi Q - Q$ is the *Bellman residual* of Q for policy π, with $\mathcal{T} Q - Q$ replacing $\mathcal{T}^\pi - Q$ for (3). Examination of the above shows that Q^π is also the fixed point of $\mathcal{T}^\pi_\lambda$. At one extreme ($\lambda = 0$) we have the Bellman operator $\mathcal{T}^\pi_{\lambda=0} Q = \mathcal{T}^\pi Q$, while at the other ($\lambda = 1$) we have the policy evaluation operator $\mathcal{T}^\pi_{\lambda=1} Q = Q^\pi$ which can be estimated using Monte Carlo methods (Sutton and Barto, 1998). Intermediate values of λ trade off estimation bias with sample variance (Kearns and Singh, 2000).

We seek to evaluate a *target policy* π using trajectories drawn from a *behaviour policy* μ. If $\pi = \mu$, we are *on-policy*; otherwise, we are *off-policy*. We will consider trajectories of the form:

$$x_0 = x, a_0 = a, r_0, x_1, a_1, r_1, x_2, a_2, r_2, \ldots$$

with $a_t \sim \mu(\cdot|x_t)$, $r_t = r(x_t, a_t)$ and $x_{t+1} \sim P(\cdot|x_t, a_t)$. We denote by $\mathcal{F}_t$ this sequence up to time t, and write $\mathbb{E}_\mu$ the expectation with respect to both μ and the MDP transition probabilities. Throughout, we write $\|\cdot\|$ for supremum norm.

2 Off-Policy Algorithms

We are interested in two related off-policy learning problems. In the *policy evaluation* setting, we are given a fixed policy π whose value Q^π we wish to estimate from sample trajectories drawn from a behaviour policy μ. In the *control* setting, we consider a sequence of policies that depend on our own sequence of Q-functions (such as ε-greedy policies), and seek to approximate Q^*.

The general form that we consider for comparing several return-based off-policy algorithms is:

$$\mathcal{R}Q(x,a) := Q(x,a) + \mathbb{E}_\mu \left[\sum_{t\geq 0} \gamma^t \Big(\prod_{s=1}^{t} c_s \Big) \big(r_t + \gamma \mathbb{E}_\pi Q(x_{t+1}, \cdot) - Q(x_t, a_t) \big) \right], \tag{4}$$

for some non-negative coefficients (c_s), where we write $\mathbb{E}_\pi Q(x,\cdot) := \sum_a \pi(a|x) Q(x,a)$ and write $\big(\prod_{s=1}^t c_s\big) = 1$ when $t = 0$. By extension of the idea of eligibility traces (Sutton and Barto, 1998), we informally call the coefficients (c_s) the *traces* of the operator.

Importance sampling (IS): $c_s = \frac{\pi(a_s|x_s)}{\mu(a_s|x_s)}$**.** Importance sampling is the simplest way to correct for the discrepancy between μ and π when learning from off-policy returns (Precup et al., 2000, 2001; Geist and Scherrer, 2014). The off-policy correction uses the product of the likelihood ratios between π and μ. Notice that the $\mathcal{R}Q$ operator (4) defined with this choice of (c_s) yields Q^π for any Q. For $Q = 0$ we recover the basic IS estimate $\sum_{t\geq 0} \gamma^t \big(\prod_{s=1}^t c_s\big) r_t$, thus (4) can be seen as a variance reduction technique (with a baseline Q). It is well known that IS estimates can suffer from large – even possibly infinite – variance (mainly due to the variance of the product $\frac{\pi(a_1|x_1)}{\mu(a_1|x_1)} \cdots \frac{\pi(a_t|x_t)}{\mu(a_t|x_t)}$), which has motivated further variance reduction techniques such as (Mahmood and Sutton, 2015; Mahmood et al., 2015; Hallak et al., 2015).

Off-policy Q$^\pi(\lambda)$ and Q$^*(\lambda)$: $c_s = \lambda$**.** A recent alternative proposed by Harutyunyan et al. (2016) introduces an off-policy correction based on a Q-baseline (instead of correcting the probability of the sample path like in IS). This approach, called Q$^\pi(\lambda)$ and Q$^*(\lambda)$ for policy evaluation and control, respectively, corresponds to the choice $c_s = \lambda$. It offers the advantage of avoiding the blow-up of the variance of the product of ratios encountered with IS. Interestingly, this operator contracts around Q^π provided that μ and π are sufficiently close to each other. Defining $\varepsilon := \max_x \|\pi(\cdot|x) - \mu(\cdot|x)\|_1$ the amount of "off-policyness", the authors prove that the operator defined by (4) with $c_s = \lambda$ is a contraction mapping around Q^π for $\lambda < \frac{1-\gamma}{\gamma\varepsilon}$, and around Q^* for the worst case of $\lambda < \frac{1-\gamma}{2\gamma}$. Unfortunately, Q$^\pi(\lambda)$ requires knowledge of ε, and the condition for Q$^*(\lambda)$ is very conservative. Neither Q$^\pi(\lambda)$, nor Q$^*(\lambda)$ are safe as they do not guarantee convergence for arbitrary π and μ.

Tree-backup (TB) (λ): $c_s = \lambda \pi(a_s|x_s)$**.** The TB($\lambda$) algorithm of Precup et al. (2000) corrects for the target/behaviour discrepancy by multiplying each term of the sum by the product of target policy probabilities. The corresponding operator defines a contraction mapping (not only in expectation but also for any sample trajectory) for any policies π and μ, which makes it a safe algorithm. However, this algorithm is not efficient in the near on-policy case (where μ and π are similar) as it unnecessarily cuts the traces, preventing it to make use of full returns: we need not discount stochastic on-policy transitions (as shown by Harutyunyan et al.'s results about Q$^\pi$).

Retrace(λ): $c_s = \lambda \min\Big(1, \frac{\pi(a_s|x_s)}{\mu(a_s|x_s)}\Big)$. Our contribution is an algorithm – Retrace(λ) – that takes the best of the three previous algorithms. Retrace(λ) uses the importance sampling ratio truncated at 1. Compared to IS, it does not suffer from the variance explosion of the product of importance sampling ratios. Now, similarly to $Q^\pi(\lambda)$ and unlike TB(λ), it does not cut the traces in the on-policy case, making it possible to benefit from the full returns. In the off-policy case, the traces are safely cut, similarly to TB(λ). In particular, $\min\big(1, \frac{\pi(a_s|x_s)}{\mu(a_s|x_s)}\big) \geq \pi(a_s|x_s)$: Retrace($\lambda$) does not cut the traces as much as TB(λ).

In the subsequent sections, we will show the following:

- The Retrace(λ) operator is a γ-contraction around Q^π, for *arbitrary* policies μ and π,
- Taking c_s to be no greater than the ratio π/μ is sufficient to guarantee this property,
- Under mild assumptions, the control version of Retrace(λ), where π is replaced by a sequence of increasingly greedy policies, is also a contraction, and

	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(λ)	λ	Low	only for π close to μ	yes
TB(λ)	$\lambda\pi(a_s\|x_s)$	Low	for any π, μ	no
Retrace(λ)	$\lambda\min(1, \frac{\pi(a_s\|x_s)}{\mu(a_s\|x_s)})$	Low	for any π, μ	yes

Table 1: Properties of several algorithms defined in terms of the general operator given in (4). †Guaranteed convergence of the expected operator $\mathcal{R}$.

- The online Retrace(λ) algorithm converges a.s. to Q^* in the control case. In the control case, convergence does not require the GLIE assumption.
- As a corollary, we prove the convergence of Watkins's Q(λ) to Q^*.

3 Analysis of Retrace(λ)

We will in turn analyse both off-policy policy evaluation and control settings. We will show that $\mathcal{R}$ is a contraction mapping in both settings (under a mild additional assumption for the control case).

3.1 Policy Evaluation

Consider a fixed target policy π. For ease of exposition we consider a fixed behaviour policy μ, noting that our result extends to the setting of sequences of behaviour policies $(\mu_k : k \in \mathbb{N})$.

Our first result states the γ-contraction of the operator (4) defined by any set of non-negative coefficients $c_s = c_s(a_s, \mathcal{F}_s)$ (in order to emphasize that c_s can be a function of the whole history $\mathcal{F}_s$) under the assumption that $c_s \leq \frac{\pi(a_s|x_s)}{\mu(a_s|x_s)}$.

Theorem 1. *The operator $\mathcal{R}$ defined by (4) has a unique fixed point Q^π. Furthermore, if for each $a_s \in \mathcal{A}$ and each history $\mathcal{F}_s$ we have $c_s = c_s(a_s, \mathcal{F}_s) \in \big[0, \frac{\pi(a_s|x_s)}{\mu(a_s|x_s)}\big]$, then for any Q-function Q*

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

The following lemma will be useful in proving Theorem 1 (proof in the appendix).

Lemma 1. *The difference between $\mathcal{R}Q$ and its fixed point Q^π is*

$$\mathcal{R}Q(x,a) - Q^\pi(x,a) = \mathbb{E}_\mu\Big[\sum_{t\geq 1}\gamma^t\Big(\prod_{i=1}^{t-1} c_i\Big)\Big(\big[\mathbb{E}_\pi[(Q-Q^\pi)(x_t,\cdot)] - c_t(Q-Q^\pi)(x_t,a_t)\big]\Big)\Big].$$

Proof (Theorem 1). The fact that Q^π is the fixed point of the operator $\mathcal{R}$ is obvious from (4) since $\mathbb{E}_{x_{t+1}\sim P(\cdot|x_t,a_t)}\big[r_t + \gamma\mathbb{E}_\pi Q^\pi(x_{x+1},\cdot) - Q^\pi(x_t,a_t)\big] = (\mathcal{T}^\pi Q^\pi - Q^\pi)(x_t,a_t) = 0$, since Q^π is the fixed point of $\mathcal{T}^\pi$. Now, from Lemma 1, and defining $\Delta Q := Q - Q^\pi$, we have

$$\begin{aligned}
\mathcal{R}Q(x,a) - Q^\pi(x,a) &= \sum_{t\geq 1}\gamma^t\mathbb{E}_{x_{1:t},a_{1:t}}\Big[\Big(\prod_{i=1}^{t-1} c_i\Big)\Big(\big[\mathbb{E}_\pi\Delta Q(x_t,\cdot) - c_t\Delta Q(x_t,a_t)\big]\Big)\Big] \\
&= \sum_{t\geq 1}\gamma^t\mathbb{E}_{x_{1:t},a_{1:t-1}}\Big[\Big(\prod_{i=1}^{t-1} c_i\Big)\Big(\big[\mathbb{E}_\pi\Delta Q(x_t,\cdot) - \mathbb{E}_{a_t}[c_t(a_t,\mathcal{F}_t)\Delta Q(x_t,a_t)|\mathcal{F}_t]\big]\Big)\Big] \\
&= \sum_{t\geq 1}\gamma^t\mathbb{E}_{x_{1:t},a_{1:t-1}}\Big[\Big(\prod_{i=1}^{t-1} c_i\Big)\sum_b\big(\pi(b|x_t) - \mu(b|x_t)c_t(b,\mathcal{F}_t)\big)\Delta Q(x_t,b)\Big].
\end{aligned}$$

Now since $\pi(a|x_t) - \mu(a|x_t)c_t(b, \mathcal{F}_t) \geq 0$, we have that $\mathcal{R}Q(x,a) - Q^\pi(x,a) = \sum_{y,b} w_{y,b} \Delta Q(y,b)$, i.e. a linear combination of $\Delta Q(y,b)$ weighted by non-negative coefficients:

$$w_{y,b} := \sum_{t\geq 1} \gamma^t \mathbb{E}_{x_{1:t}, a_{1:t-1}} \Big[\Big(\prod_{i=1}^{t-1} c_i \Big) \big(\pi(b|x_t) - \mu(b|x_t) c_t(b, \mathcal{F}_t) \big) \mathbb{I}\{x_t = y\} \Big].$$

The sum of those coefficients is:

$$\begin{aligned}
\sum_{y,b} w_{y,b} &= \sum_{t\geq 1} \gamma^t \mathbb{E}_{x_{1:t}, a_{1:t-1}} \Big[\Big(\prod_{i=1}^{t-1} c_i \Big) \sum_b \big(\pi(b|x_t) - \mu(b|x_t) c_t(b, \mathcal{F}_t) \big) \Big] \\
&= \sum_{t\geq 1} \gamma^t \mathbb{E}_{x_{1:t}, a_{1:t-1}} \Big[\Big(\prod_{i=1}^{t-1} c_i \Big) \mathbb{E}_{a_t}[1 - c_t(a_t, \mathcal{F}_t) | \mathcal{F}_t] \Big] = \sum_{t\geq 1} \gamma^t \mathbb{E}_{x_{1:t}, a_{1:t}} \Big[\Big(\prod_{i=1}^{t-1} c_i \Big) (1 - c_t) \Big] \\
&= \mathbb{E}_\mu \Big[\sum_{t\geq 1} \gamma^t \Big(\prod_{i=1}^{t-1} c_i \Big) - \sum_{t\geq 1} \gamma^t \Big(\prod_{i=1}^{t} c_i \Big) \Big] = \gamma C - (C - 1),
\end{aligned}$$

where $C := \mathbb{E}_\mu \big[\sum_{t\geq 0} \gamma^t \big(\prod_{i=1}^t c_i \big) \big]$. Since $C \geq 1$, we have that $\sum_{y,b} w_{y,b} \leq \gamma$. Thus $\mathcal{R}Q(x,a) - Q^\pi(x,a)$ is a sub-convex combination of $\Delta Q(y,b)$ weighted by non-negative coefficients $w_{y,b}$ which sum to (at most) γ, thus $\mathcal{R}$ is a γ-contraction mapping around Q^π. □

Remark 1. *Notice that the coefficient C in the proof of Theorem 1 depends on (x,a). If we let $\eta(x,a) := 1 - (1-\gamma)\mathbb{E}_\mu \Big[\sum_{t\geq 0} \gamma^t (\prod_{s=1}^t c_s) \Big]$, then we have shown that*

$$|\mathcal{R}Q(x,a) - Q^\pi(x,a)| \leq \eta(x,a) \|Q - Q^\pi\|.$$

Thus $\eta(x,a) \in [0, \gamma]$ is a (x,a)-specific contraction coefficient, which is γ when $c_1 = 0$ (the trace is cut immediately) and can be close to zero when learning from full returns ($c_t \approx 1$ for all t).

3.2 Control

In the control setting, the single target policy π is replaced by a sequence of policies which depend on Q_k. While most prior work has focused on strictly greedy policies, here we consider the larger class of *increasingly greedy* sequences. We now make this notion precise.

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:*

$$P^{\pi_{k+1}} Q_{k+1} \geq P^{\pi_k} Q_{k+1}.$$

Intuitively, this means that each π_{k+1} is at least as greedy as the previous policy π_k for Q_{k+1}. Many natural sequences of policies are increasingly greedy, including ε_k-greedy policies (with non-increasing ε_k) and softmax policies (with non-increasing temperature). See proofs in the appendix.

We will assume that $c_s = c_s(a_s, \mathcal{F}_s) = c(a_s, x_s)$ is Markovian, in the sense that it depends on x_s, a_s (as well as the policies π and μ) only but not on the full past history. This allows us to define the (sub)-probability transition operator

$$(P^{c\mu} Q)(x,a) := \sum_{x'} \sum_{a'} p(x'|x,a) \mu(a'|x') c(a', x') Q(x', a').$$

Finally, an additional requirement to the convergence in the control case, we assume that Q_0 satisfies $\mathcal{T}^{\pi_0} Q_0 \geq Q_0$ (this can be achieved by a pessimistic initialization $Q_0 = -R_{MAX}/(1-\gamma)$).

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 (4) for π_k and μ_k and a Markovian $c_s = c(a_s, x_s) \in [0, \frac{\pi(a_s|x_s)}{\mu(a_s|x_s)}]$. Assume the target policies π_k are ε_k-away from the greeedy 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 $Q_k \to Q^*$.

Sketch of Proof (The full proof is in the appendix). Using $P^{c\mu_k}$, the Retrace(λ) operator rewrites

$$\mathcal{R}_k Q = Q + \sum_{t\geq 0}\gamma^t (P^{c\mu_k})^t(\mathcal{T}^{\pi_k}Q - Q) = Q + (I-\gamma P^{c\mu_k})^{-1}(\mathcal{T}^{\pi_k}Q - Q).$$

We now lower- and upper-bound the term $Q_{k+1}-Q^*$.

Upper bound on $Q_{k+1}-Q^*$. We prove that $Q_{k+1}-Q^* \leq A_k(Q_k - Q^*)$ with $A_k := \gamma(I-\gamma P^{c\mu_k})^{-1}\big[P^{\pi_k} - P^{c\mu_k}\big]$. Since $c_t \in [0, \frac{\pi(a_t|x_t)}{\mu(a_t|x_t)}]$ we deduce that A_k has non-negative elements, whose sum over each row, is at most γ. Thus

$$Q_{k+1} - Q^* \leq \gamma \|Q_k - Q^*\| e. \tag{5}$$

Lower bound on $Q_{k+1}-Q^*$. Using the fact that $\mathcal{T}^{\pi_k}Q_k \geq \mathcal{T}^{\pi^*}Q_k - \varepsilon_k\|Q_k\|e$ we have

$$\begin{aligned} Q_{k+1}-Q^* &\geq Q_{k+1} - \mathcal{T}^{\pi_k}Q_k + \gamma P^{\pi^*}(Q_k - Q^*) - \gamma\varepsilon_k\|Q_k\|e \\ &= \gamma P^{c\mu_k}(I-\gamma P^{c\mu_k})^{-1}(\mathcal{T}^{\pi_k}Q_k - Q_k) + \gamma P^{\pi^*}(Q_k - Q^*) - \varepsilon_k\|Q_k\|e. \end{aligned} \tag{6}$$

Lower bound on $\mathcal{T}^{\pi_k}Q_k - Q_k$. Since (π_k) is increasingly greedy, we have

$$\begin{aligned} \mathcal{T}^{\pi_{k+1}}Q_{k+1} - Q_{k+1} &\geq \mathcal{T}^{\pi_k}Q_{k+1} - Q_{k+1} = r + (\gamma P^{\pi_k} - I)\mathcal{R}Q_k \\ &= B_k(\mathcal{T}^{\pi_k}Q_k - Q_k), \end{aligned} \tag{7}$$

where $B_k := \gamma[P^{\pi_k} - P^{c\mu_k}](I-\gamma P^{c\mu_k})^{-1}$. Since $P^{\pi_k} - P^{c\mu_k}$ and $(I-\gamma P^{c\mu_k})^{-1}$ are non-negative matrices, so is B_k. Thus

$$\mathcal{T}^{\pi_k}Q_k - Q_k \geq B_{k-1}B_{k-2}\dots B_0(\mathcal{T}^{\pi_0}Q_0 - Q_0) \geq 0,$$

since we assumed $T^{\pi_0}Q_0 - Q_0 \geq 0$. Thus, (6) implies that

$$Q_{k+1} - Q^* \quad \geq \quad \gamma P^{\pi^*}(Q_k - Q^*) - \varepsilon_k\|Q_k\|e.$$

and combining the above with (5) we deduce $\|Q_{k+1}-Q^*\| \leq \gamma\|Q_k - Q^*\| + \varepsilon_k\|Q_k\|$. When $\varepsilon \to 0$, we further deduce that Q_k are bounded, thus $Q_k \to Q^*$. $\square$

3.3 Online algorithms

So far we have analysed the contraction properties of the expected $\mathcal{R}$ operators. We now describe online algorithms which can learn from sample trajectories. We analyze the algorithms in the *every visit* form (Sutton and Barto, 1998), which is the more practical generalization of the first-visit form. In this section, we will only consider the Retrace(λ) algorithm defined with the coefficient $c = \lambda\min(1, \pi/\mu)$. For that c, let us rewrite the operator $P^{c\mu}$ as $\lambda P^{\pi\wedge\mu}$, where $P^{\pi\wedge\mu}Q(x,a) := \sum_y\sum_b \min(\pi(b|y),\mu(b|y))Q(y,b)$, and write the Retrace operator $\mathcal{R}Q = Q + (I - \lambda\gamma P^{\pi\wedge\mu})^{-1}(\mathcal{T}^\pi Q - Q)$. We focus on the control case, noting that a similar (and more general) result can be derived for policy evaluation.

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

$$Q_{k+1}(x,a) \leftarrow Q_k(x,a) + \alpha_k \sum_{t\geq s}\delta_t^{\pi_k}\sum_{j=s}^{t}\gamma^{t-j}\Big(\prod_{i=j+1}^{t}c_i\Big)\mathbb{I}\{x_j, a_j = x, a\}, \tag{8}$$

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.*

The proof extends similar convergence proofs of TD(λ) by Bertsekas and Tsitsiklis (1996) and of optimistic policy iteration by Tsitsiklis (2003), and is provided in the appendix. Notice that compared to Theorem 2 we do not assume that $\mathcal{T}^{\pi_0}Q_0 - Q_0 \geq 0$ here. However, we make the additional (rather technical) assumption that P^{π_k} and $P^{\pi_k \wedge \mu_k}$ commute at the limit. This is satisfied for example when the probability assigned by the behavior policy $\mu_k(\cdot|x)$ to the greedy action $\pi_{Q_k}(x)$ is independent of x. Examples include ε-greedy policies, or more generally mixtures between the greedy policy π_{Q_k} and an arbitrary distribution μ (see Lemma 5 in the appendix for the proof):

$$\mu_k(a|x) = \varepsilon \frac{\mu(a|x)}{1 - \mu(\pi_{Q_k}(x)|x)} \mathbb{I}\{a \neq \pi_{Q_k}(x)\} + (1-\varepsilon)\mathbb{I}\{a = \pi_{Q_k}(x)\}. \tag{9}$$

Notice that the mixture coefficient ε needs not go to 0.

4 Discussion of the results

4.1 Choice of the trace coefficients c_s

Theorems 1 and 2 ensure convergence to Q^π and Q^* for any trace coefficient $c_s \in [0, \frac{\pi(a_s|x_s)}{\mu(a_s|x_s)}]$. However, to make the best choice of c_s, we need to consider the *speed* of convergence, which depends on both (1) the variance of the online estimate, which indicates how many online updates are required in a single iteration of $\mathcal{R}$, and (2) the contraction coefficient of $\mathcal{R}$.

Variance The variance of the estimate strongly depends on the variance of the product trace $(c_1 \dots c_t)$, which is not an easy quantity to control in general, as the (c_s) are usually not independent. However, assuming independence and stationarity of (c_s), we have that $\mathbb{V}\big(\sum_t \gamma^t c_1 \dots c_t\big)$ is at least $\sum_t \gamma^{2t}\mathbb{V}(c)^t$, which is finite only if $\mathbb{V}(c) < 1/\gamma^2$. Thus, an important requirement for a numerically stable algorithm is for $\mathbb{V}(c)$ to be as small as possible, and certainly no more than $1/\gamma^2$. This rules out importance sampling (for which $c \propto \frac{\pi(a|x)}{\mu(a|x)}$, and $\mathbb{V}(c|x) \propto \sum_a \mu(a|x)\big(\frac{\pi(a|x)}{\mu(a|x)} - 1\big)^2 = \sum_a \frac{\pi(a_t|x_t)^2}{\mu(a_t|x_t)} - 1$, which may be larger than $1/\gamma^2$ for some π and μ), and is the reason we take $c_s \leq 1$.

Contraction speed The contraction coefficient $\eta \in [0, \gamma]$ of $\mathcal{R}$ (see Remark 1) depends on how much the traces have been cut, and should be as small as possible (since it takes $\log(1/\varepsilon)/\log(1/\eta)$ iterations of $\mathcal{R}$ to obtain an ε-approximation). It is smallest when the traces are not cut at all (i.e. if $c_s = 1$ for all s, $\mathcal{R}$ is the policy evaluation operator which produces Q^π in a single iteration). Indeed, when the traces are cut, we do not benefit from learning from full returns (in the extreme, $c_1 = 0$ and $\mathcal{R}$ reduces to the Bellman operator with $\eta = \gamma$). Although (c_s) should be as large as possible, they probably should not be larger than 1, or the update rule would consider the future to be more important than the present. A reasonable trade-off between low variance (when c_s are small) and high contraction speed (when c_s are large) is given by Retrace(λ), for which we provde the convergence of the online algorithm.

If we relax the assumption that the trace is Markovian (in which case only the result for policy evaluation has been proven so far) we could trade off a low trace at some time for a possibly larger-than-1 trace at another time, as long as their product is less than 1. A possible choice could be

$$c_t = \lambda \min\Big(\frac{1}{c_1 \dots c_{t-1}}, \frac{\pi(a_t|x_t)}{\mu(a_t|x_t)}\Big). \tag{10}$$

4.2 Other topics of discussion

No GLIE assumption. The crucial point of Theorem 2 is that convergence to Q^* occurs for *arbitrary* behaviour policies. Thus the online result in Theorem 3 does not require the behaviour policies to become greedy in the limit of infinite exploration (i.e. GLIE assumption, Singh et al., 2000). We believe Theorem 3 provides the first convergence result to Q^* for a λ-return (with $\lambda > 0$) algorithm that does not require this (hard to satisfy) assumption.

Proof of Watkins' Q(λ). As a corollary of Theorem 3 when selecting our target policies π_k to be greedy w.r.t. Q_k (i.e. $\varepsilon_k = 0$), we deduce that Watkins' Q(λ) (e.g., Watkins, 1989; Sutton and Barto, 1998) converges a.s. to Q^* (under the assumption that μ_k commutes asymptotically with the greedy policies, which is satisfied for e.g. μ_k defined by (9)). We believe this is the first such proof.

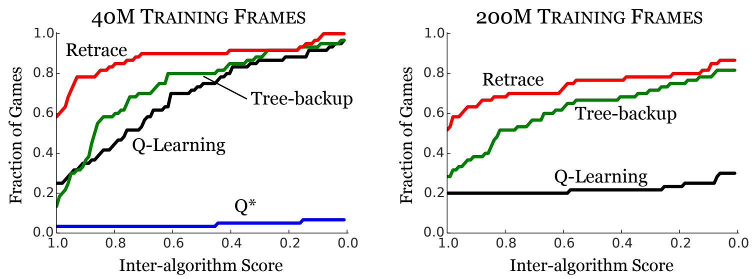

Figure 1: Inter-algorithm score distribution for λ-return ($\lambda = 1$) variants and Q-Learning ($\lambda = 0$).

Increasingly greedy policies The assumption that the sequence of target policies (π_k) is increasingly greedy w.r.t. the sequence of (Q_k) is more general that just considering greedy policies w.r.t. (Q_k) (which is Watkins's Q(λ)), and may be more efficient as well. Indeed, using non-greedy target policies π_k can speed up convergence as the traces will not be cut as frequently. Of course, in order to converge to Q^*, we eventually need the target policies (and not the behaviour policies, as mentioned above) to become greedy in the limit (i.e. $\varepsilon_k \to 0$ as defined in Theorem 2).

Comparison to $Q^\pi(\lambda)$. Unlike Retrace(λ), Q^π does not need to know the behaviour policy μ. However, it fails to converge when μ is far from π. Retrace(λ) uses its knowledge of μ (for the chosen actions) to cut the traces and safely handle arbitrary policies π and μ.

Comparison to TB(λ). Similarly to Q^π, TB(λ) does not need the knowledge of the behaviour policy μ. But as a consequence, TB(λ) is not able to benefit from possible near on-policy situations, cutting traces unnecessarily when π and μ are close.

Continuous action space. Let us mention that Theorems 1 and 2 extend to the case of (measurable) continuous or infinite action spaces. The trace coefficients will make use of the densities $\min(1, d\pi/d\mu)$ instead of the probabilities $\min(1, \pi/\mu)$. This would not be possible with TB(λ).

Open questions include: (1) Removing the technical assumption that P^{π_k} and $P^{\pi_k \wedge \mu_k}$ asymptotically commute, (2) Relaxing the Markov assumption in the control case in order to allow trace coefficients c_t of the form (10).

5 Experimental Results

To validate our theoretical results, we employ Retrace(λ) in an experience replay (Lin, 1993) setting, where sample transitions are stored within a large but bounded *replay memory* and subsequently replayed as if they were new experience. Naturally, older data in the memory is usually drawn from a policy which differs from the current policy, offering an excellent point of comparison for the algorithms presented in Section 2.

Our agent adapts the DQN architecture of Mnih et al. (2015) to replay short sequences from the memory (details in Appendix F) instead of single transitions. The Q-function target for a sample sequence $x_t, a_t, r_t, \cdots, x_{t+k}$ is

$$\Delta Q(x_t, a_t) = \sum_{s=t}^{k-1} \gamma^{s-t} \Big(\prod_{i=t+1}^{s} c_i \Big) \big[r(x_s, a_s) + \gamma \mathbb{E}_\pi Q(x_{s+1}, \cdot) - Q(x_s, a_s) \big].$$

We compare our algorithms' performance on 60 different Atari 2600 games in the Arcade Learning Environment (Bellemare et al., 2013) using Bellemare et al.'s inter-algorithm score distribution. Inter-algorithm scores are normalized so that 0 and 1 respectively correspond to the worst and best score for a particular game, within the set of algorithms under comparison. If $g \in \{1, \ldots, 60\}$ is a game and $z_{g,a}$ the inter-algorithm score on g for algorithm a, then the score distribution function is $f(x) := |\{g : z_{g,a} \geq x\}|/60$. Roughly, a strictly higher curve corresponds to a better algorithm.

Across values of λ, $\lambda = 1$ performs best, save for Q^* where $\lambda = 0.5$ obtains slightly superior performance. However, Q^* diverges for larger λ values (see Figure 1, left), and yields poor performance for smaller ones. Both Retrace and TB(λ) achieve dramatically higher performance than

Q-Learning early on and maintain their advantage throughout. Compared to TB(λ), Retrace(λ) offers a narrower but still marked advantage, being the best performer on 30 games; TB(λ) claims 15 of the remainder. Per-game performance details appear in Table 2 in Appendix F.

References

Bellemare, M. G., Naddaf, Y., Veness, J., and Bowling, M. (2013). The Arcade Learning Environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279.

Bertsekas, D. P. and Tsitsiklis, J. N. (1996). *Neuro-Dynamic Programming*. Athena Scientific.

Geist, M. and Scherrer, B. (2014). Off-policy learning with eligibility traces: A survey. *The Journal of Machine Learning Research*, 15(1):289–333.

Hallak, A., Tamar, A., Munos, R., and Mannor, S. (2015). Generalized emphatic temporal difference learning: Bias-variance analysis. *arXiv:1509.05172*.

Harutyunyan, A., Bellemare, M. G., Stepleton, T., and Munos, R. (2016). Q(λ) with off-policy corrections. *arXiv:1602.04951*.

Kearns, M. J. and Singh, S. P. (2000). Bias-variance error bounds for temporal difference updates. In *Conference on Computational Learning Theory*, pages 142–147.

Lin, L. (1993). Scaling up reinforcement learning for robot control. In *Machine Learning: Proceedings of the Tenth International Conference*, pages 182–189.

Mahmood, A. R. and Sutton, R. S. (2015). Off-policy learning based on weighted importance sampling with linear computational complexity. In *Conference on Uncertainty in Artificial Intelligence*.

Mahmood, A. R., Yu, H., White, M., and Sutton, R. S. (2015). Emphatic temporal-difference learning. *arXiv:1507.01569*.

Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T. P., Harley, T., Silver, D., and Kavukcuoglu, K. (2016). Asynchronous methods for deep reinforcement learning. *arXiv:1602.01783*.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. (2015). Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533.

Precup, D., Sutton, R. S., and Dasgupta, S. (2001). Off-policy temporal-difference learning with function approximation. In *International Conference on Machine Laerning*, pages 417–424.

Precup, D., Sutton, R. S., and Singh, S. (2000). Eligibility traces for off-policy policy evaluation. In *Proceedings of the Seventeenth International Conference on Machine Learning*.

Puterman, M. L. (1994). *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., New York, NY, USA, 1st edition.

Schaul, T., Quan, J., Antonoglou, I., and Silver, D. (2016). Prioritized experience replay. In *International Conference on Learning Representations*.

Singh, S., Jaakkola, T., Littman, M. L., and Szepesvári, C. (2000). Convergence results for single-step on-policy reinforcement-learning algorithms. *Machine Learning*, 38(3):287–308.

Sutton, R. and Barto, A. (1998). *Reinforcement learning: An introduction*, volume 116. Cambridge Univ Press.

Sutton, R. S. (1988). Learning to predict by the methods of temporal differences. *Machine learning*, 3(1):9–44.

Sutton, R. S. (1996). Generalization in reinforcement learning: Successful examples using sparse coarse coding. In *Advances in Neural Information Processing Systems 8*.

Tsitsiklis, J. N. (2003). On the convergence of optimistic policy iteration. *Journal of Machine Learning Research*, 3:59–72.

Watkins, C. J. C. H. (1989). *Learning from Delayed Rewards*. PhD thesis, King's College, Cambridge, UK.

A Proof of Lemma 1

Proof (Lemma 1). Let $\Delta Q := Q - Q^\pi$. We begin by rewriting (4):

$$\mathcal{R}Q(x,a) = \sum_{t\geq 0} \gamma^t \mathbb{E}_\mu \left[\Big(\prod_{s=1}^{t} c_s \Big) \Big(r_t + \gamma \left[\mathbb{E}_\pi Q(x_{t+1},\cdot) - c_{t+1} Q(x_{t+1}, a_{t+1}) \Big) \right] \right].$$

Since Q^π is the fixed point of $\mathcal{R}$, we have

$$Q^\pi(x,a) = \mathcal{R}Q^\pi(x,a) = \sum_{t\geq 0} \gamma^t \mathbb{E}_\mu \left[\Big(\prod_{s=1}^{t} c_s \Big) \Big(r_t + \gamma \left[\mathbb{E}_\pi Q^\pi(x_{t+1},\cdot) - c_{t+1} Q^\pi(x_{t+1}, a_{t+1}) \Big) \right] \right],$$

from which we deduce that

$$\begin{aligned}
\mathcal{R}Q(x,a) - Q^\pi(x,a) &= \sum_{t\geq 0} \gamma^t \mathbb{E}_\mu \Big[\Big(\prod_{s=1}^{t} c_s \Big) \Big(\gamma \big[\mathbb{E}_\pi \Delta Q(x_{t+1},\cdot) - c_{t+1} \Delta Q(x_{t+1}, a_{t+1}) \big] \Big) \Big] \\
&= \sum_{t\geq 1} \gamma^t \mathbb{E}_\mu \Big[\Big(\prod_{s=1}^{t-1} c_s \Big) \Big(\big[\mathbb{E}_\pi \Delta Q(x_t,\cdot) - c_t \Delta Q(x_t, a_t) \big] \Big) \Big].
\end{aligned}$$

□

B Increasingly greedy policies

Recall the definition of an increasingly greedy sequence of policies.

Definition 2. *We say that a sequence of policies (π_k) is increasingly greedy w.r.t. a sequence of functions (Q_k) if the following property holds for all k:*

$$P^{\pi_{k+1}} Q_{k+1} \geq P^{\pi_k} Q_{k+1}.$$

It is obvious to see that this property holds if all policies π_k are greedy w.r.t. Q_k. Indeed in such case, $\mathcal{T}^{\pi_{k+1}} Q_{k+1} = \mathcal{T} Q_{k+1} \geq \mathcal{T}^\pi Q_{k+1}$ for any π.

We now prove that this property holds for ε_k-greedy policies (with non-increasing (ε_k)) as well as soft-max policies (with non-decreasing (β_k)), as stated in the two lemmas below.

Of course not all policies satisfy this property (a counter-example being $\pi_k(a|x) := \arg\min_{a'} Q_k(x, a')$).

Lemma 2. *Let (ε_k) be a non-increasing sequence. Then the sequence of policies (π_k) which are ε_k-greedy w.r.t. the sequence of functions (Q_k) is increasingly greedy w.r.t. that sequence.*

Proof. From the definition of an ε-greedy policy we have:

$$\begin{aligned}
P^{\pi_{k+1}} Q_{k+1}(x,a) &= \sum_y p(y|x,a) \big[(1-\varepsilon_{k+1}) \max_b Q_{k+1}(y,b) + \varepsilon_{k+1} \frac{1}{A} \sum_b Q_{k+1}(y,b) \big] \\
&\geq \sum_y p(y|x,a) \big[(1-\varepsilon_k) \max_b Q_{k+1}(y,b) + \varepsilon_k \frac{1}{A} \sum_b Q_{k+1}(y,b) \big] \\
&\geq \sum_y p(y|x,a) \big[(1-\varepsilon_k) Q_{k+1}(y, \arg\max_b Q_k(y,b)) + \varepsilon \frac{1}{A} \sum_b Q_{k+1}(y,b) \big] \\
&= P^{\pi_k} Q_{k+1},
\end{aligned}$$

where we used the fact that $\varepsilon_{k+1} \leq \varepsilon_k$. □

Lemma 3. *Let (β_k) be a non-decreasing sequence of soft-max parameters. Then the sequence of policies (π_k) which are soft-max (with parameter β_k) w.r.t. the sequence of functions (Q_k) is increasingly greedy w.r.t. that sequence.*

Proof. For any Q and y, define $\pi_\beta(b) = \frac{e^{\beta Q(y,b)}}{\sum_{b'} e^{\beta Q(y,b')}}$ and $f(\beta) = \sum_b \pi_\beta(b) Q(y,b)$. Then we have

$$\begin{aligned} f'(\beta) &= \sum_b \big[\pi_\beta(b)Q(y,b) - \pi_\beta(b)\sum_{b'}\pi_\beta(b')Q(y,b')\big]Q(y,b) \\ &= \sum_b \pi_\beta(b)Q(y,b)^2 - \big(\sum_b \pi_\beta(b)Q(y,b)\big)^2 \\ &= \mathbb{V}_{b\sim\pi_\beta}\big[Q(y,b)\big] \geq 0. \end{aligned}$$

Thus $\beta \mapsto f(\beta)$ is a non-decreasing function, and since $\beta_{k+1} \geq \beta_k$, we have

$$\begin{aligned} P^{\pi_{k+1}}Q_{k+1}(x,a) &= \sum_y p(y|x,a)\sum_b \frac{e^{\beta_{k+1}Q_{k+1}(y,b)}}{\sum_{b'} e^{\beta_{k+1}Q_{k+1}(y,b')}} Q_{k+1}(y,b) \\ &\geq \sum_y p(y|x,a)\sum_b \frac{e^{\beta_k Q_{k+1}(y,b)}}{\sum_{b'} e^{\beta_k Q_{k+1}(y,b')}} Q_{k+1}(y,b) \\ &= P^{\pi_k}Q_{k+1}(x,a). \end{aligned}$$

□

C Proof of Theorem 2

As mentioned in the main text, since c_s is Markovian, we can define the (sub)-probability transition operator

$$(P^{c\mu}Q)(x,a) := \sum_{x'}\sum_{a'} p(x'|x,a)\mu(a'|x')c(a',x')Q(x',a').$$

The Retrace(λ) operator then writes

$$\mathcal{R}_k Q = Q + \sum_{t\geq 0}\gamma^t (P^{c\mu_k})^t(\mathcal{T}^{\pi_k}Q - Q) = Q + (I - \gamma P^{c\mu_k})^{-1}(\mathcal{T}^{\pi_k}Q - Q).$$

Proof. We now lower- and upper-bound the term $Q_{k+1} - Q^*$.

Upper bound on $Q_{k+1} - Q^*$. Since $Q_{k+1} = \mathcal{R}_k Q_k$, we have

$$\begin{aligned} Q_{k+1} - Q^* &= Q_k - Q^* + (I - \gamma P^{c\mu_k})^{-1}\big[\mathcal{T}^{\pi_k}Q_k - Q_k\big] \\ &= (I - \gamma P^{c\mu_k})^{-1}\big[\mathcal{T}^{\pi_k}Q_k - Q_k + (I - \gamma P^{c\mu_k})(Q_k - Q^*)\big] \\ &= (I - \gamma P^{c\mu_k})^{-1}\big[\mathcal{T}^{\pi_k}Q_k - Q^* - \gamma P^{c\mu_k}(Q_k - Q^*)\big] \\ &= (I - \gamma P^{c\mu_k})^{-1}\big[\mathcal{T}^{\pi_k}Q_k - \mathcal{T}Q^* - \gamma P^{c\mu_k}(Q_k - Q^*)\big] \\ &\leq (I - \gamma P^{c\mu_k})^{-1}\big[\gamma P^{\pi_k}(Q_k - Q^*) - \gamma P^{c\mu_k}(Q_k - Q^*)\big] \\ &= \gamma(I - \gamma P^{c\mu_k})^{-1}\big[P^{\pi_k} - P^{c\mu_k}\big](Q_k - Q^*), \\ &= A_k(Q_k - Q^*), \end{aligned} \tag{11}$$

where $A_k := \gamma(I - \gamma P^{c\mu_k})^{-1}\big[P^{\pi_k} - P^{c\mu_k}\big]$.

Now let us prove that A_k has non-negative elements, whose sum over each row is at most γ. Let e be the vector with 1-components. By rewriting A_k as $\gamma\sum_{t\geq 0}\gamma^t(P^{c\mu_k})^t(P^{\pi_k} - P^{c\mu_k})$ and noticing that

$$(P^{\pi_k} - P^{c\mu_k})e(x,a) = \sum_{x'}\sum_{a'} p(x'|x,a)[\pi_k(a'|x') - c(a',x')\mu_k(a'|x')] \geq 0, \tag{12}$$

it is clear that all elements of A_k are non-negative. We have

$$\begin{aligned} A_k e &= \gamma \sum_{t\geq 0} \gamma^t (P^{c\mu_k})^t \big[P^{\pi_k} - P^{c\mu_k}\big] e \\ &= \gamma \sum_{t\geq 0} \gamma^t (P^{c\mu_k})^t e - \sum_{t\geq 0} \gamma^{t+1} (P^{c\mu_k})^{t+1} e \\ &= e - (1-\gamma) \sum_{t\geq 0} \gamma^t (P^{c\mu_k})^t e \\ &\leq \gamma e, \end{aligned} \tag{13}$$

(since $\sum_{t\geq 0} \gamma^t (P^{c\mu_k})^t e \geq e$). Thus A_k has non-negative elements, whose sum over each row, is at most γ. We deduce from (11) that $Q_{k+1} - Q^*$ is upper-bounded by a sub-convex combination of components of $Q_k - Q^*$; the sum of their coefficients is at most γ. Thus

$$Q_{k+1} - Q^* \leq \gamma \|Q_k - Q^*\| e. \tag{14}$$

Lower bound on $Q_{k+1} - Q^*$. We have

$$\begin{aligned} Q_{k+1} &= Q_k + (I - \gamma P^{c\mu_k})^{-1} (\mathcal{T}^{\pi_k} Q_k - Q_k) \\ &= Q_k + \sum_{i\geq 0} \gamma^i (P^{c\mu_k})^i (\mathcal{T}^{\pi_k} Q_k - Q_k) \\ &= \mathcal{T}^{\pi_k} Q_k + \sum_{i\geq 1} \gamma^i (P^{c\mu_k})^i (\mathcal{T}^{\pi_k} Q_k - Q_k) \\ &= \mathcal{T}^{\pi_k} Q_k + \gamma P^{c\mu_k} (I - \gamma P^{c\mu_k})^{-1} (\mathcal{T}^{\pi_k} Q_k - Q_k). \end{aligned} \tag{15}$$

Now, from the definition of ε_k we have $\mathcal{T}^{\pi_k} Q_k \geq \mathcal{T} Q_k - \varepsilon_k \|Q_k\| \geq \mathcal{T}^{\pi^*} Q_k - \varepsilon_k \|Q_k\|$, thus

$$\begin{aligned} Q_{k+1} - Q^* &= Q_{k+1} - \mathcal{T}^{\pi_k} Q_k + \mathcal{T}^{\pi_k} Q_k - \mathcal{T}^{\pi^*} Q_k + \mathcal{T}^{\pi^*} Q_k - \mathcal{T}^{\pi^*} Q^* \\ &\geq Q_{k+1} - \mathcal{T}^{\pi_k} Q_k + \gamma P^{\pi^*} (Q_k - Q^*) - \varepsilon_k \|Q_k\| e \end{aligned}$$

Using (15) we derive the lower bound:

$$Q_{k+1} - Q^* \geq \gamma P^{c\mu_k} (I - \gamma P^{c\mu_k})^{-1} (\mathcal{T}^{\pi_k} Q_k - Q_k) + \gamma P^{\pi^*} (Q_k - Q^*) - \varepsilon_k \|Q_k\|. \tag{16}$$

Lower bound on $\mathcal{T}^{\pi_k} Q_k - Q_k$. By hypothesis, (π_k) is increasingly greedy w.r.t. (Q_k), thus

$$\begin{aligned} \mathcal{T}^{\pi_{k+1}} Q_{k+1} - Q_{k+1} &\geq \mathcal{T}^{\pi_k} Q_{k+1} - Q_{k+1} \\ &= \mathcal{T}^{\pi_k} \mathcal{R} Q_k - \mathcal{R} Q_k \\ &= r + (\gamma P^{\pi_k} - I) \mathcal{R} Q_k \\ &= r + (\gamma P^{\pi_k} - I) \big[Q_k + (I - \gamma P^{c\mu_k})^{-1} (\mathcal{T}^{\pi_k} Q_k - Q_k)\big] \\ &= \mathcal{T}^{\pi_k} Q_k - Q_k + (\gamma P^{\pi_k} - I)(I - \gamma P^{c\mu_k})^{-1} (\mathcal{T}^{\pi_k} Q_k - Q_k) \\ &= \gamma \big[P^{\pi_k} - P^{c\mu_k}\big] (I - \gamma P^{c\mu_k})^{-1} (\mathcal{T}^{\pi_k} Q_k - Q_k) \\ &= B_k (\mathcal{T}^{\pi_k} Q_k - Q_k), \end{aligned} \tag{17}$$

where $B_k := \gamma [P^{\pi_k} - P^{c\mu_k}](I - \gamma P^{c\mu_k})^{-1}$. Since $P^{\pi_k} - P^{c\mu_k}$ has non-negative elements (as proven in (12)) as well as $(I - \gamma P^{c\mu_k})^{-1}$, then B_k has non-negative elements as well. Thus

$$\mathcal{T}^{\pi_k} Q_k - Q_k \geq B_{k-1} B_{k-2} \dots B_0 (\mathcal{T}^{\pi_0} Q_0 - Q_0) \geq 0,$$

since we assumed $T^{\pi_0} Q_0 - Q_0 \geq 0$. Thus (16) implies that

$$Q_{k+1} - Q^* \quad \geq \quad \gamma P^{\pi^*} (Q_k - Q^*) - \varepsilon_k \|Q_k\|.$$

and combining the above with (14) we deduce

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

Now assume that $\varepsilon_k \to 0$. We first deduce that Q_k is bounded. Indeed as soon as $\varepsilon_k < (1-\gamma)/2$, we have

$$\|Q_{k+1}\| \leq \|Q^*\| + \gamma \|Q_k - Q^*\| + \frac{1-\gamma}{2} \|Q_k\| \leq (1+\gamma) \|Q^*\| + \frac{1+\gamma}{2} \|Q_k\|.$$

Thus $\limsup \|Q_k\| \leq \frac{1+\gamma}{1-(1+\gamma)/2} \|Q^*\|$. Since Q_k is bounded, we deduce that $\limsup Q_k = Q^*$. $\square$

D Proof of Theorem 3

We first prove convergence of the general online algorithm.

Theorem 4. *Consider the algorithm*

$$Q_{k+1}(x,a) = (1-\alpha_k(x,a))Q_k(x,a) + \alpha_k(x,a)(\mathcal{R}_k Q_k(x,a) + \omega_k(x,a) + \upsilon_k(x,a)), \quad (18)$$

and assume that (1) ω_k is a centered, $\mathcal{F}_k$-measurable noise term of bounded variance, and (2) υ_k is bounded from above by $\theta_k(\|Q_k\|+1)$, where (θ_k) is a random sequence that converges to 0 a.s. Then, under the same assumptions as in Theorem 3, we have that $Q_k \to Q^*$ almost surely.

Proof. We write $\mathcal{R}$ for $\mathcal{R}_k$. Let us prove the result in three steps.

Upper bound on $\mathcal{R}Q_k - Q^*$. The first part of the proof is similar to the proof of (14), so we have

$$\mathcal{R}Q_k - Q^* \le \gamma\|Q_k - Q^*\|e. \quad (19)$$

Lower bound on $\mathcal{R}Q_k - Q^*$. Again, similarly to (16) we have

$$\begin{aligned}\mathcal{R}Q_k - Q^* \quad &\ge \quad \gamma\lambda P^{\pi_k\wedge\mu_k}(I-\gamma\lambda P^{\pi_k\wedge\mu_k})^{-1}(\mathcal{T}^{\pi_k}Q_k - Q_k)\\ &\quad\quad +\gamma P^{\pi^*}(Q_k - Q^*) - \varepsilon_k\|Q_k\|. \end{aligned}\quad (20)$$

Lower-bound on $\mathcal{T}^{\pi_k}Q_k - Q_k$. Since the sequence of policies (π_k) is increasingly greedy w.r.t. (Q_k), we have

$$\begin{aligned}\mathcal{T}^{\pi_{k+1}}Q_{k+1} - Q_{k+1} \quad &\ge \quad \mathcal{T}^{\pi_k}Q_{k+1} - Q_{k+1}\\ &= \quad (1-\alpha_k)\mathcal{T}^{\pi_k}Q_k + \alpha_k\mathcal{T}^{\pi_k}(\mathcal{R}Q_k + \omega_k + \upsilon_k) - Q_{k+1}\\ &= \quad (1-\alpha_k)(\mathcal{T}^{\pi_k}Q_k - Q_k) + \alpha_k\big[\mathcal{T}^{\pi_k}\mathcal{R}Q_k - \mathcal{R}Q_k + \omega'_k + \upsilon'_k\big], \end{aligned}\quad (21)$$

where $\omega'_k := (\gamma P^{\pi_k} - I)\omega_k$ and $\upsilon'_k := (\gamma P^{\pi_k} - I)\upsilon_k$. It is easy to see that both ω'_k and υ'_k continue to satisfy the assumptions on ω_k, and υ_k. Now, from the definition of the $\mathcal{R}$ operator, we have

$$\begin{aligned}\mathcal{T}^{\pi_k}\mathcal{R}Q_k - \mathcal{R}Q_k &= r + (\gamma P^{\pi_k} - I)\mathcal{R}Q_k\\ &= r + (\gamma P^{\pi_k} - I)\big[Q_k + (I-\gamma\lambda P^{\pi_k\wedge\mu_k})^{-1}(\mathcal{T}^{\pi_k}Q_k - Q_k)\big]\\ &= \mathcal{T}^{\pi_k}Q_k - Q_k + (\gamma P^{\pi_k} - I)(I-\gamma\lambda P^{\pi_k\wedge\mu_k})^{-1}(\mathcal{T}^{\pi_k}Q_k - Q_k)\\ &= \gamma(P^{\pi_k} - \lambda P^{\pi_k\wedge\mu_k})(I-\gamma\lambda P^{\pi_k\wedge\mu_k})^{-1}(\mathcal{T}^{\pi_k}Q_k - Q_k). \end{aligned}$$

Using this equality into (21) and writing $\xi_k := \mathcal{T}^{\pi_k}Q_k - Q_k$, we have

$$\xi_{k+1} \ge (1-\alpha_k)\xi_k + \alpha_k\big[B_k\xi_k + \omega'_k + \upsilon'_k\big], \quad (22)$$

where $B_k := \gamma(P^{\pi_k} - \lambda P^{\pi_k\wedge\mu_k})(I-\gamma\lambda P^{\pi_k\wedge\mu_k})^{-1}$. The matrix B_k is non-negative but may not be a contraction mapping (the sum of its components per row may be larger than 1). Thus we cannot directly apply Proposition 4.5 of Bertsekas and Tsitsiklis (1996). However, as we have seen in the proof of Theorem 2, the matrix $A_k := \gamma(I-\gamma\lambda P^{\pi_k\wedge\mu_k})^{-1}(P^{\pi_k} - \lambda P^{\pi_k\wedge\mu_k})$ is a γ-contraction mapping. So now we relate B_k to A_k using our assumption that P^{π_k} and $P^{\pi_k\wedge\mu_k}$ commute asymptotically, i.e. $\|P^{\pi_k}P^{\pi_k\wedge\mu_k} - P^{\pi_k\wedge\mu_k}P^{\pi_k}\| = \eta_k$ with $\eta_k \to 0$. For any (sub)-transition matrices U and V, we have

$$\begin{aligned}U(I-\lambda\gamma V)^{-1} \quad &= \quad \sum_{t\ge0}(\lambda\gamma)^t UV^t\\ &= \quad \sum_{t\ge0}(\lambda\gamma)^t\Big[\sum_{s=0}^{t-1}V^s(UV-VU)V^{t-s-1} + V^tU\Big]\\ &= \quad (I-\lambda\gamma V)^{-1}U + \sum_{t\ge0}(\lambda\gamma)^t\sum_{s=0}^{t-1}V^s(UV-VU)V^{t-s-1}. \end{aligned}$$

Replacing U by P^{π_k} and V by $P^{\pi_k\wedge\mu_k}$, we deduce

$$\|B_k - A_k\| \le \gamma\sum_{t\ge0}t(\lambda\gamma)^t\eta_k = \gamma\frac{1}{(1-\lambda\gamma)^2}\eta_k.$$

Thus, from (22),

$$\xi_{k+1} \geq (1-\alpha_k)\xi_k + \alpha_k\big[A_k\xi_k + \omega'_k + \upsilon''_k\big], \tag{23}$$

where $\upsilon''_k := \upsilon'_k + \gamma\sum_{t\geq 0} t(\lambda\gamma)^t \eta_k \|\xi_k\|$ continues to satisfy the assumptions on υ_k (since $\eta_k \to 0$).

Now, let us define another sequence ξ'_k as follows: $\xi'_0 = \xi_0$ and

$$\xi'_{k+1} = (1-\alpha_k)\xi'_k + \alpha_k(A_k\xi'_k + \omega'_k + \upsilon''_k).$$

We can now apply Proposition 4.5 of Bertsekas and Tsitsiklis (1996) to the sequence (ξ'_k). The matrices A_k are non-negative, and the sum of their coefficients per row is bounded by γ, see (13), thus A_k are γ-contraction mappings and have the same fixed point which is 0. The noise ω'_k is centered and $\mathcal{F}_k$-measurable and satisfies the bounded variance assumption, and υ''_k is bounded above by $(1+\gamma)\theta'_k(\|Q_k\|+1)$ for some $\theta'_k \to 0$. Thus $\lim_k \xi'_k = 0$ almost surely.

Now, it is straightforward to see that $\xi_k \geq \xi'_k$ for all $k \geq 0$. Indeed by induction, let us assume that $\xi_k \geq \xi'_k$. Then

$$\begin{aligned}
\xi_{k+1} &\geq (1-\alpha_k)\xi_k + \alpha_k(A_k\xi_k + \omega'_k + \upsilon''_k)\\
&\geq (1-\alpha_k)\xi'_k + \alpha_k(A_k\xi'_k + \omega'_k + \upsilon''_k)\\
&= \xi'_{k+1},
\end{aligned}$$

since all elements of the matrix A_k are non-negative. Thus we deduce that

$$\liminf_{k\to\infty} \xi_k \geq \lim_{k\to\infty} \xi'_k = 0 \tag{24}$$

Conclusion. Using (24) in (20) we deduce the lower bound:

$$\liminf_{k\to\infty} \mathcal{R}Q_k - Q^* \geq \liminf_{k\to\infty} \gamma P^{\pi^*}(Q_k - Q^*), \tag{25}$$

almost surely. Now combining with the upper bound (19) we deduce that

$$\|\mathcal{R}Q_k - Q^*\| \leq \gamma\|Q_k - Q^*\| + O(\varepsilon_k\|Q_k\|) + O(\xi_k).$$

The last two terms can be incorporated to the $\upsilon_k(x,a)$ and $\omega_k(x,a)$ terms, respectively; we thus again apply Proposition 4.5 of Bertsekas and Tsitsiklis (1996) to the sequence (Q_k) defined by (18) and deduce that $Q_k \to Q^*$ almost surely. $\square$

It remains to rewrite the update (8) in the form of (18), in order to apply Theorem 4.

Let $z^k_{s,t}$ denote the accumulating trace (Sutton and Barto, 1998):

$$z^k_{s,t} := \sum_{j=s}^{t} \gamma^{t-j}\Big(\prod_{i=j+1}^{t} c_i\Big)\mathbb{I}\{(x_j,a_j) = (x_s,a_s)\}.$$

Let us write $Q^o_{k+1}(x_s,a_s)$ to emphasize the online setting. Then (8) can be written as

$$Q^o_{k+1}(x_s,a_s) \leftarrow Q^o_k(x_s,a_s) + \alpha_k(x_s,a_s)\sum_{t\geq s}\delta^{\pi_k}_t z^k_{s,t}, \tag{26}$$

$$\delta^{\pi_k}_t := r_t + \gamma\mathbb{E}_{\pi_k}Q^o_k(x_{t+1},\cdot) - Q^o_k(x_t,a_t),$$

Using our assumptions on finite trajectories, and $c_i \leq 1$, we can show that:

$$\mathbb{E}\Big[\sum_{t\geq s} z^k_{s,t}|\mathcal{F}_k\Big] < \mathbb{E}\left[T_k^2|\mathcal{F}_k\right] < \infty \tag{27}$$

where T_k denotes trajectory length. Now, let $D_k := D_k(x_s,a_s) := \sum_{t\geq s}\mathbb{P}\{(x_t,a_t) = (x_s,a_s)\}$. Then, using (27), we can show that the total update is bounded, and rewrite

$$\mathbb{E}_{\mu_k}\Big[\sum_{t\geq s}\delta^{\pi_k}_t z^k_{s,t}\Big] = D_k(x_s,a_s)\big(\mathcal{R}_kQ_k(x_s,a_s) - Q(x_s,a_s)\big).$$

Finally, using the above, and writing $\alpha_k = \alpha_k(x_s, a_s)$, (26) can be rewritten in the desired form:

$$
\begin{aligned}
Q^o_{k+1}(x_s, a_s) &\leftarrow (1-\tilde{\alpha}_k) Q^o_k(x_s, a_s) + \tilde{\alpha}_k \big(\mathcal{R}_k Q^o_k(x_s, a_s) + \omega_k(x_s, a_s) + \upsilon_k(x_s, a_s)\big), \qquad (28)\\
\omega_k(x_s, a_s) &:= (D_k)^{-1} \left(\sum_{t \ge s} \delta_t^{\pi_k} z^k_{s,t} - \mathbb{E}_{\mu_k} \left[\sum_{t \ge s} \delta_t^{\pi_k} z^k_{s,t} \right] \right),\\
\upsilon_k(x_s, a_s) &:= (\tilde{\alpha}_k)^{-1} \big(Q^o_{k+1}(x_s, a_s) - Q_{k+1}(x_s, a_s) \big),\\
\tilde{\alpha}_k &:= \alpha_k D_k.
\end{aligned}
$$

It can be shown that the variance of the noise term ω_k is bounded, using (27) and the fact that the reward function is bounded. It follows from Assumptions 1-3 that the modified stepsize sequence $(\tilde{\alpha}_k)$ satisfies the conditions of Assumption 1. The second noise term $\upsilon_k(x_s, a_s)$ measures the difference between online iterates and the corresponding offline values, and can be shown to satisfy the required assumption analogously to the argument in the proof of Prop. 5.2 in Bertsekas and Tsitsiklis (1996). The proof relies on the eligibility coefficients (27) and rewards being bounded, the trajectories being finite, and the conditions on the stepsizes being satisfied.

We can thus apply Theorem 4 to (28), and conclude that the iterates $Q^o_k \to Q^*$ as $k \to \infty$, w.p. 1.

E Asymptotic commutativity of P^{π_k} and $P^{\pi_k \wedge \mu_k}$

Lemma 4. *Let (π_k) and (μ_k) two sequences of policies. If there exists α such that for all x, a,*

$$\min(\pi_k(a|x), \mu_k(a|x)) = \alpha \pi_k(a|x) + o(1), \qquad (29)$$

then the transition matrices P^{π_k} and $P^{\pi_k \wedge \mu_k}$ asymptotically commute: $\|P^{\pi_k} P^{\pi_k \wedge \mu_k} - P^{\pi_k \wedge \mu_k} P^{\pi_k}\| = o(1)$.

Proof. For any Q, we have

$$
\begin{aligned}
(P^{\pi_k} P^{\pi_k \wedge \mu_k}) Q(x,a) &= \sum_y p(y|x,a) \sum_b \pi_k(b|y) \sum_z p(z|y,b) \sum_c (\pi_k \wedge \mu_k)(c|z) Q(z,c)\\
&= \alpha \sum_y p(y|x,a) \sum_b \pi_k(b|y) \sum_z p(z|y,b) \sum_c \pi_k(c|z) Q(z,c) + \|Q\| o(1)\\
&= \sum_y p(y|x,a) \sum_b (\pi_k \wedge \mu_k)(b|y) \sum_z p(z|y,b) \sum_c \pi_k(c|z) Q(z,c) + \|Q\| o(1)\\
&= (P^{\pi_k \wedge \mu_k} P^{\pi_k}) Q(x,a) + \|Q\| o(1).
\end{aligned}
$$

□

Lemma 5. *Let (π_{Q_k}) a sequence of (deterministic) greedy policies w.r.t. a sequence (Q_k). Let (π_k) a sequence of policies that are ε_k away from (π_{Q_k}), in the sense that, for all x,*

$$\|\pi_k(\cdot|x) - \pi_{Q_k}(x)\|_1 := 1 - \pi_k(\pi_{Q_k}(x)|x) + \sum_{a \neq \pi_{Q_k}(x)} \pi_k(a|x) \le \varepsilon_k.$$

Let (μ_k) a sequence of policies defined by:

$$\mu_k(a|x) = \frac{\alpha \mu(a|x)}{1 - \mu(\pi_{Q_k}(x)|x)} \mathbb{I}\{a \neq \pi_{Q_k}(x)\} + (1-\alpha)\mathbb{I}\{a = \pi_{Q_k}(x)\}, \qquad (30)$$

for some arbitrary policy μ and $\alpha \in [0,1]$. Assume $\varepsilon_k \to 0$. Then the transition matrices P^{π_k} and $P^{\pi_k \wedge \mu_k}$ asymptotically commute.

Proof. The intuition is that asymptotically π_k gets very close to the deterministic policy π_{Q_k}. In that case, the minimum distribution $(\pi_k \wedge \mu_k)(\cdot|x)$ puts a mass close to $1-\alpha$ on the greedy action $\pi_{Q_k}(x)$, and no mass on other actions, thus $(\pi_k \wedge \mu_k)$ gets very close to $(1-\alpha)\pi_k$, and Lemma 4 applies (with multiplicative constant $1-\alpha$).

Indeed, from our assumption that π_k is ε-away from π_{Q_k} we have:

$$\pi_k(\pi_{Q_k}(x)|x) \ge 1 - \varepsilon_k, \text{ and } \pi_k(a \neq \pi_{Q_k}(x)|x) \le \varepsilon_k.$$

We deduce that

$$\begin{aligned}(\pi_k \wedge \mu_k)(\pi_{Q_k}(x)|x) &= \min(\pi_k(\pi_{Q_k}(x)|x), 1-\alpha) \\ &= 1-\alpha+O(\varepsilon_k) \\ &= (1-\alpha)\pi_k(\pi_{Q_k}(x)|x)+O(\varepsilon_k),\end{aligned}$$

and

$$\begin{aligned}(\pi_k \wedge \mu_k)(a \neq \pi_{Q_k}(x)|x) &= O(\varepsilon_k) \\ &= (1-\alpha)\pi_k(a|x)+O(\varepsilon_k).\end{aligned}$$

Thus Lemma 4 applies (with a multiplicative constant $1-\alpha$) and P^{π_k} and $P^{\pi_k \wedge \mu_k}$ asymptotically commute. □

F Experimental Methods

Although our experiments' learning problem closely matches the DQN setting used by Mnih et al. (2015) (i.e. single-thread off-policy learning with large replay memory), we conducted our trials in the multi-threaded, CPU-based framework of Mnih et al. (2016), obtaining ample result data from affordable CPU resources. Key differences from the DQN are as follows. Sixteen threads with private environment instances train simultaneously; each infers with and finds gradients w.r.t. a local copy of the network parameters; gradients then update a "master" parameter set and local copies are refreshed. Target network parameters are simply shared globally. Each thread has private replay memory holding 62,500 transitions (1/16$^{\text{th}}$ of DQN's total replay capacity). The optimiser is unchanged from (Mnih et al., 2016): "Shared RMSprop" with step size annealing to 0 over 3×10^8 environment frames (summed over threads). Exploration parameter (ε) behaviour differs slightly: every 50,000 frames, threads switch randomly (probability 0.3, 0.4, and 0.3 respectively) between three schedules (anneal ε from 1 to 0.5, 0.1, or 0.01 over 250,000 frames), starting new schedules at the intermediate positions where they left old ones.[1]

Our experiments comprise 60 Atari 2600 games in ALE (Bellemare et al., 2013), with "life" loss treated as episode termination. The control, minibatched (64 transitions/minibatch) one-step Q-learning as in (Mnih et al., 2015), shows performance comparable to DQN in our multi-threaded setup. Retrace, TB, and Q* runs use minibatches of four 16-step sequences (again 64 transitions/minibatch) and the current exploration policy as the target policy π. All trials clamp rewards into $[-1, 1]$. In the control, Q-function targets are clamped into $[-1, 1]$ prior to gradient calculation; analogous quantities in the multi-step algorithms are clamped into $[-1, 1]$, then scaled (divided by) the sequence length. Coarse, then fine logarithmic parameter sweeps on the games *Asterix*, *Breakout*, *Enduro*, *Freeway*, *H.E.R.O*, *Pong*, *Q*bert*, and *Seaquest* yielded step sizes of 0.0000439 and 0.0000912, and RMSprop regularisation parameters of 0.001 and 0.0000368, for control and multi-step algorithms respectively. Reported performance averages over four trials with different random seeds for each experimental configuration.

[1]We evaluated a DQN-style single schedule for ε, but our multi-schedule method, similar to the one used by Mnih et al., yielded improved performance in our multi-threaded setting.

	Tree-backup(λ)	Retrace(λ)	DQN	Q*(λ)
Alien	2508.62	**3109.21**	2088.81	154.35
Amidar	1221.00	**1247.84**	772.30	16.04
Assault	7248.08	**8214.76**	1647.25	260.95
Asterix	**29294.76**	28116.39	10675.57	285.44
Asteroids	1499.82	**1538.25**	1403.19	308.70
Atlantis	**2115949.75**	2110401.90	1712671.88	3667.18
Bank Heist	**808.31**	797.36	549.35	1.70
Battle Zone	22197.96	**23544.08**	21700.01	3278.93
Beam Rider	15931.60	**17281.24**	8053.26	621.40
Berzerk	967.29	**972.67**	627.53	247.80
Bowling	40.96	**47.92**	37.82	15.16
Boxing	91.00	93.54	**95.17**	-29.25
Breakout	288.71	298.75	**332.67**	1.21
Carnival	**4691.73**	4633.77	4637.86	353.10
Centipede	1199.46	1715.95	1037.95	**3783.60**
Chopper Command	6193.28	**6358.81**	5007.32	534.83
Crazy Climber	**115345.95**	114991.29	111918.64	1136.21
Defender	32411.77	**33146.83**	13349.26	1838.76
Demon Attack	68148.22	**79954.88**	8585.17	310.45
Double Dunk	**-1.32**	-6.78	-5.74	-23.63
Elevator Action	1544.91	2396.05	**14607.10**	930.38
Enduro	1115.00	**1216.47**	938.36	12.54
Fishing Derby	22.22	**27.69**	15.14	-98.58
Freeway	**32.13**	32.13	31.07	9.86
Frostbite	960.30	935.42	**1124.60**	45.07
Gopher	13666.33	**14110.94**	11542.46	50.59
Gravitar	30.18	29.04	**271.40**	13.14
H.E.R.O.	**25048.33**	21989.46	17626.90	12.48
Ice Hockey	**-3.84**	-5.08	-4.36	-15.68
James Bond	560.88	641.51	**705.55**	21.71
Kangaroo	11755.01	**11896.25**	4101.92	178.23
Krull	**9509.83**	9485.39	7728.66	429.26
Kung-Fu Master	25338.05	**26695.19**	17751.73	39.99
Montezuma's Revenge	**0.79**	0.18	0.10	0.00
Ms. Pac-Man	2461.10	**3208.03**	2654.97	298.58
Name This Game	**11358.81**	11160.15	10098.85	1311.73
Phoenix	13834.27	**15637.88**	9249.38	107.41
Pitfall	**-37.74**	-43.85	-392.63	-121.99
Pooyan	5283.69	**5661.92**	3301.69	98.65
Pong	**20.25**	20.20	19.31	-20.99
Private Eye	73.44	**87.36**	44.73	-147.49
Q*Bert	13617.24	**13700.25**	12412.85	114.84
River Raid	14457.29	**15365.61**	10329.58	922.13
Road Runner	34396.52	32843.09	**50523.75**	418.62
Robotank	36.07	41.18	**49.20**	5.77
Seaquest	3557.09	2914.00	**3869.30**	175.29
Skiing	-25055.94	-25235.75	-25254.43	**-24179.71**
Solaris	1178.05	1135.51	**1258.02**	674.58
Space Invaders	**6096.21**	5623.34	2115.80	227.39
Star Gunner	66369.18	**74016.10**	42179.52	266.15
Surround	**-5.48**	-6.04	-8.17	-9.98
Tennis	-1.73	-0.30	**13.67**	-7.37
Time Pilot	8266.79	**8719.19**	8228.89	657.59
Tutankham	164.54	**199.25**	167.22	2.68
Up and Down	14976.51	**18747.40**	9404.95	530.59
Venture	10.75	22.84	**30.93**	0.09
Video Pinball	103486.09	**228283.79**	76691.75	6837.86
Wizard Of Wor	7402.99	**8048.72**	612.52	189.43
Yar's Revenge	14581.65	**26860.57**	15484.03	1913.19
Zaxxon	12529.22	**15383.11**	8422.49	0.40
Times Best	16	30	12	2

Table 2: Final scores achieved by the different λ-return variants ($\lambda = 1$). Highlights indicate high scores.