This is a digitally remastered version of a paper published in the *Proceedings of the 18th International Conference on Machine Learning* (2001), pp. 417–424. This version should be crisper but otherwise identical to the published article.

Off-Policy Temporal-Difference Learning with Function Approximation

Doina Precup DPRECUP@CS.MCGILL.CA
School of Computer Science, McGill University, Montreal, Quebec, Canada H3A 2A7

Richard S. Sutton SUTTON@RESEARCH.ATT.COM
Sanjoy Dasgupta DASGUPTA@RESEARCH.ATT.COM
AT&T Shannon Laboratory, 180 Park Ave., Florham Park, NJ 07932 USA

Abstract

We introduce the first algorithm for off-policy temporal-difference learning that is stable with linear function approximation. Off-policy learning is of interest because it forms the basis for popular reinforcement learning methods such as Q-learning, which has been known to diverge with linear function approximation, and because it is critical to the practical utility of multi-scale, multi-goal, learning frameworks such as options, HAMs, and MAXQ. Our new algorithm combines TD(λ) over state–action pairs with importance sampling ideas from our previous work. We prove that, given training under any ϵ-soft policy, the algorithm converges w.p.1 to a close approximation (as in Tsitsiklis and Van Roy, 1997; Tadic, 2001) to the action-value function for an arbitrary target policy. Variations of the algorithm designed to reduce variance introduce additional bias but are also guaranteed convergent. We also illustrate our method empirically on a small policy evaluation problem. Our current results are limited to episodic tasks with episodes of bounded length.

Although Q-learning remains the most popular of all reinforcement learning algorithms, it has been known since about 1996 that it is unsound with linear function approximation (see Gordon, 1995; Bertsekas and Tsitsiklis, 1996). The most telling counterexample, due to Baird (1995) is a seven-state Markov decision process with linearly independent feature vectors, for which an exact solution exists, yet for which the approximate values found by Q-learning diverge to infinity. This problem prompted the development of residual gradient methods (Baird, 1995), which are stable but much slower than Q-learning, and fitted value iteration (Gordon, 1995, 1999), which is also stable but limited to restricted, weaker-than-linear function approximators. Of course, Q-learning has been used with linear function approximation since its invention (Watkins, 1989), often with good results, but the soundness of this approach is no longer an open question. There exist non-pathological Markov decision processes for which it diverges; it is absolutely unsound in this sense.

A sensible response is to turn to some of the other reinforcement learning methods, such as Sarsa, that are also efficient and for which soundness remains a possibility. An important distinction here is between methods that must follow the policy they are learning about, called *on-policy* methods, and those that can learn from behavior generated by a different policy, called *off-policy* methods. Q-learning is an off-policy method in that it learns the optimal policy even when actions are selected according to a more exploratory or even random policy. Q-learning requires only that all actions be tried in all states, whereas on-policy methods like Sarsa require that they be selected with specific probabilities.

Although the off-policy capability of Q-learning is appealing, it is also the source of at least part of its instability problems. For example, in one version of Baird's counterexample, the TD(λ) algorithm, which underlies both Q-learning and Sarsa, is applied with linear function approximation to learn the action-value function Q^π for a given policy π. Operating in an on-policy mode, updating state–action pairs according to the same distribution with which they would be experienced under π, this method is stable and convergent near the best possible solution (Tsitsiklis and Van

Roy, 1997; Tadic, 2001). However, if state-action pairs are updated according to a different distribution, say that generated by following the greedy policy, then the estimated values again diverge to infinity. This and related counterexamples suggest that at least some of the reason for the instability of Q-learning is that it is an off-policy method; they also make it clear that this part of the problem can be studied in a purely policy-evaluation context.

Despite these problems, there remains substantial reason for interest in off-policy learning methods. Several researchers have argued for an ambitious extension of reinforcement learning ideas into modular, multi-scale, and hierarchical architectures (Sutton, Precup & Singh, 1999; Parr, 1998; Parr & Russell, 1998; Dietterich, 2000). These architectures rely on off-policy learning to learn about multiple subgoals and multiple ways of behaving from the singular stream of experience. For these approaches to be feasible, some efficient way of combining off-policy learning and function approximation must be found.

Because the problems with current off-policy methods become apparent in a policy evaluation setting, it is there that we focus in this paper. In previous work we considered multi-step off-policy policy evaluation in the tabular case. In this paper we introduce the first off-policy policy evaluation method consistent with linear function approximation. Our mathematical development focuses on the episodic case, and in fact on a single episode. Given a starting state and action, we show that the expected *off*-policy update under our algorithm is the same as the expected *on*-policy update under conventional TD(λ). This, together with some variance conditions, allows us to prove convergence and bounds on the error in the asymptotic approximation identical to those obtained by Tsitsiklis and Van Roy (1997; Bertsekas and Tsitsiklis, 1996).

1. Notation and Main Result

We consider the standard episodic reinforcement learning framework (see, e.g., Sutton & Barto, 1998) in which a learning agent interacts with a Markov decision process (MDP). Our notation focuses on a single episode of T time steps, $s_0, a_0, r_1, s_1, a_1, r_2, \ldots, r_T, s_T$, with states $s_t \in \mathcal{S}$, actions $a_t \in \mathcal{A}$, and rewards $r_t \in \Re$. We take the initial state and action, s_0 and a_0, to be given arbitrarily. Given a state and action, s_t and a_t, the next reward, r_{t+1}, is a random variable with mean $r_{s_t}^{a_t}$ and the next state, s_{t+1}, is chosen with probabilities $p_{s_t s_{t+1}}^{a_t}$. The final state is a special terminal state that may not occur on any preceding time step.

Given a state, s_t, $0 < t < T$, the action a_t is selected according to probability $\pi(s_t, a_t)$ or $b(s_t, a_t)$ depending on whether policy π or policy b is in force. We always use π to denote the *target policy*, the policy that we are learning about. In the on-policy case, π is also used to generate the actions of the episode. In the off-policy case, the actions are instead generated by b, which we call the *behavior policy*.

In either case, we seek an approximation to the action-value function $Q^\pi : \mathcal{S} \times \mathcal{A} \mapsto \Re$ for the target policy π:

$$Q^\pi(s,a) = E_\pi\big\{r_{t+1} + \cdots + \gamma^{T-1} r_T \mid s_t = s, a_t = a\big\},$$

where $0 \leq \gamma \leq 1$ is a discount-rate parameter. We consider approximations that are linear in a set of feature vectors $\{\phi_{sa}\}, s \in \mathcal{S}, a \in \mathcal{A}$:

$$Q^\pi(s,a) \approx \theta^T \phi_{sa} = \sum_{i=1}^{n} \theta(i)\phi_{sa}(i),$$

where $\theta \in \Re^n$ is the learned parameter vector. The feature vector for the special terminal state is assumed to be the zero vector so that the estimated value for this state is (correctly) zero.

In this paper we restrict our attention to per-episode updating, meaning that although an increment to θ is computed on each step, θ is not actually updated until the end of the episode (by a total increment, $\Delta\theta$, equal to the sum of the increments on each step). The increments for conventional TD(λ) under per-episode updating are given by the forward-view equations:

$$\Delta\theta_t = \alpha\left(R_t^\lambda - \theta^T\phi_t\right)\phi_t,$$

$$R_t^\lambda = (1-\lambda)\sum_{n=1}^{\infty} \lambda^{n-1} R_t^{(n)},$$

$$R_t^{(n)} = r_{t+1} + \gamma r_{t+2} + \cdots + \gamma^{n-1} r_{t+n} + \gamma^n \theta^T \phi_{t+n},$$

where ϕ_t is a shorthand for $\phi_{s_t a_t}$, $\phi_t = 0$ for $t \geq T$, and $r_t = 0$ for $t > T$. $R_t^{(n)}$ is called the n-step return and R_t^λ is called the λ-return. This forward view can also be implemented incrementally using eligibility traces and a backward view (Sutton & Barto, 1998).

The most straightforward way to introduce importance sampling into linear TD(λ) is to multiply the increments for each episode by the relative probability of that episode occuring under the target and behavior policies. If we define the importance sampling ratio for time t as $\rho_t = \frac{\pi(s_t,a_t)}{b(s_t,a_t)}$, then this relative probability is $\rho_1\rho_2\cdots\rho_{T-1}$. Let us call this the *naive* importance

sampling algorithm. Our algorithm instead multiplies only by the first t importance sampling ratios:

$$\Delta\theta_t = \alpha\left(\bar{R}_t^\lambda - \theta^T\phi_t\right)\phi_t\rho_1\rho_2\cdots\rho_t, \qquad (1)$$

where $\bar{R}_t^\lambda$ is defined as R_t^λ above, except in terms of the off-policy n-step return:

$$\begin{aligned}\bar{R}_t^{(n)} &= r_{t+1} + \gamma r_{t+2}\rho_{t+1} + \cdots \\ &\quad + \gamma^{n-1} r_{t+n}\rho_{t+1}\cdots\rho_{t+n-1} \\ &\quad + \gamma^n \rho_{t+1}\cdots\rho_{t+n}\theta^T\phi_{t+n}\end{aligned}$$

The off-policy n-step return was introduced by Precup, Sutton and Singh (2000) as part of their *per-decision* importance sampling algorithm. They showed that the importance sampling ratios correct for off-policy training such that

$$E_b\{\bar{R}_t^\lambda \mid s_t, a_t\} = E_\pi\{R_t^\lambda \mid s_t, a_t\},$$

where the subscripts on the expectations indicate the policy in force (i.e., they indicate either off-policy training, b, or on-policy training, π). Here we extend this idea to the case of linear function approximation by including the correction ratios in (1). We are now ready to state our main result:

Theorem 1 *Let $\Delta\theta$ and $\Delta\bar{\theta}$ be the sum of the parameter increments over an episode under on-policy TD(λ) and importance sampled TD(λ) respectively, assuming that the starting weight vector is θ in both cases. Then*

$$E_b\{\Delta\bar{\theta} \mid s_0, a_0\} = E_\pi\{\Delta\theta \mid s_0, a_0\}, \quad \forall s_0 \in \mathcal{S}, a_0 \in \mathcal{A}.$$

Proof: To simplify the notation, we henceforth take it as implicit that expectations are conditioned on s_0, a_0. Then

$$E_b\{\Delta\bar{\theta}\} = E_b\left\{\sum_{t=0}^{\infty}\alpha\left(\bar{R}_t^\lambda - \theta^T\phi_t\right)\phi_t\rho_1\rho_2\cdots\rho_t\right\}$$

$$= E_b\left\{\sum_{t=0}^{\infty}\sum_{n=1}^{\infty}\alpha(1-\lambda)\lambda^{n-1}(\bar{R}_t^{(n)} - \theta^T\phi_t)\phi_t\rho_1\rho_2\cdots\rho_t\right\}.$$

It suffices to show that this is the same as in on-policy TD(λ), i.e., that, for any n,

$$E_b\left\{\sum_{t=0}^{\infty}\left(\bar{R}_t^{(n)} - \theta^T\phi_t\right)\phi_t\rho_1\rho_2\cdots\rho_t\right\} = E_\pi\left\{\sum_{t=0}^{\infty}\left(R_t^{(n)} - \theta^T\phi_t\right)\phi_t\right\}.$$

Let Ω_t denote the set of all possible trajectories of state-action pairs starting with s_0, a_0 and going through time t. Let ω denote one such trajectory and $p_b(\omega)$ its probability of occurring under policy b. Then

$$\begin{aligned}&E_b\left\{\sum_{t=0}^{\infty}\left(\bar{R}_t^{(n)} - \theta^T\phi_t\right)\phi_t\rho_1\rho_2\cdots\rho_t\right\} \\ &= \sum_{t=0}^{\infty}\sum_{\omega\in\Omega_t} p_b(\omega)\phi_t\prod_{k=1}^{t}\rho_k E_b\left\{\bar{R}_t^{(n)} - \theta^T\phi_t \;\middle|\; s_t, a_t\right\} \\ &\quad \text{(given the Markov property)} \\ &= \sum_{t=0}^{\infty}\sum_{\omega\in\Omega_t}\prod_{j=1}^{t} p_{s_{j-1},s_j}^{a_{j-1}} b(s_j,a_j)\phi_t\prod_{k=1}^{t}\frac{\pi(s_k,a_k)}{b(s_k,a_k)} \\ &\quad \cdot \left(E_b\left\{\bar{R}_t^{(n)} \;\middle|\; s_t, a_t\right\} - \theta^T\phi_t\right) \\ &= \sum_{t=0}^{\infty}\sum_{\omega\in\Omega_t}\prod_{j=1}^{t} p_{s_{j-1},s_j}^{a_{j-1}}\pi(s_j,a_j)\phi_t \\ &\quad \cdot \left(E_b\left\{\bar{R}_t^{(n)} \;\middle|\; s_t, a_t\right\} - \theta^T\phi_t\right) \\ &= \sum_{t=0}^{\infty}\sum_{\omega\in\Omega_t} p_\pi(\omega)\phi_t\left(E_\pi\left\{R^{(n)} \;\middle|\; s_t, a_t\right\} - \theta^T\phi_t\right) \\ &\quad \text{(using our previous result)} \\ &= E_\pi\left\{\sum_{t=0}^{\infty}\left(R_t^{(n)} - \theta^T\phi_t\right)\phi_t\right\}. \diamond\end{aligned}$$

2. Convergence and Error Bounds

Given Theorem 1, we can apply the analysis of Tsitskilis and Van Roy to prove convergence and error bounds. Their paper (Tsitsiklis & Van Roy, 1997) treated the discounted continuing (ergodic) case, whereas here we consider the episodic case. Their results for this case were published in the textbook by Bertsekas and Tsitsiklis (1996). Gurvitz also obtained similar results, and some of the ideas can be traced back to his work (Gurvitz, Lin & Hanson, unpublished). Tadic (2001) proved a similar result using different mathematical techniques, and a less restrictive set of assumptions.

Let $d : \mathcal{S}\times\mathcal{A} \mapsto [0,1], \sum_{s,a} d(s,a) = 1$ be the (arbitrary) distribution of starting state–action pairs. Let P_π be the state–action pair to state–action pair transition-probability matrix for policy π. Let $D_\pi = \sum_{t=0}^{\infty} P_\pi^t d$ denote the vector in which $D_\pi(s,a)$ is the expected number of visits to state–action pair s, a in an episode started according to d. Define the norm $||\cdot||_\pi$ over state–action-pair vectors by $||v||_\pi^2 = \sum_{s,a} v(s,a)D_\pi(s,a)v(s,a)$.

We require a number of natural assumptions: (1) the state and action sets are finite; (2) all state–action

pairs are visited under the behavior policy from d; (3) both behavior and target policies, π and b, are *proper*, meaning that $P_\pi^\infty = P_b^\infty = 0$; (4) the rewards are bounded; and (5) the step-size sequence $\{\alpha_k\}_{k=0}^\infty$ satisfies the usual stochastic approximation conditions:

$$\alpha_k \geq 0, \qquad \sum_{k=0}^{\infty} \alpha_k = \infty, \quad \text{and} \quad \sum_{k=0}^{\infty} \alpha_k^2 < \infty. \tag{2}$$

In addition, we require (6) the variance of the product of correction factors be bounded for any initial state:

$$E_b\{\rho_1^2 \rho_2^2 \rho_3^2 \cdots \rho_T^2\} < B \qquad \forall s_1 \in \mathcal{S},$$

which can be assured, for example, by simply bounding the possible episode lengths. Nevertheless, this remains a limitation of our result, as discussed further below. Finally, let Q_θ denote the approximate action–value function (vector) for any parameter value θ: $Q_\theta(s,a) = \theta^T \phi_{sa}$.

Theorem 2 *Under the assumptions 1–6 above, episodic importance sampled TD(λ) converges with probability one to some θ_∞ such that*

$$||Q_{\theta_\infty} - Q^\pi||_\pi \quad \leq \quad \min_\theta ||Q_\theta - Q^\pi||_\pi \frac{1}{1-\beta},$$

where β is the contraction factor of the matrix

$$M = (1-\lambda) \sum_{k=0}^{\infty} \lambda^k (\gamma P_\pi)^{k+1}.$$

Proof: This result is a restatement of Tsitsiklis and Van Roy's result on page 312 of Bertsekas and Tsitsiklis (1996). The assumptions together with our main result immediately satisfy the conditions of their proof. In particular, assumption 6 implies that our importance sampling corrections do not convert the usual estimator to one of unbounded variance.

The assumption of bounded variance of the correction-factor product (6) is restrictive, but not as restrictive as it might at first seem. In many cases we can assure its satisfaction by considering only "artificial" episode terminations superimposed on an original process. For example, assumption 6 is trivially met if the trial length is bounded. Even if our original MDP does not produce bounded length trials, we can consider a modified MDP that is just like the original except that all trials terminate after $T_{\max}$ steps. Sample trajectories from the original process can be used as trajectories for the modified process by truncating them after $T_{\max}$ steps. Our results assure stable convergence to a close approximation to the true evaluation function for the modified MDP and, if $T_{\max}$ is chosen large enough compared to γ or the mixing time of the original MDP, then the solutions to the original and modified MDPs will be very similar.

In our primary expected application area—learning about temporally abstract macro-actions—this kind of artificial termination is the normal way of proceeding. A macro-action consists of a target policy and a specified condition for terminating the macro-action. In this application it is not the actual process that terminates, only the macro-action. Nevertheless, the problem is formally identical to the one presented in this paper; our methods and results apply directly to learning about macro-actions. And in fact, choosing the termination process is part of designing the macro-action. Thus we can design the macro-action to have bounded variance of the correction term by terminating after $T_{\max}$ steps, for example, or whenever the correction factor becomes very large.

Thus, in many applications, the spectre of divergence due to unbounded variance can be eliminated. Nevertheless, even when bounded, high variance (and thus slow convergence) can be a major problem. In Section 6 we consider how *weighted* importance sampling methods might be adapted to reduce variance, or even remove the need for assumption 6.

3. Restarting within an Episode

The importance sampling correction product in (1) will often decay very rapidly over time, especially if the episodes are long or if the behavior and target policy are very different. Although the episode may be continuing, little more is learned once the correction factor becomes very small. In such cases one might like to pretend a new episode has started from an intermediate state of the episode. Of course, the effective starting distribution will then be different from d, which might be considered to introduce additional bias. Nevertheless, this may be desirable because of reduced variance. In this section we prove convergence of this generalized algorithm.

To formalize the idea of starting anywhere within an episode, we introduce a non-negative random variable g_t, which is allowed to depend only on events up to (and including) time t. The value g_t represents the extent to which an episode is considered to start at time t. The *function* $g : \Omega_t \mapsto \Re^+$ gives the expected value of g_t for any trajectory up through t. The forward view of the generalized algorithm is

$$\Delta\theta_t = \alpha \left(\bar{R}_t^\lambda - \theta^T \phi_t\right) \phi_t \sum_{k=0}^{t} g_k \rho_{k+1} \cdots \rho_t \tag{3}$$

Note that this algorithm is identical to the original importance sampled TD(λ) if $g_0 = 1$ and $g_t = 0, \forall t \geq 1$.

Theorem 3 *Let $\Delta\theta$ and $\Delta\bar{\theta}$ denote the sum of the parameter increments of the original importance-sampled TD(λ) (1) and the generalized version (3) respectively, Then, for any function, g, there exists an alternate starting distribution d_g such that*

$$E_b\big\{\Delta\bar{\theta} \mid s_0, a_0 \sim d\big\} = E_b\{\Delta\theta \mid s_0, a_0 \sim d_g\}.$$

Proof: To simplify notation, we allow additional subscripts on the expectations to indicate the distribution from which the initial s_0, a_0 are selected. Then

$$\begin{aligned}
&E_{b,d}\big\{\Delta\bar{\theta}\big\} \\
&= E_{b,d}\left\{\sum_{t=0}^{\infty}\sum_{k=0}^{t} g_k \prod_{j=k+1}^{t} \rho_j \alpha\left(\bar{R}_t^\lambda - \theta^T\phi_t\right)\phi_t\right\} \\
&= E_{b,d}\left\{\sum_{k=0}^{\infty}\sum_{t=k}^{\infty} g_k \prod_{j=k+1}^{t} \rho_j \alpha\left(\bar{R}_t^\lambda - \theta^T\phi_t\right)\phi_t\right\} \\
&= \sum_{k=0}^{\infty} E_{b,d}\left\{g_k \sum_{t=k}^{\infty} \prod_{j=k+1}^{t} \rho_j \alpha\left(\bar{R}_t^\lambda - \theta^T\phi_t\right)\phi_t\right\} \\
&= \sum_{k=0}^{\infty}\sum_{\omega\in\Omega_k} p_b^d(\omega) g(\omega) E_{b,s_0=s_k(\omega),a_0=a_k(\omega)}\{\Delta\theta\},
\end{aligned}$$

where Ω_k denotes the set of all trajectories ω of length k and $p_b^d(\omega)$ denotes the probability of each such trajectory occurring under b when starting from d. The final expectation above is conditional on starting in the indicated last state and action, s_k, a_k of ω. It is convenient now to define $\Omega_{k,s,a}$ as the set of all trajectories of length k ending is s, a. Then we can rewrite the above as

$$\begin{aligned}
&\sum_{k=0}^{\infty}\sum_{s,a}\sum_{\omega\in\Omega_{k,s,a}} p_b^d(\omega) g(\omega) E_{b,s_0=s,a_0=a}\{\Delta\theta\} \\
&= \sum_{s,a}\sum_{k=0}^{\infty}\sum_{\omega\in\Omega_{k,s,a}} p_b^d(\omega) g(\omega) E_{b,s_0=s,a_0=a}\{\Delta\theta\} \\
&= \sum_{s,a} E_{b,s_0=s,a_0=a}\{\Delta\theta\}\sum_{k=0}^{\infty}\sum_{\omega\in\Omega_{k,s,a}} p_b^d(\omega) g(\omega) \\
&= E_{b,d_g}\{\Delta\theta\},
\end{aligned}$$

where

$$d_g(s,a) = \sum_{k=0}^{\infty}\sum_{\omega\in\Omega_{k,s,a}} p_b^d(\omega) g(\omega)$$

is clearly a valid alternative starting distribution. $\diamond$

We have just proved that restarting in a general way, at any point during an episode, is equivalent to a conventional at-the-beginning starting distribution. The latter case we have already proved to converge; thus so must the generalized algorithm. The only difference is that the value converged to will now depend on d_g, and thus on b, rather than on d and π alone.

4. Incremental implementation

The algorithm presented in the previous section can easily be implemented in an incremental, *backward-view* fashion, using an eligibility trace vector $\vec{e}_t$ of the same dimension as θ. This implementation, which we used in the experiments that follow, is given in Figure 1.

On every episode:

1. Initialize $c_0 = g_0$, $\vec{e}_0 = c_0\phi_0$.
2. On every transition $s_t, a_t \to r_{t+1}, s_{t+1}, a_{t+1}$, for $0 \leq t < T$:

$$\begin{aligned}
\rho_{t+1} &= \pi(s_{t+1}, a_{t+1})/b(s_{t+1}, a_{t+1}) \\
\delta_t &= r_{t+1} + \gamma\,\rho_{t+1}\,\theta^T\phi_{t+1} - \theta^T\phi_t \\
\Delta\theta_t &= \alpha\,\delta_t\,\vec{e}_t \\
c_{t+1} &= \rho_{t+1}\,c_t + g_{t+1} \\
\vec{e}_{t+1} &= \gamma\,\lambda\,\rho_{t+1}\,\vec{e}_t + c_{t+1}\,\phi_{t+1}
\end{aligned}$$

3. At the end of the episode,

$$\theta \leftarrow \theta + \sum_t \Delta\theta_t$$

Figure 1. Incremental implementation of importance-sampled TD(λ)

Theorem 4 *The backward-view description given as Algorithm 1 is equivalent to the forward-view definition (3).*

Proof: From the algorithm definition,

$$c_t = \sum_{k=0}^{t} g_k \prod_{j=k+1}^{t} \rho_j.$$

Therefore, in the forward view, we can re-write the sum of the updates that occur during an episode as:

$$\begin{aligned}
&\sum_{t=0}^{T-1} \alpha\left(\bar{R}_t^\lambda - \theta^T\phi_t\right)\phi_t \sum_{k=0}^{t} g_k \rho_{k+1}\cdots\rho_t \\
&\qquad = \sum_{t=0}^{T-1} \alpha\left(\bar{R}_t^\lambda - \theta^T\phi_t\right)\phi_t c_t.
\end{aligned}$$

In the backward view, the eligibility trace at time t is:

$$\vec{e}_t = \sum_{k=0}^{t} c_k \phi_k (\gamma\lambda)^{t-k} \prod_{j=k+1}^{t} \rho_j,$$

and the sum of the updates that occur during an episode is:

$$\begin{aligned}
\sum_{t=0}^{T-1} \alpha\delta_t \vec{e}_t &= \sum_{t=0}^{T-1} \alpha\delta_t \sum_{k=0}^{t} c_k \phi_k (\gamma\lambda)^{t-k} \prod_{j=k+1}^{t} \rho_j \\
&= \sum_{t=0}^{T-1} \alpha c_t \phi_t \sum_{k=t}^{T-1} (\gamma\lambda)^{k-t} \delta_k \prod_{j=t+1}^{k} \rho_j \\
&= \sum_{t=0}^{T-1} \alpha c_t \phi_t \left(\bar{R}_t^{\lambda} - \theta^T \phi_t \right). \quad \diamond
\end{aligned}$$

5. An Empirical Illustration

To illustrate our algorithm we use the 11 x 11 grid-world environment depicted in Figure 2. The MDP is deterministic and has 4 actions, moving `up`, `down`, `left` or `right`. If the agent bumps into a wall, it remains in the same state. The four corner states are terminal. The agent receives a reward of +1 for the actions entering the bottom-right and upper-left corners, and −1 for entering the other two corners. All the other rewards are 0. The initial state is in the center, and the initial action is chosen randomly to be `right` or `left`. The target policy chooses `down` 40% of the time and `up` 10% of the time, with `right` and `left` chosen 25% of the time. The behavior policy is similar except with reversed `up`/`down` probabilities; it chooses `down` 10% of the time and `up` 40% of the time. In order to ensure that all the conditions of our convergence theorem are respected, trials are limited to 1000 time steps. However, this upper limit was never reached during our experiments.

The features used by the function approximator are overlapping stripes of width 3, parallel to the vertical axis. There are 13 such stripes. One consequence is that under the target policy, allactions from the leftmost column have negative value, whereas all actions from the rightmost column have positive values. The situation is reversed under the behavior policy.

We implemented the incremental (backward view) version of importance sampled TD(λ), with $\lambda = 0$ and $\lambda = 0.9$, and updates taking place only at the end of an episode. Because the results are very similar, we only present the data for $\lambda = 0.9$. The initial parameter of the function approximator was $\theta_0 = \vec{0}$.

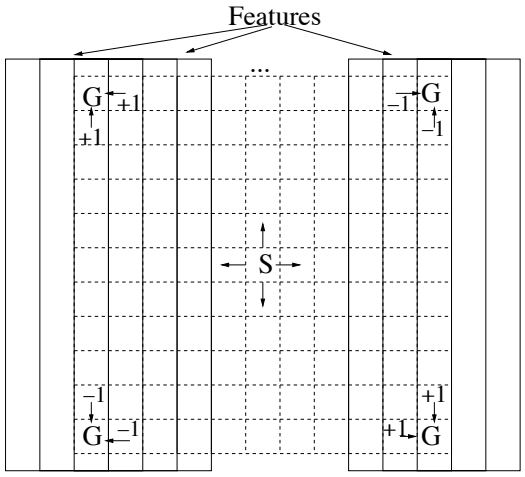

Figure 2. Gridworld MDP used in empirical illustration. An 11 x 11 grid is overlayed with stripes, each 3 cells wide.

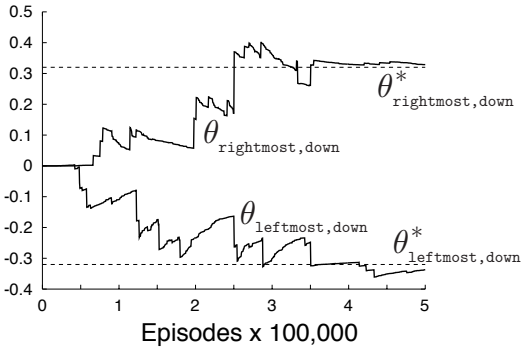

Figure 3. Trajectory of two elements of the parameter vector, under our algorithm, converging to their asymptotic values. The step size α was reduced over time. leftmost and rightmost features respectively

Figure 3 shows the parameter values corresponding to the leftmost and rightmost features, for the `down` action, for a *single* learning run. We used a decaying schedule for the learning rate parameter α, starting with a value of $\alpha_0 = 2^{-12}$ for $T = 10^6$ time steps, then using $\alpha_0/2$ for $2T$ time steps, $\alpha_0/4$ for $4T$ time steps, etc. As predicted by the theory, the parameter vector converged to the correct value. Similar results occurred for all elements of the parameter vector and for all actions.

We also compared our algorithm with the naive importance sampling algorithm (Section 1). We expected the naive algorithm to have higher variance than our approach, due to the higher variance of the importance sampling correction factors. For each algorithm, we experimented with fixed values of α between 2^{-12} and 2^{-17}. For each parameter value, we performed 50

independent runs of 100000 episodes each, saving the parameter values after every 100 episodes. Figure 4 compares the root mean squared error of the two algorithm,s compared to the asymptotic parameter values, for each learning rate, averaged over the 50 runs and over the last 10 data points. As expected, our algorithm showed significantly lower error than the naive algorithm.

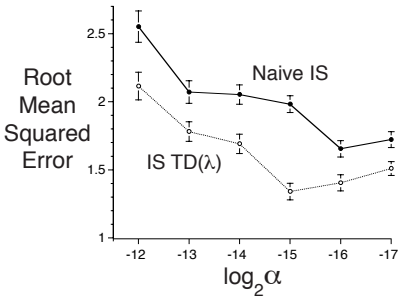

Figure 4. Comparison of the naive importance sampling algorithm with ours, after 100000 episodes.

6. Possibility of Weighted Importance Sampling Methods

We have introduced a new, off-policy version of linear TD(λ) and shown that it converges near the best solution consistent with its structure. However, excessive variance remains an issue, and there may be algorithms that reach the same asymptotic solution faster or under more general conditions.

One salient possibility is to devise some sort of *weighted* importance sampling version of our algorithm. Weighted importance sampling is widely known to produce lower variance estimates than conventional importance sampling, at the cost of introducing transient bias (bias that decreases to zero as the number of samples increases to infinity).

For example, in our earlier work with table-lookup approximations (Precup, Sutton and Singh, 2000), we discussed an importance sampling estimate

$$Q_N^{IS}(s,a) = \frac{\sum_{i=1}^{N} R_i w_i}{N},$$

where each R_i is a return in an episode under the behavior policy after an occurrence of state–action pair s, a, and the weight w_i is a product of importance sampling correction ratios $\rho_{t+1}\rho_{t+2}\cdots\rho_{T-1}$ (where t is the time of occurrence of s, a, and T the last time, within the episode). As in the current paper, this weight is chosen such that the product $R_i w_i$ has the proper expected value for the target policy, i.e., such that $E_b\{R_i w_i\} = Q^\pi(s,a)$. By the law of large numbers, Q_N^{IS} converges w.p.1 to $Q^\pi(s,a)$, if the R_i are bounded. But the w_i might have infinite variance, and so Q_N^{IS} might also have unbounded variance. However, the corresponding *weighted* version of this tabular importance sampling estimator,

$$Q_N^{ISW}(s,a) = \frac{\sum_{i=1}^{N} R_i w_i}{\sum_{i=1}^{N} w_i},$$

which also converges to Q^π w.p.1, has variance which goes to zero as N grows, as we now show. First we need an additional definition and some standard results (e.g. Durrett, 1996):

Definition 1 *A sequence of e_N* converges in probability *to H iff, for any $\epsilon > 0$,*

$$\lim_{N\to\infty} Pr\{|e_N - H| > \epsilon\} = 0.$$

Theorem 5 Weak Law of Large Numbers. *Let $\{X_i\}_{i=1}^{\infty}$ be a sequence of i.i.d. random variables such that $E\{|X_i|\} < \infty$, then the estimator $e_N = \frac{1}{N}\sum_{i=1}^{N} X_i$ converges in probability to $E\{X_i\}$.*

(Under these same hypotheses, the stronger law of large numbers (convergence w.p.1) also holds, but we will not need it for our result.)

Theorem 6 *If $|e_N|$ is bounded, then convergence in probability of e_N to H implies that* $\lim_{N\to\infty} var(e_N) = 0$.

Proof: Suppose that $|e_N| \le C$, for some constant C. This also implies that $|H| \le C$ and that $|e_N - H| \le 2C$.

Pick any $\epsilon > 0$. Then:

$$\mathrm{var}(e_N) \le E\{(e_N - H)^2\} \le \epsilon^2 + 4C^2 \cdot Pr\{|e_N - H| > \epsilon\}.$$

Now take the limit as $N \to \infty$. Since the e_N converge in probability, the rightmost term goes to zero and so $\lim_{N\to\infty} \mathrm{var}(e_N) \le \epsilon^2$.

But this is true for any $\epsilon > 0$, so the theorem follows.$\diamond$

Using these we can show:

Theorem 7 *For $\gamma < 1$, $var(Q_N^{ISW})$ goes to zero as N goes to infinity.*

Proof: First we show convergence in probability. We can write the estimator as a "top" part over a "bottom" part (dropping the s, a everywhere):

$$Q_N^{ISW} = \frac{T_N}{B_N}$$

where

$$T_N = \frac{1}{N}\sum_{i=1}^{N} R_i w_i \qquad \text{and} \qquad B_N = \frac{1}{N}\sum_{i=1}^{N} w_i.$$

Because $E\{w_i\} = 1$ is finite, we can apply the weak law of large numbers twice here to show that T_N converges in probability to Q^π and B_N converges in probability to 1. Thus we know that the top is very close to Q^π except for a tiny probability and the bottom is very close to 1 except for a tiny probability. Now we can ignore what happens with tiny probability; that will correspond to the tiny probability with which Q_N^{ISW} is allowed to be significantly different from Q^π. So consider the cases when top and bottom *are* very near Q^π and 1 respectively. If we pick the "very near" close enough, then we can also bound the difference of the ratio $\frac{T_N}{B_N}$ from $\frac{Q^\pi}{1}$. So we get that Q_N^{ISW} is arbitrarily close to Q^π except for an arbitrarily small probability, i.e., Q_N^{ISW} converges in probability to Q^π.

Now we seek to apply Theorem 6, for which we need only to show that $|Q_N^{ISW}|$ is bounded. From its definition, $|Q_N^{ISW}|$ can clearly be no greater than the largest possible $|R_i|$. For bounded individual rewards and $\gamma < 1$, we have $|R_i| < \frac{r_{\max}}{1-\gamma}$. Thus, Theorem 6 applies and so $\lim_{N\to\infty} \mathrm{var}(Q_N^{ISW}) = 0$. $\diamond$

Thus, in the tabular case the weighted algorithm has vanishing variance. The same cannot be said for the conventional importance sampling algorithm. It seems plausible that a similar pattern of results could hold for the case with linear function approximation. To explore this possibility would of course require some form of weighted importance sampling that was consistent with function approximation.

Acknowledgements

The authors gratefully thank Csaba Szepeszvari for pointing out a problem in the previous version of our convergence proof, and David McAllester and Satinder Singh for many helpful discussions about importance sampling.

References

Baird, L. C. (1995). Residual algorithms: Reinforcement learning with function approximation. *Proceedings of the Twelfth International Conference on Machine Learning*, pp. 30–37. Morgan Kaufmann. See particularly the Nov. 22 amended version at http://www.leemon.com/papers/residual/residual.pdf.

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

Dietterich, T. G. (2000). Hierarchical reinforcement learning with the MAXQ value function decomposition. *Journal of Artificial Intelligence Research 13*:227-303.

Durrett, Richard (1996). *Probability: Theory and Examples*. Duxbury Press.

Gordon, G. J. (1995). Stable function approximation in dynamic programming. *Proceedings of the Twelfth Int. Conf. on Machine Learning*, pp. 261–268. Morgan Kaufmann.

Gordon, G. J. (1999). *Approximate Solutions to Markov Decision Processes*. Doctoral Thesis. Dept. of Computer Science, Carnegie-Mellon University, Technical Report CMU-CS-99-143.

Gurvits, L., Lin, L.-J., and Hanson, S. J. (unpublished). Incremental learning of evaluation functions for absorbing Markov chains: New methods and theorems.

Parr, R. (1998). Hierarchical control and learning for Markov decision processes. PhD Thesis, University of California at Berkeley.

Parr, R., Russell, S. (1998). Reinforcement learning with hierarchies of machines. *Advances in Neural Information Processing Systems 10*, pp. 1043–1049. MIT Press, Cambridge, MA.

Precup, D., Sutton, R. S., Singh, S. (2000). Eligibility traces for off-policy policy evaluation. Proceedings of the 17th International Conference on Machine Learning, pp. 759-766. Morgan Kaufmann.

Sutton, R. S., Precup, D., Singh, S. (1999). Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence 112*:181-211.

Sutton, R. S., Barto, A. G. (1998). *Reinforcement Learning: An Introduction*. MIT Press.

Tadic, V. (2001). On the Convergence of Temporal-Difference Learning with Linear Function Approximation. *Machine Learning 42(3)*:241–267.

Tsitsiklis, J. N., and Van Roy, B. (1997). An analysis of temporal-difference learning with function approximation. *IEEE Transactions on Automatic Control 42*:674–690.

Watkins, C. J. C. H. (1989). *Learning from Delayed Rewards*. Ph.D. thesis, Cambridge University.