Efficient Asymptotic Approximation in Temporal Difference Learning

Frédérick Garcia and Florent Serre[1]

Abstract. TD(λ) is an algorithm that learns the value function associated to a policy in a Markov Decision Process (MDP). We propose in this paper an asymptotic approximation of online TD(λ) with accumulating eligibility trace, called ATD(λ). We then use the Ordinary Differential Equation (ODE) method to analyse ATD(λ) and to optimize the choice of the λ parameter and the learning stepsize, and we introduce ATD, a new efficient temporal difference learning algorithm.

1 INTRODUCTION

The TD(λ) algorithm for learning the value function of a given policy is surely one of the most important results that the reinforcement learning methodology has produced in the domain of sequential decision problems under uncertainty [12]. The TD(λ) learning rule is the core of most of the existing reinforcement learning algorithms, and is also now used in order to improve and to extend the range of applicability of classical stochastic dynamic programming algorithms [2].

The convergence properties of TD(λ) have been deeply studied and are now well established [4, 2]. Surprisingly, very few works (see [9]) have been concerned with the analysis of the effects of the λ parameter choice on the rate of convergence of TD(λ), which appears to be in practice an important and difficult question [2, pages 200-201].

Another important point concerning TD(λ) is relative to the computational cost of the update process. Classical online implementations, based on eligibility traces for look-up table representations [10], have an update complexity bounded by the size of the state-space, which can be too much high. Despite several attempts to solve that problem [3, 15], there is still a need for getting new insights into the part played by the multiple state evaluation update in the efficiency of TD(λ).

In this paper, we propose an original approach for tackling these two questions, by considering an asymptotic approximation of online TD(λ) based on accumulating eligibility traces. Two main results are presented. First, a new algorithm called ATD(λ) is introduced, which is asymptotically equivalent to TD(λ), but with a small constant update complexity per iteration (section 3). Secondly, by considering the ordinary differential equation (ODE) method, we propose a criterion for optimizing ATD(λ) as a function of the stepsizes and the parameter λ. Solving this optimization problem led us to a new efficient algorithm called ATD, which appears to be very similar to an average-reward variant of TD(λ) as recently presented in [13] (section 4). The experimental study we conducted confirms the soundness of the criterion and the optimal behaviour of ATD (section 5).

2 TEMPORAL DIFFERENCE LEARNING ALGORITHMS

Like most of the reinforcement learning algorithms, TD(λ) can be described within the framework of Markov Decision Processes (MDP). The standard stationnary infinite-horizon MDP model we consider here [8] is defined by a finite state space S of size n_S and a finite action space A, by a markovian dynamic on S characterized by the transition probabilities $p(s'|s,a)$ of moving from s to s' by applying the action a at any instant $t \in \mathbb{N}$, and by the local reward functions $r(s,a,s') \in \mathbb{R}$ associated to each transition (s,a,s').

A policy, or decision rule, is a function $\pi : S \rightarrow A$ that assigns an action $a = \pi(s)$ to any possible state s. Given an initial state s_0, following a policy π defines a set of possible trajectories $s_0 \rightarrow s_1 \rightarrow \cdots \rightarrow s_n \ldots$, with the probabilities $p(s_{i+1} \mid s_i, \pi(s_i))$. To each of these trajectories is also associated a reward sequence $r_0 \rightarrow r_1 \rightarrow \cdots \rightarrow r_n \ldots$, with $r_i = r(s_i, \pi(s_i), s_{i+1})$.

The optimization problem associated to a MDP is to search for a policy π that maximizes for any initial state a value function defined as a measure of the expected sum of the rewards r_i along a trajectory. The most common optimality criterion for stationnary infinite-horizon MDP corresponds to the discounted value function from S to $\mathbb{R}$:

$$\forall s_0 \quad V^\pi(s_0) = E[\sum_{i=0}^{\infty} \gamma^i r(s_i, \pi(s_i), s_{i+1})],$$

where the discount factor $0 \leq \gamma < 1$ is a coefficient depending on the application domain.

The TD(λ) method [12] learns an estimation of the value function V^π of a policy π from the observation of trajectories of the process obtained by following π. Maintaining an estimation of the transition probabilities $p(s' \mid s, \pi(s))$ is not required and thus the method is well adapted for unknown and large Markov Decision Processes.

For the discounted value function, the potentially infinite length of the trajectories leads to retain among the different variants of the TD(λ) method (see [12][2, sec. 5.3.3]) the ones based on eligibility traces and online updates of the value function V^π [11]. The principle of these algorithms consists in updating after each observed transition (s_n, s_{n+1}, r_n) the estimated value function V_n by

$$\forall s \in S \quad V_{n+1}(s) \leftarrow V_n(s) + \alpha_n(s) z_n(s) d_n \qquad (1)$$

where V_n is the current estimation of the value function V^π at time n, $d_n = r_n + \gamma V_n(s_{n+1}) - V_n(s_n)$ is the temporal difference error

[1] INRA - Unité de Biométrie et Intelligence Artificielle. BP 27, Auzeville. 31326 Castanet Tolosan cedex, France. email:fgarcia@toulouse.inra.fr

ECAI 2000. *14th European Conference on Artificial Intelligence*
Edited by Werner Horn
Published in 2000 by IOS Press

term, $\alpha_n(s)$ is the stepsize and $z_n(s)$ is the eligibility trace vector (of dimension n_S).

The classical *accumulating eligibility trace* is iteratively updated according to
$\forall s \in S \quad z_{-1}(s) = 0$, and for $n \geq 0$

$$z_n(s) \leftarrow \begin{cases} \gamma\lambda z_{n-1}(s) & \text{if } s \neq s_n, \\ \gamma\lambda z_{n-1}(s) + 1 & \text{if } s = s_n. \end{cases} \quad (2)$$

Hence, eligibility traces decay exponentially according to the product of a parameter λ and the discount factor γ, with $\gamma\lambda < 1$, and when a state is visited, its trace is increased by 1.

It has been shown that this algorithm converges with probability 1 to the value function V^π under some general suitable assumptions on the stepsizes $\alpha_n(s)$ and on the Markov chain $\{s_n\}$ defined by π [2].

With respect to the number of transitions required to converge, intermediate or large values of λ seem to work best, but with a strong dependency on the choice of the stepsize. Nevertheless, update costs of V_n and z_n are proportional to the size of S for $\lambda > 0$: for large state space problems, practical implementations must limit the number of states to be updated, or try to group and postpone updates until they are really needed [3, 15]

3 THE ATD(λ) ALGORITHM

This section is dedicated to the analysis of the asymptotic average behaviour of the accumulating trace $z_n(s)$ when $n \to \infty$, in order to define an asymptotic approximation of online TD(λ). In the following we denote by P the transition probability matrix of the Markov chain defined on S by the policy π, with $P_{s,s'} = p(s' \mid s, \pi(s))$. We assume that P defines a recurrent aperiodic irreductible Markov chain, and therefore has a unique invariant distribution vector $\mu > 0$. Let M be the diagonal matrix of diagonal coeffcients $\mu(s)$, and $P^* = lim_{n\to\infty} P^n = UM$ with $\forall s, s'\ U_{s,s'} = 1$. Finally, we define the reward vector R by $\forall s \in S$, $R(s) = \sum_{s'} P_{s,s'} r(s, \pi(s), s')$.

3.1 Asymptotic accumulating trace

Let us consider a trajectory $s_0 \to s_1 \to \cdots \to s_n \ldots$ obtained by the simulation of the policy π from the initial state s_0. In the following proposition we give the asymptotic expectation of the accumulating trace vector on this trajectory.

Proposition 1 *(asymptotic average accumulating trace)*

$$\forall s \in S: \quad E[z(s)] = \lim_{n\to\infty} E[z_n(s)] = \frac{\mu(s)}{1-\gamma\lambda}. \quad (3)$$

Proof: (sketch) We define the following arriving times:

$$\begin{cases} T_0(s) = \inf\{n > 0 / s_n = s\} \\ \forall p > 0: \quad T_p(s) = \inf\{n > T_{p-1}(s) / s_n = s\}. \end{cases}$$

We know then that for a recurrent irreductible Markov chain, and for $p > 1$, the interarrival times $T_p(s) - T_{p-1}(s)$ are independent and identically distributed random variables, with $E[T_1(s) - T_0(s)] = \frac{1}{\mu(s)}$, and are independent of $T_0(s)$. Since $\forall s \in S$, $T_n(s) = (T_n(s) - T_{n-1}(s)) + \cdots + (T_1(s) - T_0(s)) + T_0(s)$, $T_n(s)$ is a delayed renewal process [7]. It is easy to show that the accumulating trace (2) can be written as

$$z_n(s) = \sum_{p=0}^{n} (\gamma\lambda)^{n-T_p(s)} \mathbb{1}_{T_p(s)\leq n},$$

$$\text{and thus } E[z_n(s)] = \sum_{k=0}^{k=n} (\gamma\lambda)^k \sum_{p=0}^{\infty} P(T_p(s) = k).$$

From this last relation, we apply the Blackwell's renewall theorem [7] that establishes the formula. □

As we see, the asymptotic expectation of the trace is equal to the product of the factor $\frac{1}{1-\gamma\lambda}$, and the invariant distribution vector of the Markov chain. This formula allows us to interpret the role of the trace $z_n(s)$ in the update rule (1) as a gain that asymptotically and in average puts more credit to states that occur more frequently.

Proposition 1 establishes that the trace $z_n(s)$ is a random variable of mean equal to $\frac{\mu(s)}{1-\gamma\lambda}$ when $n \to \infty$. We can also consider the asymptotic expectation of the trace at time n conditioned on the current state s_n, which can be defined from (2) and (3) by

$$\bar{z}_n(s) = \begin{cases} \chi\mu(s) & \text{if } s \neq s_n, \\ \chi\mu(s) + 1 & \text{if } s = s_n. \end{cases}$$

with $\chi = \dfrac{\gamma\lambda}{1-\gamma\lambda}$.

3.2 Average asymptotic TD(λ)

We propose in this paper to substitute in the TD(λ) update rule (1) the trace factor $z_n(s)$ by its asymptotic average value $\bar{z}_n(s)$. That defines a new algorithm called ATD(λ) that can be written in vector notation

$$V_{n+1} \leftarrow V_n + A_n \Gamma^z D_n. \quad (4)$$

Here, D_n is the temporal difference error vector

$$D_n = \begin{pmatrix} 0 \\ d_n \\ 0 \end{pmatrix} \leftarrow s_n,$$

A_n is the diagonal stepsize matrix

$$A_n = \begin{pmatrix} \alpha_n(s_1) & & \\ & \ddots & \\ & & \alpha_n(s_N) \end{pmatrix}$$

and Γ^z the eligibility matrix

$$\Gamma^z = I + \chi \begin{pmatrix} | & & | \\ \mu & \ldots & \mu \\ | & & | \end{pmatrix} = I + \chi MU.$$

ATD(λ) has the same convergence properties as TD(λ). Nevertheless, a direct implementation of this update rule (4) would have a complexity similar to TD(λ), proportional to n_S. Fortunately, under a simple assumption on the stepsizes $\alpha_n(s)$, there exists an equivalent expression of (4) that requires a small number of updates per iteration, independant of the number of states in S.

Proposition 2 *(ATD(λ))*
We assume that $\forall n, s\, \alpha_n(s) = \alpha_n \alpha(s)$, with $\alpha_n, \alpha(s) > 0$, $\sum_n \alpha_n = +\infty$ and $\sum_n {\alpha_n}^2 < +\infty$. Let A be the diagonal matrix of the $\alpha(s)$ coefficients, and $A_n = \alpha_n A$. Then,
(i) V_n converges with probability 1 to V^π ;
(ii) $V_n = W_n + \chi A\mu\rho_n$, where W_n is a new relative value function

on S and ρ_n a scaling factor, that can be both updated after each transition by

$$\begin{array}{rcl} W_{n+1}(s_n) & \leftarrow & W_n(s_n) + \alpha_n \alpha(s_n) d_n, \\ \rho_{n+1} & \leftarrow & \rho_n + \alpha_n d_n, \\ \textit{with } d_n & = & r_n + \gamma V_n(s_{n+1}) - V_n(s_n) \\ & = & r_n + \gamma W_n(s_{n+1}) - W_n(s_n) \\ & & + \chi \rho_n (\gamma \alpha(s_{n+1}) \mu(s_{n+1}) - \alpha(s_n) \mu(s_n)). \end{array}$$

Proof: (i) (sketch) Let $X_n = (s_n, s_{n+1}, r_n)$. (4) can be written $V_{n+1} = V_n + \alpha_n(B(X_n)V_n + b(X_n))$ where $B(X_n)$ is a $n_S \times n_S$ matrix, and $b(X_n)$ a vector in $\mathbb{R}^{n_S}$. Let B and b be the expected values of $B(X_n)$ and $b(X_n)$ with respect to the invariant distribution μ. We have $B = AM(\gamma P - I + (\gamma - 1)\chi P^*)$ and $b = AM(I + \chi P^* R)$. Since the bounded series $B(X_n)$ and $b(X_n)$ converge exponentially fast to B and b, and B is negative definite, we can apply the proposition 4.8 in [2, sec. 4.4.1] that establishes the convergence of V_n to V^π which is the unique solution V of $BV + b = 0$.

$$\begin{array}{rcl} \text{(ii) We have } V_{n+1} & = & V_n + A_n \Gamma^z D_n \\ & = & V_n + A_n D_n + \chi A_n M U D_n \\ & = & V_n + A_n D_n + \chi \alpha_n A \mu d_n \\ & = & \displaystyle\sum_{p=0}^{n-1} A_p D_p + \chi A \mu \sum_{p=0}^{n-1} \alpha_p d_p. \end{array}$$

The proposition is established by defining the two series $\rho_n = \sum_{p=0}^{n-1} \alpha_p d_p$ and $W_n = \sum_{p=0}^{n-1} A_p D_p$. □

In order to calculate the error term d_n, the steady state probabilities $\mu(s)$ are needed. They can be estimated online by

$$\forall s \quad \mu(s) \approx \frac{N_n(s)}{n}$$

where $N_n(s)$ is the number of visits of state s at time n.

This ATD(λ) algorithm is particularly interesting because it only requires 3 updates per transition ($W_n(s_n)$, ρ_n and $N_n(s_n)$) and thus can be really faster than the original algorithm (4). Hence, as it is an asymptotic approximation of the TD(λ) algorithm (1), one can expect from ATD(λ) the convergence efficiency of TD(λ) for $\lambda > 0$, with the lower computational cost of TD(0).

4 ATD: OPTIMIZING ATD(λ)

The asymptotic approximation ATD(λ) of TD(λ)

$$V_{n+1} \leftarrow V_n + A_n \Gamma^z D_n.$$

corresponds to a gain matrix variant of the basic TD(0) algorithm

$$V_{n+1} \leftarrow V_n + \frac{1}{n} D_n.$$

The use of a gain matrix in order to guide and accelerate the convergence of a stochastic adaptive algorithm is a classic result in stochastic approximation theory [1, 6]. Our objective in this section is to compare the ATD(λ) gain matrix with the optimal one provided by the ordinary differential equation (ODE) method, and to optimize the $\alpha_n(s)$ and λ choices by minimizing the difference between them.

4.1 Optimal gain matrix for TD(0)

The ODE method proposes some analytic tools for analysing and optimizing the convergence of the general stochastic algorithm

$$\theta_{n+1} \leftarrow \theta_n + \frac{1}{n} H(\theta_n, X_n)$$

where θ_n is the parameter vector, and X_n the input random vector that brings some information on θ_n at time n. Classic reinforcement learning algorithms like Q-Learning or TD(λ) can be analysed with the ODE method [2, 6, 5].

As ATD(λ) appears to a be a gain matrix variant of TD(0), it is natural to compare the corresponding gain matrix $A_n \Gamma^z$ with the optimal gain matrix for TD(0) that can be derived from the ODE theory.

For the TD(0) algorithm, we have $\theta_n = V_n$, $X_n = (s_n, s_{n+1}, r_n)$ and $H(\theta_n, X_n) = D_n$. We know that $V_n \to V^\pi$ and the optimal gain matrix is the one that minimizes the asymptotic variance $\lim_{n\to\infty} \| V_n^- V^\pi \|^2$. It is defined by

$$\Gamma^* = -h_V(V^\pi)^{-1}$$

where h_V is the jacobian matrix of the function

$$\begin{array}{rcl} h(V) & = & \displaystyle\lim_{n\to\infty} E_V(H(V, X_n)) \\ & = & M(R + \gamma P V - V) \end{array}$$

We obtain $h_V(V^\pi) = M(\gamma P - I)$. Since $M > 0$ and $0 \leq \gamma < 1$, the inverse of $M(\gamma P - I)$ exists and we have

$$\begin{array}{rcl} \Gamma^*(s, s') & = & (I - \gamma P)^{-1} M^{-1} \\ & = & \displaystyle\frac{1}{\mu(s')} \left(\sum_{k=0}^{\infty} \gamma^k P^k (s_k = s' | s_0 = s) \right). \end{array}$$

Note that the modified TD(0) update rule obtained by considering the optimal gain matrix

$$V_{n+1} \leftarrow V_n + \frac{1}{n} \Gamma^* D_n$$

cannot be implemented in practice because the matrix P is unknown, and the estimation of the inverse of $(I - \gamma P)^{-1}$ might be very costly.

4.2 The ATD algorithm

We consider the case $A_n \Gamma^z = \frac{1}{n} \Gamma_{\lambda,\alpha}$ and we propose to optimize the choice of $\alpha(s)$ and λ by minimizing the norm $\| \Gamma_{\lambda,\alpha} - \Gamma^* \|_2$ of the difference between the matrix gain $\Gamma_{\alpha,\lambda}$ and Γ^*. Due to the complexity of the global minimization problem

$$min_{\lambda,\alpha(s)} \| \Gamma_{\lambda,\alpha} - \Gamma^* \|_2,$$

we only considered the optimization of the λ parameter for different stepsize definitions A_n. The best results were obtained for $\alpha_n(s) = \frac{1}{n} \frac{1}{\mu(s)}$, that is $A_n = \frac{1}{n} M^{-1}$. In that case the minimization of the criterion $\| M^{-1} \Gamma^z - \Gamma^* \|_2$ leads to the optimal χ value

$$\chi_{opt} = \frac{1}{{n_S}^2} \left(\sum_{s,s'} \Gamma^*_{s,s'} - \sum_s \frac{1}{\mu(s)} \right)$$

and $\lambda_{opt} = \frac{1}{\gamma} \frac{\chi_{opt}}{1 + \chi_{opt}}$.

For many simulated problems (see section 5) this λ_{opt} value was very close to 1, as it is theoretically the case if the invariant distribution is uniform: $\forall s\, \mu(s) = \frac{1}{n_S}$. For that reason, we introduce ATD as the new algorithm defined from ATD(λ) by the choices $\lambda = 1$ and $\forall n, s\, \alpha_n(s) = \frac{1}{n}\frac{1}{\mu(s)}$.

Note that for different stepsizes, the optimal λ values can of course be different. For instance, the choice $\alpha_n(s) = \frac{1}{n}$ leads to an optimal value $\lambda_{opt} \approx \frac{1}{\gamma}$.

From proposition (2), it is possible to implement efficiently ATD with the two update rules on W_n and ρ_n. This factored algorithm has some specific properties described in the following proposition (e stands for the vector with all components equal to 1).

Proposition 3 *(ATD)*
(i) ATD can be implemented by

$$\begin{array}{rcl} V_n & = & W_n + \frac{\gamma}{1-\gamma}\rho_n e, \\ \textit{with } W_{n+1}(s_n) & \leftarrow & W_n(s_n) + \frac{1}{n}\frac{1}{\mu(s_n)} d_n, \\ \rho_{n+1} & \leftarrow & \rho_n + \frac{1}{n} d_n, \\ \textit{and } d_n & = & r_n - \gamma\rho_n + \gamma W_n(s_{n+1}) - W_n(s_n). \end{array}$$

(ii) (W_n, ρ_n) converges in probability 1 to (W^π, ρ^π) such that $V^\pi = W^\pi + \frac{\gamma}{1-\gamma}\rho^\pi e$ is the discounted value function of the policy π, and ρ^π its average gain:

$$\rho^\pi = \lim_{n\to\infty} E\left[\frac{1}{n}\sum_{p=0}^{n-1} r_p\right].$$

Proof: (i) Since $\forall n, s\, \alpha_n = \frac{1}{n}$ and $\alpha(s) = \frac{1}{\mu(s)}$, proposition (2) establishes that $V_n = W_n + \chi A\mu\rho_n$. Then we note that $\forall s\, \chi A\mu(s) = \frac{\gamma}{1-\gamma}\frac{1}{\mu(s)}\mu(s) = \frac{\gamma}{1-\gamma}$, and

$$\begin{array}{rcl} d_n & = & r_n + \gamma V_n(s_{n+1}) - V_n(s_n) \\ & = & r_n + \gamma W_n(s_{n+1}) + \frac{\gamma^2}{1-\gamma}\rho_n - W_n(s_n) - \frac{\gamma}{1-\gamma}\rho_n \\ & = & r_n + \gamma W_n(s_{n+1}) - W_n(s_n) - \gamma\rho_n. \end{array}$$

(ii) (sketch) We first show by induction that $\forall n\, \rho_n = \sum_s \mu(s) W_n(s)$. Indeed,

$$\begin{array}{rcl} \rho_{n+1} & = & \rho_n + \frac{1}{n} d_n \\ & = & \sum_s \mu(s) W_n(s) + \frac{1}{n} d_n \\ & = & \sum_s \mu(s) W_{n+1}(s) - \mu(s_n)(\frac{1}{\mu(s_n) n} d_n) + \frac{1}{n} d_n \\ & = & \sum_s \mu(s) W_{n+1}(s). \end{array}$$

It follows that the W_n iteration can be made independant on ρ_n and, similarly to the proof of proposition (2), can be written $W_{n+1} = W_n + \frac{1}{n}(B(X_n)W_n + b(X_n))$. We then show that $B(X_n)$ and $b(X_n)$ converge exponentially fast to the negative definite matrix $B = \gamma P - I - \gamma P*$ and $b = R$. Thus W_n converges w.p.1 to the unique solution W^π of $(I - \gamma P + \gamma P^*)W^\pi = R$. By multiplicating both sides by P^*, and from the relations $P^*P = PP^* = P^*$, we obtain $P^*W^\pi = P^*R$.
Since $\rho_n = \sum_s \mu(s) W_n(s)$, we have $\rho_n \to \rho^\pi$ with $\rho^\pi = \sum_s \mu(s) W^\pi(s)$, and from $P^*W^\pi = P^*R$, $\rho^\pi = \sum_s \mu(s) R(s)$, which is an equivalent definition of the average gain of the policy π. Then from (i) and proposition (2), V_n converges w.p.1 to $V^\pi = W^\pi + \frac{\gamma}{1-\gamma}\rho^\pi e$. □

Surprisingly, one can note that the ATD algorithm that has been defined for $\gamma < 1$ can directly be extended to the limit case $\gamma = 1$. Indeed, the new update rules on ρ_n and W_n with $d_n = r_n - \rho_n + W_n(s_n{+}1) - W_n(s_n)$ exactly correspond to the average cost temporal difference learning algorithm recently proposed by Tsitsiklis and Van Roy in order to learn the bias-value function and the average gain of a policy π [13]. Hence ATD can be seen as a bridge between TD(λ) algorithms for the discounted expected cost problems, and average cost TD(λ) algorithms for the average cost per stage problems.

5 SIMULATIONS

In order to analyse experimentally the quality of the asymptotic approximation of TD(λ) by ATD(λ), the choice of the norm $\parallel \Gamma_{\lambda,\alpha} - \Gamma^* \parallel_2$ as an evaluation of the pair (λ, α), and finally the optimality of the ATD algorithm, we made some simulations on randomly generated Markov chains, with all components $P_{s,s'} > 0$, and with random state rewards $R(s) \in [0, 1]$. We implemented TD(λ), ATD(λ) and ATD learning algorithms as described in previous sections by considering a unique long trajectory on S. All parameters were initially set to 0. The stepsizes $\frac{1}{n\mu(s)}$ were estimated online by $\frac{1}{N_n(s)}$. For each problem, V^π and Γ^* were exactly calculated, and three criteria were considered: the gain error $\parallel \Gamma_{\lambda,\alpha} - \Gamma^* \parallel_2$, the relative learning error $\frac{\|V_n - V^\pi\|_2}{\|V^\pi\|_2}$ along a trajectory, and the total computation time to achieve a 3% relative learning error.

We present here some typical results obtained during our simulations. Figure 1 shows the gain error of a Markov chain with 50 states and $\gamma = 0.7$, as a function of λ for two choices of stepsize. As we can see, the optimal λ value for $\alpha_n(s) = \frac{1}{n\mu(s)}$ is very close to 1, and leads to a better gain error than for the choice $\alpha_n(s) = \frac{1}{n\mu(s_n)}$ (this stepsize was retained because it also presented interesting properties).

On Figure 2, we plot the learning error $\frac{\|V_n - V^\pi\|_2}{\|V^\pi\|_2}$ obtained at $n = 50000$ for different λ and $\alpha_n(s)$ values on the same Markov chain than in figure 1. The results correspond to the mean values of 100 different runs. As we can see the behaviors of TD(λ) and ATD(λ) are very similar, and the experimental optimal λ values are very close to the one theoretically obtained by minimizing the gain error, as illustrated in figure 1.

Finally, we compare ATD and differents instances of TD(λ) and ATD(λ) on figure 3, with respect to the computation time (mean value on 100 runs) required to get a learning error equal to 3%, as a function of the size of the state space. The main conclusion is that ATD exhibits a very good learning behaviour, with better results than TD(λ) for $\alpha_n(s) = \frac{1}{n\mu(s)}$ and $\lambda = 1$, and than TD(λ) and ATD(λ) for $\alpha_n(s) = \frac{1}{n\mu(s_n)}$ and optimal λ values, these two algorithms being experimentally equivalent. Hence, the constant number of updates per iteration and the optimal choice $\alpha_n(s) = \frac{1}{n\mu(s)}$ and $\lambda = 1$ allows us to define a very efficient new temporal difference learning algorithm.

6 CONCLUSION

In this paper we have introduced ATD(λ), an original asymptotic approximation of online TD(λ) based on accumulating eligibility

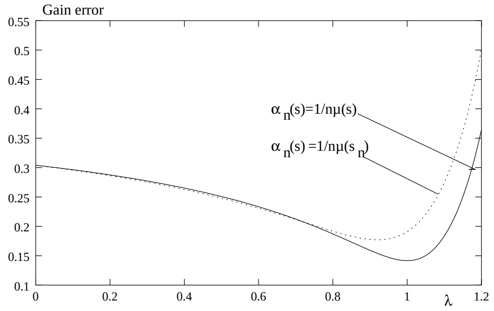

Figure 1. ATD(λ) - Gain error - 50 states, $\gamma = 0.7$

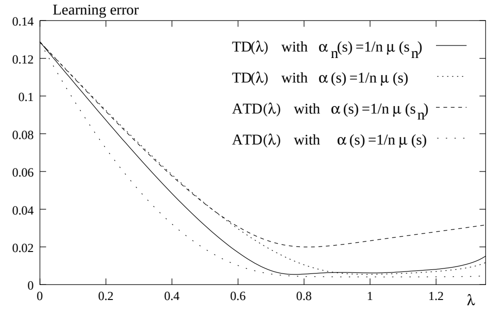

Figure 2. TD(λ) and ATD(λ) - Learning error 50 states, $\gamma = 0.7$, N_{tot} = 50000

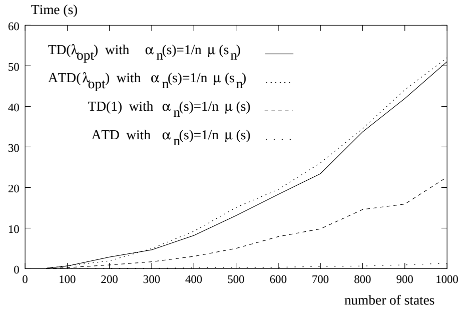

Figure 3. TD(λ) and ATD(λ) - Computation time $\gamma = 0.7$, 3% learning error

trace. By minimizing the norm of the difference between the matrix gain of ATD(λ) and the optimal matrix gain corresponding to TD(0), we have shown that there exists a strong interplay between the optimal λ value and the choice of the stepsize, and that the pair $(\lambda = 1, \alpha_n(s) = \frac{1}{N_n(s)})$ defines a new very efficient temporal difference learning algorithm called ATD, which can be interpreted as a discounted variant of the average gain temporal difference learning algorithm recently presented in [13].

Our results have been obtained for the accumulating eligibility trace. We also considered in a parallel work the case of the replacing trace [10], but it was no more possible to handle analytically the criterion $min_{\lambda,\alpha(s)} \parallel \Gamma_{\lambda,\alpha} - \Gamma^* \parallel_2$. We also limited our analysis to stepsizes proportional to $\frac{1}{n}$ as it is assumed in the ODE method, despite the fact that the few interesting results already known on convergence rates of TD(λ) algorithms were obtained for constant stepsizes (but general convergence proofs only exist for decreasing stepsizes). Nevertheless, we think that ATD, with its constant update complexity and its optimal λ value, is a very promising and simple algorithm, and we intend in the future to compare it with efficient approximate implementations of TD(λ), like TTD [3]. Furthermore, the process we followed here to define ATD(λ) and ATD can be also applied to Q(λ)-learning [14, 12], and we intend to compare these forthcoming results with the efficient implementation of Q(λ) introduced by Wiering and Schmidhuber [15].

REFERENCES

[1] A. Benveniste, M. Metivier, and P. Priouret, *Adaptive Algorithms and Stochastic Approximation*, Springer-Verlag, Berlin,New York, 1990.

[2] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming*, Athena Scientific, Belmont (MA), 1996.

[3] P. Cichosz, 'Truncating temporal differences: On the efficient implementation of TD(λ) for reinforcement learning', *Journal of Artificial Intelligence Research (JAIR)*, **2**, 287–318, (1995).

[4] P. Dayan, 'The Convergence of TD(λ) for General λ', *Machine Learning*, **8**, 341–362, (1992).

[5] F. Garcia and S. Ndiaye, 'A Learning Rate Analysis of Reinforcement-Learning Algorithms in Finite-Horizon', in *International Conference on Machine Learning*, volume 15, (1998).

[6] H. J. Kushner and G. G Yin, *Stochastic Approximation Algorithms and Applications*, Springer, 1997.

[7] S.P. Meyn and R.L. Tweedie, *Markov Chains and Stochastic Stability*, Springer-Verlag, 1993.

[8] M. L. Puterman, *Markov decision processes: discrete stochastic dynamic programming*, Wiley-Interscience, New York, 1994.

[9] S. P. Singh and P. Dayan, 'Analytical Mean Squared Error Curves in Temporal Difference learning'. unpublished report, 1996.

[10] S. P. Singh and R. S. Sutton, 'Reinforcement learning with replacing eligibility traces', *Machine Learning*, **22**, 123–158, (1996).

[11] R. S. Sutton, 'Learning to predict by the methods of temporal differences', *Machine Learning*, **3**, 9–44, (1988).

[12] R. S Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, MIT Press, Cambridge, Massachusetts, 1998.

[13] J. N. Tsitsiklis and B. Van Roy, 'Average cost temporal-difference learning', *Automatica*, **35**(11), 1799–1808, (1999).

[14] C. J. Watkins, *Learning from Delayed Rewards*, Ph.D. dissertation, Cambridge University, Cambridge, England, 1989.

[15] M. A. Wiering and J. Schmidhuber, 'Fast online Q(λ)', *Machine Learning*, **33**(1), 105–115, (1998).