Multi-Step Average-Reward Prediction via Differential TD(λ)

Abhishek Naik
University of Alberta; Amii
Edmonton, Alberta
abhishek.naik@ualberta.ca

Richard S. Sutton
DeepMind; University of Alberta; Amii
Edmonton, Alberta
rsutton@ualberta.ca

Abstract

We present Differential TD(λ), an average-reward prediction algorithm that extends Wan et al.'s (2021) Differential TD(0) from the one-step tabular case to that of multi-step linear function approximation using eligibility traces. We characterize the algorithm's fixed point and present a theorem for its almost-sure convergence. Our analysis builds on prior work by Tsitsiklis and Van Roy (1999), who proposed a trace-based prediction algorithm that is restricted to the on-policy setting. Preliminary experiments show that Differential TD(λ) converges to the fixed point predicted by the theory and that an intermediate value of the bootstrapping parameter λ results in the most sample efficiency.

Keywords: reinforcement learning; credit assignment; eligibility traces; continuing problems; average reward

Acknowledgements

We gratefully acknowledge discussions with Yi Wan and Huizhen Yu that have helped in working out the theoretical results in this paper. This work was supported by funding from Alberta Machine Intelligence Institute (Amii), DeepMind, and CIFAR.

1 Motivation

How does the mind work? This question has intrigued several generations of scientists who have studied this general question within disciplines such as psychology, cognitive neuroscience, ethology, and artificial intelligence (AI). At the intersection of these disciplines lies reinforcement learning (RL), which abstracts away biology from the computation of how a mind *could* work. It formalizes the notion of learning to achieve goals via trial and error (Sutton & Barto 2018).

Humans frequently make decisions that affect the course of their lives. For instance, choosing academics over a career in sports, choosing one life partner over another. More generally, the consequences of everyday decisions that humans and other animals make can span unpredictable timescales. Such a problem of life is well modeled as a *continuing* problem, in which entities make decisions and live with their consequences throughout their lifetimes. In contrast, the *episodic* formulation in RL better models problems in which the span of decisions' consequences are bound within episodic boundaries. The game of chess is a prototypical example of an episodic problem: each game starts afresh; the consequences of moves are restricted to a single game. The continuing problem setting is hence distinct from the more commonly studied episodic setting, and is also important because it models a variety of interesting real-world problems.

The average-reward formulation is one of two ways in which the continuing problem setting can be formulated. The discounted-reward formulation is the other. The relative ease of theoretical analysis and the positive empirical results make the discounted formulation appear a prime candidate for the continuing setting. Unfortunately, the standard discounted formulation does not carry forward to the continuing setting when the problem setting involves control with function approximation; the objective function is not well-defined. On the other hand, the average-reward objective function is clear and well-defined even for continuing control with function approximation (Sutton & Barto 2018: Ch. 10, Naik et al. 2019). Interestingly, there is some evidence that the average-reward formulation explains animal behavioral data better than the discounted formulation (e.g., Daw & Touretzky 2000, 2002).

Despite its promise, the average-reward formulation has received little attention in RL. After Schwartz's pioneering work in 1993, there was a flurry of research in average-reward RL that receded in the early 2000s—perhaps because most of the proposed methods relied on some special information about the problem that is not typically available. Wan et al. (2021) recently proposed a family of one-step average-reward algorithms for learning and planning in continuing problems that do not require such special information and are guaranteed to converge with tabular representations.

Efficient credit assignment online is critical to AI. Multi-step algorithms have been shown to propagate information faster over longer temporal spans (Sutton & Barto 2018). Eligibility traces are typically used to implement multi-step updates online and independent of the temporal span of prediction (Sutton 1988, van Hasselt & Sutton 2015). Recent work in neuroscience has found evidence of eligibility traces in the brain (see Gerstner et al. 2018 for a review). Tsitsiklis and Van Roy (1999) were the first to propose an average-reward prediction method that implements multi-step updates using eligibility traces. The nature of updates, though, restricts their algorithm to the on-policy case. Off-policy learning is key to a learning entity's ability to learn about behaviors related to but beyond its active behavior (Sutton et al. 2011). This paper takes the first step towards extending Wan et al.'s tabular one-step off-policy methods to the case of multi-step methods with function approximation. In particular, we propose Differential TD(λ), a multi-step average-reward prediction method that is guaranteed to converge with linear function approximation. We focus on the on-policy case in this short paper; the full off-policy treatment will appear in the full version.

2 Background

We formalize the interaction of the agent and the environment via a finite Markov decision process (MDP) defined by the tuple $M \doteq (S, A, R, p)$. At each time step t, the agent observes the state of the environment $S_t \in S$, selects an action $A_t \in A$, and observes a scalar reward $R_{t+1} \in R$ and the next state S_{t+1} according to the probability distribution $p(s', r|s, a) = \Pr(S_{t+1} = s', R_{t+1} = r | S_t = s, A_t = a)\ \forall s, s' \in S, a \in A, r \in R$. Such an interaction carries on ad infinitum.

In the prediction problem, the agent's goal is to estimate the value of a particular way of selecting actions. We assume this *policy* $\pi : S \times A \to [0, 1]$ is stationary.

Assumption 1. *The Markov chain corresponding to the target policy is irreducible and aperiodic.*

It follows that there is a unique stationary distribution $\mathbf{d}_\pi$ and a unique reward rate $r(\pi)$ for the Markov chain induced by policy π. Then the reward rate and differential return G_t is defined as:

$$r(\pi) \doteq \sum_s d_\pi(s) \sum_a \pi(a|s) \sum_{s',r} p(s', r|s, a)\, r, \qquad G_t \doteq R_{t+1} - r(\pi) + R_{t+2} - r(\pi) + \dots.$$

The differential value function is defined for every $s \in S$ as:

$$v_\pi(s) \doteq \mathbb{E}[G_t \mid S_t = s, A_{t:} \sim \pi] = \mathbb{E}[R_{t+1} - r(\pi) + R_{t+2} - r(\pi) + \dots \mid S_t = s, A_{t:} \sim \pi].$$

The differential value function satisfies a recursive Bellman equation:

$$v(s) = \sum_s \pi(a|s) \sum_{s',r} p(s',r|s,a)\big[r - r(\pi) + v(s')\big], \quad \forall s, \qquad \text{or,} \qquad \mathbf{v} = \mathbf{r}_\pi - r(\pi)\mathbf{1} + \mathbf{P}_\pi \mathbf{v}, \tag{1}$$

where $\mathbf{v}$ is the value function vector, $\mathbf{r}_\pi$ is the expected one-step reward from each state under policy π, $\mathbf{1}$ is a vector of all ones, and $\mathbf{P}_\pi$ is the state-to-state transition matrix under policy π. Note that the Bellman equations for the differential value function specify an under-determined system: for any solution $(\mathbf{v}, r(\pi))$, there are infinitely many solutions of the form $(\mathbf{v} + c\mathbf{1}, r(\pi))$ for $c \in \mathbb{R}$. The differential value function has a property that $\mathbf{d}_\pi^\top \mathbf{v}_\pi = 0$, that is, the average of the differential value function weighted by the stationary distribution of the policy is zero (see, e.g., Wan et al.'s (2021) Lemma B.11). Following Wan et al. (2021), we call $\mathbf{v}_\pi$ the *centered* differential value function.

In this work, we deal with the case where the differential value function is approximated by linear function approximation: $v_\pi(s) \approx \hat{v}(\mathbf{x}(s), \mathbf{w}) = \mathbf{w}^\top \mathbf{x}(s)$, where $\mathbf{x}(s)$ is the d-dimensional feature vector corresponding to state s and $\mathbf{w}$ is a vector of d learnable parameters. The function $\mathbf{x} : \mathcal{S} \to \mathbb{R}^d$ is typically a lossy encoding of the state of the environment ($d << |\mathcal{S}|$). $\mathbf{X}$ denotes a $|\mathcal{S}| \times d$ matrix whose i^{th} row is the d-dimensional feature vector corresponding to state s_i. Π denotes a projection matrix that projects a vector onto the subspace spanned by the feature vectors: $\Pi = \mathbf{X}(\mathbf{X}^\top \mathbf{D}_\pi \mathbf{X})^{-1}\mathbf{X}^\top \mathbf{D}_\pi$, where $\mathbf{D}_\pi$ is a $|\mathcal{S}| \times |\mathcal{S}|$ diagonal matrix with the i^{th} diagonal element as $d_\pi(s_i)$.

Assumption 2. *Features in $\mathbf{X}$ are linearly independent and bounded.*

We denote the n-step differential return by $G_{t:t+n}$ and the λ-return for $\lambda \in [0,1)$ by G_t^λ:

$$G_{t:t+n} \doteq \sum_{m=1}^{n} \big[R_{t+m} - r(\pi)\big] + v_\pi(S_{t+n}), \qquad G_t^\lambda \doteq (1-\lambda)\big[G_{t:t+1} + \lambda G_{t:t+2} + \lambda^2 G_{t:t+3} + \dots\big].$$

We can now define a Bellman operator $T^\lambda : \mathbb{R}^d \to \mathbb{R}^d$ as:

$$T^\lambda \mathbf{v} \doteq (1-\lambda) \sum_{k=0}^{1} \lambda^k \Big[\sum_{t=0}^{k} \big(\mathbf{P}_\pi^t \mathbf{r}_\pi - r(\pi)\,\mathbf{1}\big) + \mathbf{P}_\pi^{k+1}\mathbf{v}\Big].$$

Note that $\lambda = 0$ leads to (1). Iteratively applying this operator: $T^\lambda \hat{v} = \hat{v}$ is the basis of value-iteration algorithms to estimate $\mathbf{v}_\pi$ $\hat{v} = \mathbf{X}\mathbf{w}$ and $r(\pi)$. If $\exists \mathbf{w}_1 \in \mathbb{R}^d \ni \mathbf{X}\mathbf{w}_1 = \mathbf{1}$—like in the tabular case—then there exist multiple solutions to $T^\lambda \hat{v} = \hat{v}$ corresponding to $\mathbf{w} + c\mathbf{w}_1$, with $c \in \mathbb{R}$. Call $(\hat{v}, r(\pi))$ as a differential TD fixed point.

Now we are ready to see the new algorithm and its convergence result.

3 Differential TD(λ): A First Look

Wan et al.'s (2021) on-policy Differential TD(0) updates a weight vector $\mathbf{w} \in \mathbb{R}^d$ and a scalar reward-rate estimate $R \in \mathbb{R}$ at every time step:

$$\mathbf{w}_{t+1} \doteq \mathbf{w}_t + \alpha_t \delta_t \mathbf{x}_t, \tag{2}$$

$$R_{t+1} \doteq R_t + \eta \alpha_t \delta_t, \tag{3}$$

where, $\delta_t \doteq R_{t+1} - R_t + \mathbf{w}_t^\top \mathbf{x}_{t+1} - \mathbf{w}_t^\top \mathbf{x}_t$ is the TD error, and α, η are step-size parameters. Wan et al. (2021) show the tabular version of this one-step algorithm converges under mild conditions on the step sizes and the diversity of updates.

We extend on-policy Differential TD(0) in two ways: Differential TD(λ) uses *multi-step* updates and converges with *linear function approximation*. The multi-step updates are performed via eligibility traces. The trace vector, denoted by $\mathbf{z} \in \mathbb{R}^d$, is updated along with the weight vector and scalar reward-rate estimate as:

$$\mathbf{z}_{t+1} \doteq \lambda \mathbf{z}_t + \mathbf{x}_t, \tag{4}$$

$$\mathbf{w}_{t+1} \doteq \mathbf{w}_t + \alpha_t \delta_t \mathbf{z}_{t+1}, \tag{5}$$

$$R_{t+1} \doteq R_t + \eta \alpha_t \delta_t, \tag{6}$$

where, $\lambda \in [0,1)$ is the bootstrapping parameter and δ_t is as defined above. We impose the usual Robbins-Monro condition on the step sizes and then present the convergence result.

Assumption 3. *The step-size sequence α_t is positive, deterministic, and satisfies: $\sum_{t=0}^{\infty} \alpha_t = \infty$, $\sum_{t=0}^{\infty} \alpha_t^2 < \infty$; $\eta \in \mathbb{R}^+$.*

Theorem 1. *Under Assumptions 1, 2, 3, on-policy linear Differential TD(λ)* (4)–(6) *converges for all $\lambda \in [0,1)$ with probability 1:*

1. *R converges to the unique reward rate of the target policy: $r(\pi)$.*
2. *$\mathbf{w}$ converges to a solution of: $T^\lambda(\mathbf{X}\mathbf{w}) = \mathbf{X}\mathbf{w}$.*
3. *Let $\mathbf{w}_\infty$ be the solution from above. Then the following error bound holds w.r.t. the centered differential value function $\mathbf{v}_\pi$:*

$$\inf_{c \in \mathbb{R}} \|\mathbf{X}\mathbf{w}_\infty - (\mathbf{v}_\pi + c\mathbf{1})\|_{\mathbf{D}_\pi} \leq \frac{1}{\sqrt{(1-\tau_\lambda^2)}} \inf_{c,\mathbf{w}} \|\mathbf{X}\mathbf{w} - (\mathbf{v}_\pi + c\mathbf{1})\|_{\mathbf{D}_\pi},$$

where τ_λ is a mixing factor that exists in $[0,1)$ and $\lim_{\lambda \to 1} \tau_\lambda = 1$.

Proof. (Sketch) The proof builds on Tsitsiklis and Van Roy's (1999) theory for Average-Cost TD(λ). First, we combine the updates (5)–(6) and show it mimics a first-order ordinary differential equation of the form: $\dot{\mathbf{u}} = \mathbf{b} - \mathbf{A}\mathbf{u}$. Next, we analyze the steady-state expectations of $\mathbf{A}_t$ and $\mathbf{b}_t$ and show that the fixed point of Differential TD(λ) and is same as that of Average-Cost TD(λ). The error bound then carries over from Tsitsiklis and Van Roy's (1999) Theorem 3 (intuitively, the expression compares the estimated value function to the 'closest' valid solution of (1)). Finally, we show that Differential TD(λ) satisfies the general conditions under which an iterative algorithm of the form $\mathbf{u}_{t+1} \doteq \mathbf{u}_t + \alpha_t(\mathbf{b}_t - \mathbf{A}_t\mathbf{u}_t)$ converges, and substantiate that Differential TD(λ) converges with probability one to a vector $\mathbf{u}$ satisfying $\mathbf{A}\mathbf{u} = \mathbf{b}$, where $\mathbf{u}_0 = r(\pi), \mathbf{u}_{1:d} = \mathbf{w}$. The complete analysis is available in the full version of this paper. □

Differential TD(λ) is off-policy ready: Wan et al. (2021) showed that using the TD error to update the reward-rate estimate enables off-policy learning: when following a behavior policy b to evaluate the target policy π, multiplying an importance-sampling ratio $\rho_t = \pi(A_t|S_t)/b(A_t|S_t)$ to both the updates (2)–(3) results in an off-policy algorithm that is guaranteed to converge to a differential TD fixed point (defined in Section 2). This result carries over to the multi-step case when using tabular representations. In other words, tabular off-policy Differential TD(λ) is guaranteed to converge to a differential TD fixed point. In contrast, Tsitsiklis and Van Roy's (1999) Average-Cost TD(λ) is restricted to the on-policy setting. It updates the weight vector with (5) but uses a sample average of observed rewards to estimate the reward rate of the target policy: $R_{t+1} \doteq R_t + \eta\alpha(R_{t+1} - R_t)$. Such an update is not amenable to the off-policy setting.

4 Preliminary Empirical Results

In this short paper, we present preliminary empirical results designed to check (1) if Differential TD(λ) indeed converges to the fixed point predicted by the theory, and (2) if an intermediate value of the bootstrapping parameter λ works best within the bias-variance trade-off that it introduces. Additional experiments are presented in the full version of the paper (including an on-policy comparison with Tsitsiklis and Van Roy's Average-Cost TD(λ) as well as a demonstration of Differential TD(λ)'s convergence in the tabular off-policy setting).

Consider an MDP with N states and two actions—left and right—available in each state. The states are arranged in a chain (refer to the adjoining Figure 1) such that the left and right actions lead deterministically to the previous and next states respectively. The right action in the right-most state and the left action in the left-most state lead to the middle-most state with a reward of +1 and −1 respectively. All the other transitions result in zero reward.

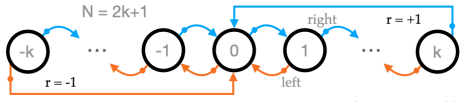

Figure 1: A continuing random-walk domain. See adjoining text for details.

For the on-policy experiment, we evaluated a policy that takes the right and left actions in each of the $N = 19$ states with equal probability. A one-hot representation of the states is observed by the agent—in this tabular case, the differential value function is completely representable. We applied Differential TD(λ) to this problem with about 800 combinations of its parameters α, η, λ for 100 independent runs of 10,000 time steps each. The weight vector and reward-rate estimates were all initialized to zero. We evaluated the quality of the reward-rate estimate by the squared error w.r.t. the true reward rate of the target policy; for value estimates, we computed the root mean squared error of the estimated value function w.r.t. the nearest solution of the Bellman equations (1). We follow Wan et al.'s terminology to denote the latter metric as 'RMSVE (TVR)' (for details, see Wan et al.'s (2021) Appendix C.4).

We observed that most values of λ led to convergence in the reward-rate and value error. Figure 2(left) shows learning curves for three values of λ corresponding to the parameters that resulted in the least value error averaged over the entire

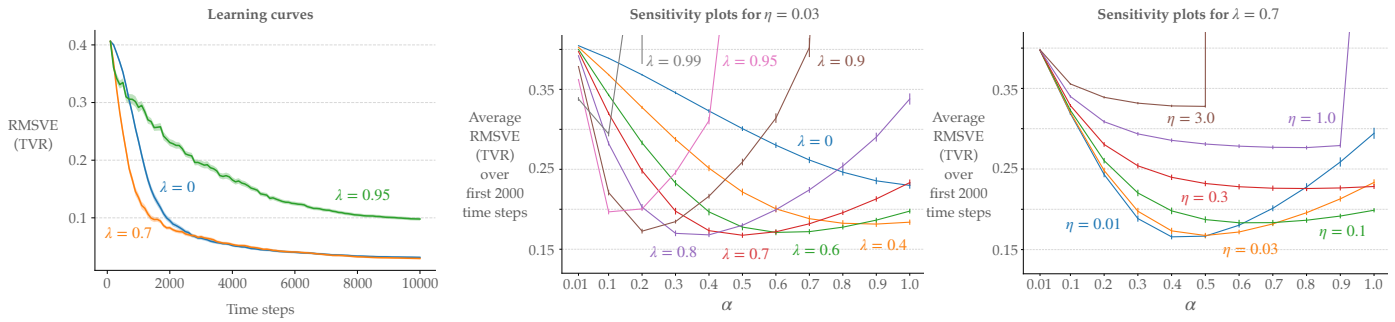

Figure 2: Learning curves and parameter studies for Differential TD(λ) on the continuing random-walk task. *Left:* Learning curves for three values of λ showing the value error (defined in the main text) converges towards zero at varying rates. *Center:* Parameter studies for a given η showing an intermediate value of λ achieves lower value error with a few samples for a larger range of α. *Right:* Parameter studies for a given λ showing the variation in performance across a wide range of η. The shaded region and the error bars denote one standard error computed over 100 independent runs.

training period (we only present plots for the value error; the error in the reward-rate estimate reliably went to zero as predicted by the theory). We saw that convergence was fast for small and intermediate values of lambda but unduly slow for large values—perhaps due to the large variance in updates caused by large traces. Figure 2(left) also shows that an intermediate value of λ led to a faster decrease in value error in the early part of training compared to $\lambda = 0$, which can be attributed to more efficient credit assignment using multi-step updates than one-step updates. This observation is further corroborated by Figure 2(center) which shows the average value error over the first 2000 time steps for different values of λ and α. For $\lambda = 0$, the value error remained high for a large range of α, while intermediate values of λ resulted in the lowest value error in this early stage of training. Larger values of λ were unstable for even small values of α. This particular sensitivity plot corresponds to a fixed value of $\eta (= 0.03)$; we observed similar trends across values of η.

Figure 2(right) shows sensitivity of Differential TD(λ)'s performance corresponding to the metric as above but this time for a fixed value of $\lambda (= 0.7)$ and different values of η and α. For a given λ, the performance seemed to depend on the choice of η, with intermediate values achieving good performance over a large range of α. Overall, these preliminary experiments show that Differential TD(λ) converges with one-hot representations—in line with the theory—and that an intermediate value of λ results in the most sample efficiency.

5 Conclusions and Future Work

This paper takes the first steps in extending the Differential family of average-reward algorithms from the tabular one-step case to that of multi-step with linear function approximation. We proposed the Differential TD(λ) algorithm, sketched its convergence theory, and demonstrated its effectiveness with preliminary experiments.

The most immediate direction of future work involves a comparison with Tsitsiklis and Van Roy's Average-Cost TD(λ), which is a proven-convergent multi-step on-policy average-reward algorithm. The main difference between Differential TD(λ) and Average-Cost TD(λ) lies in the reward-rate estimate's update. Apart from an empirical comparison in the on-policy case, this difference can also be characterized theoretically (e.g., in terms of variance of updates). Our ongoing work on both of these fronts will be presented in the full version of this paper.

In this paper we only briefly touched upon Differential TD(λ)'s off-policy capability—it needs to be fully explored. Unlike Average-Cost TD(λ), Differential TD(λ) is guaranteed to converge in the tabular off-policy case. But the general function approximation case requires more than importance-sampling-based 'posterior' corrections. In the discounted formulation, additional 'prior' corrections are performed by methods like GTD and ETD. Furthermore, several other ideas need to be extended to the average-reward case, including on-policy prediction methods like True Online TD(λ), off-policy prediction methods without importance-sampling like Tree Backup(λ), and control methods like Q(λ).

Reinforcement learning as a framework proposes *one* set of algorithmic ideas for artificial entities to learn to perform various tasks in their lifetimes. Related disciplines such as psychology and cognitive neuroscience aim to uncover *the* set of mechanisms underlying natural organisms' behavior. The literature suggests that the average-reward model as well as eligibility traces (independently) explain certain behavioral and neurophysiological data from animals. Such findings further encourage studying these ideas within reinforcement learning for developing general artificial intelligent systems.

References

Daw, N. D., & Touretzky, D. S. (2000). Behavioral considerations suggest an average reward TD model of the dopamine system. *Neurocomputing*.

Daw, N. D., & Touretzky, D. S. (2002). Long-Term Reward Prediction in TD Models of the Dopamine System. *Neural Computation*.

Gerstner, W., Lehmann, M., Liakoni, V., Corneil, D., & Brea, J. (2018). Eligibility Traces and Plasticity on Behavioral Time Scales: Experimental Support of NeoHebbian Three-Factor Learning Rules. *Frontiers in Neural Circuits*.

Naik, A., Shariff, R., Yasui, N., & Sutton, R. S. (2019). Discounted Reinforcement Learning Is Not an Optimization Problem. *Optimization Foundations for Reinforcement Learning Workshop at the Conference on Neural Information Processing Systems*. Also *ArXiv:1910.02140*.

Schwartz, A. (1993). A Reinforcement Learning Method for Maximizing Undiscounted Rewards. *ICML*.

Sutton, R. S. (1988). Learning to Predict by the Methods of Temporal Differences. *Machine Learning*.

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement Learning: An Introduction.* MIT Press.

Sutton, R. S., Modayil, J., Delp, M., Degris, T., Pilarski, P. M., White, A., Precup, D. (2011). Horde: A scalable real-time architecture for learning knowledge from unsupervised sensorimotor interaction. *AAMAS*.

Tsitsiklis, J. N., & Van Roy, B. (1999). Average cost temporal-difference learning. *Automatica*.

van Hasselt, H., & Sutton, R. S. (2015). Learning to Predict Independent of Span. *ArXiv:1508.04582*.

Wan, Y., Naik, A., & Sutton, R. S. (2021). Learning and Planning in Average-Reward Markov Decision Processes. *ICML*.