Chaining Value Functions for Off-Policy Learning

Simon Schmitt,[1 2] **John Shawe-Taylor,**[2] **Hado van Hasselt**[1]
[1]DeepMind
[2]University College London, UK
suschmitt@google.com

Abstract

To accumulate knowledge and improve its policy of behaviour, a reinforcement learning agent can learn 'off-policy' about policies that differ from the policy used to generate its experience. This is important to learn counterfactuals, or because the experience was generated out of its own control. However, off-policy learning is non-trivial, and standard reinforcement-learning algorithms can be unstable and divergent. In this paper we discuss a novel family of off-policy prediction algorithms which are convergent by construction. The idea is to first learn on-policy about the data-generating behaviour, and then bootstrap an off-policy value estimate on this on-policy estimate, thereby constructing a value estimate that is partially off-policy. This process can be repeated to build a chain of value functions, each time bootstrapping a new estimate on the previous estimate in the chain. Each step in the chain is stable and hence the complete algorithm is guaranteed to be stable. Under mild conditions this comes arbitrarily close to the off-policy TD solution when we increase the length of the chain. Hence it can compute the solution even in cases where off-policy TD diverges. We prove that the proposed scheme corresponds to an iterative decomposition of the inverse key matrix. Empirically we evaluate the idea on challenging MDPs such as Baird's counter example and observe favourable results.

Value estimation is key to decision making and reinforcement learning (Sutton and Barto 2018). To accumulate knowledge and improve its policy of behaviour, an agent can estimate values *off-policy* corresponding to policies that differ from the policy used to generate the experience it learns from. This can be useful to learn counterfactuals, or because the experience was generated out of its own control. Indeed the applications of off-policy learning are manifold: learning to exploit while exploring as e.g. in ϵ-greedy, learning multiple policies concurrently (Sutton et al. 2011; Badia et al. 2020), for representation shaping (Jaderberg et al. 2017), to minimize costly mistakes (Hauskrecht and Fraser 2000) or to learn from demonstrations (Hester et al. 2018).

However, off-policy learning is non-trivial, because standard reinforcement-learning algorithms can be unstable: (Baird 1995) showed that off-policy TD predictions can diverge to infinity in what is now known as *Baird's MDP*. (Sutton and Barto 2018) attribute this to the popular combination of function approximation (to support large state spaces) and bootstrapping (to reduce variance) in the off-policy context since called the *deadly triad*. Both are essential and ubiquitous in deep reinforcement learning (van Hasselt et al. 2018) hence algorithms that are convergent even in the face of the deadly triad are a prominent research direction.

Over the years, several variants and solutions have been proposed (Sutton et al. 2009; Maei 2011; van Hasselt, Mahmood, and Sutton 2014; Sutton, Mahmood, and White 2016), but these do not uniformly outperform off-policy TD (Hackman 2013) and sometimes suffer from high (even infinite) variance (Sutton, Mahmood, and White 2016).

In this paper we analyze a novel family of off-policy prediction algorithms that is convergent (i.e. breaks the deadly triad) and conceptually simple. The idea is to first learn on-policy about the data-generating behaviour, and then bootstrap an off-policy value estimate on this on-policy estimate, thereby constructing a value estimate that is partially off-policy. This process can be repeated to build a chain of value functions, each time bootstrapping a new estimate on the previous estimate in the chain. Each step in the chain is stable and hence the complete algorithm is guaranteed to be stable. When employing off-policy TD at each step in the chain we call it *chained TD* learning. While off-policy TD sometimes diverges and is unable to obtain its own solution (fixed point) we prove that chained TD always converges and that its solution comes arbitrarily close to the off-policy TD solution under mild conditions when we increase the length of the chain.

Interestingly our approach can be interpreted as estimating the value of following the target policy for a finite number of steps k and then following the behaviour indefinitely. For TD learning – contrary to estimating the target value directly – this is guaranteed to be stable as we prove in this paper.

While in practice we use a finite number of value functions we also consider what happens if $k \to \infty$ and use this to acquire insights into the convergence of the popular – albeit different – technique of *target networks* (Mnih et al. 2015).

We prove convergence of the expected chained TD update with a single learning rate and empirically confirm it on Baird's counter example that we augment to include rewards, where TD, TDC, GTD2 and ETD either diverge or make little progress.

1 Background

We consider state values $v(s)$ that are parameterised by parameter vector θ—for instance the weights of a neural network. The goal is to approximate the true value of each state s under target policy π, as defined by

$$\begin{aligned} v_\pi(s) &:= \mathbb{E}\left[\sum_{i=0}^{\infty} \gamma^i R_{t+i+1} \mid S_t = s\right] \\ &= \mathbb{E}\left[R_{t+1} + \gamma v(S_{t+1}) \mid S_t = s\right] . \end{aligned}$$

Off-policy TD (Sutton and Barto 2018) is an iterative process

$$\theta_{t+1} := \theta_t + \alpha \rho_t \left[R_t + \gamma v(S_{t+1}) - v(S_t)\right] \nabla_{\theta_t} v(S_t) \quad (1)$$

where each update aims to improve the parameters θ_t such that the new estimate $v_{\theta_{t+1}}$ on average gets closer to the target value v_π, even when following a different policy μ. Here α is the step-size, γ is the discount and R_t is the reward observed when transitioning from state S_t to S_{t+1} after executing action $A_t \sim \mu(A_t|S_t)$. In update (1), $\rho_t := \pi(A_t|S_t)/\mu(A_t|S_t)$ is the *importance-sampling ratio* between the probability of selecting action A_t under the target policy π and under the behaviour policy μ – not to be confused with the spectral radius of a matrix $\rho\left(\mathbf{M}\right)$. Unfortunately, when using function approximation, convergence of this algorithm can only be guaranteed in the on-policy setting where $\pi = \mu$ (Baird 1995; Sutton and Barto 2018).

This is an actively pursued research area where a series of solutions have been proposed (Sutton et al. 2009; Maei 2011; van Hasselt, Mahmood, and Sutton 2014; Sutton, Mahmood, and White 2016), but these often suffer from either performing worse than off-policy TD when it does not diverge (Hackman 2013) or even from infinite variance (Sutton, Mahmood, and White 2016). Our approach is similar in spirit to (De Asis et al. 2020) that estimate a new kind of return: fixed horizon returns (i.e. the rewards only from the next k steps) instead of the typical discounted return. This special return can also be estimated through a series of value functions, is guaranteed to converge albeit to a different fixed point.

2 Chaining Off-Policy Predictors

We want an off-policy algorithm that is 1) stable (i.e., convergent) and 2) with low bias with respect to the true values v_π. To this extend we propose a novel family of algorithms and show that it satisfies these desiderata.

Starting with the behaviour value

$$v^0 := v_\mu$$

the idea is to define a series of value functions $\{v^k\}_{k\in\mathbb{N}_0}$ recursively such that they approach the desired target value:

$$\lim_{k\to\infty} v^k \to v_\pi$$

This is achieved recursively by employing an off-policy estimator OPE such as off-policy TD learning that estimates v^{k+1} by bootstrapping off the previous value v^k:

$$v^{k+1} := \mathbb{E}_{\tau\sim\mu}\left[\mathrm{OPE}(\pi, v^k, \tau, \mu)\right]$$

This principle can be applied to any off-policy estimator that employs trajectories τ sampled from μ and a bootstrap value v^k to predict the values of target policy π e.g. $v^{k+1}(s) := \mathbb{E}_{\tau\sim\mu}\left[\rho_t(R_t + \gamma v^k(S_{t+1})) \mid S_t = s\right]$.

The idea of chaining off-policy estimators has a natural interpretation: v_μ is the value of the behaviour policy μ and v^k has the value of at first performing k steps according to the target policy π and then following μ indefinitely. As k increases the value becomes more and more off-policy and v^0 ultimately becomes irrelevant. This perspective illustrates that $v^k \to v_\pi$ as k increases (i.e. that v^k becomes unbiased). We analyse bias and convergence in the following section.

If estimated *sequentially* convergence is guaranteed by induction: $v^0 := v_\mu$ can be estimated on-policy, and hence TD is *stable* (i.e. converges). Then, for each $k > 0$, v^k is stable because it bootstraps off a stable v^{k-1}. In the next section we prove that convergence is also guaranteed for chained off-policy learning if all parameters are updated *concurrently*, e.g., when learning all value functions online.

A concrete stochastic update for each such value function when transitioning from state S_t to S_{t+1} and observing a reward R_{t+1} is given by

$$\theta_{t+1}^{k+1} := \theta_t^{k+1} + \alpha \rho_t \delta_t^k \nabla_{\theta_t^k} v_t^k(S_t) , \qquad (2)$$

where θ_t^k are the parameters of the k^{th}value function after observing t transitions , $v_t^k(s) := v_{\theta_t^k}(s)$ and

$$\delta_t^k := R_{t+1} + \gamma v_t^k(S_{t+1}) - v_t^{k+1}(S_t) .$$

We call this *chained (off-policy) TD* learning. *Sequential chained TD* only updates θ^k on timesteps after the previous θ^{k-1} has converged, while *concurrent chained TD* updates all $\{\theta^k\}_k$ at each timestep (see Algorithm 1).

In the next sections we analyse this algorithm theoretically (Section 3) and empirically (Section 4).

3 Analysis

To analyse Algorithm 1 in this section we consider linear function approximation, so that $v_\theta(s) = \theta^\top \phi(S_t)$, where $\phi(S_t)$ are the *features* observed at time t. We recall that off-policy TD sometimes diverges and is unable to obtain its own solution (fixed point) $\theta_\pi := \mathbf{A}_\pi^{-1}\mathbf{b}_\pi$, then we show that chained TD is always convergent and can compute θ_π under mild conditions via the following steps:

1. Section 3.2 observes that sequentially chained TD defines a recursion of fixed points: The fixed point θ_*^k of value function v^k can be computed from θ_*^{k-1}.
2. Section 3.3 shows that this recursion approaches the off-policy solution under mild conditions: $\lim_{k\to\infty} \theta_*^k = \theta_\pi$.
3. Section 3.4 proves convergence of both expected sequential and concurrent chained TD to the fixed points: $\lim_{t\to\infty} \theta_t^k = \theta_*^k$.

Hence chained TD is convergent for any fixed k and the attained fixed points of the k^{th}value function θ_*^k indeed approaches the off-policy TD solution $\theta_\pi := \mathbf{A}_\pi^{-1}\mathbf{b}_\pi$ under mild conditions that we investigate further in 3.3. Then

Algorithm 1: **Sequential chained TD** is described below. **Concurrent chained TD** is obtained by moving line 2 between line 6 and 7. Note that T needs to be specified large enough to ensure convergence.

```
Input: $\pi$, $\mu$, number of chains $K$, number of update steps $T$
Parameter: step size $\alpha$
 1: Initialize all $\{\theta^k\}_{k\in\mathbb{Z}.k\leq K}$ randomly, $t \leftarrow 0$.
 2: for $k \leftarrow 0$ to K do
 3:     for $i \leftarrow 1$ to T do
 4:         $t \leftarrow t + 1$
 5:         Play one action $A_t$ with $\mu$.
 6:         Observe next state $S_{t+1}$ and reward $R_{t+1}$.
 7:         if $k = 0$ then
 8:             $\delta \leftarrow R_{t+1} + \gamma v_{\theta^0}(S_{t+1}) - v_{\theta^0}(S_t)$; $\rho \leftarrow 1$
 9:         else
10:             $\delta \leftarrow R_{t+1} + \gamma v_{\theta^{k-1}}(S_{t+1}) - v_{\theta^k}(S_t)$
11:             $\rho \leftarrow \frac{\pi(A_t|S_t)}{\mu(A_t|S_t)}$
12:         end if
13:         $\theta^k \leftarrow \theta^k + \alpha\rho\delta\nabla_\theta v^k(S_t)$
14:     end for
15:     $\theta^{k+1} \leftarrow \theta^k$    ▷ Only used in sequential chained TD.
16: end for
17: return $\{\theta^k_t\}_{k\in\mathbb{Z}.k\leq K}$
```

chained TD learning is unbiased wrt. θ_π in the limit: i.e. $\lim_{k\to\infty}\theta^k_* = \theta_\pi$.

Off-policy TD and chained TD can be analysed through their expected updates which can be written in matrix form. For off-policy TD (Equation (1)) we obtain:

$$\theta_{t+1} = \theta_t + \alpha\left(\mathbf{b}_\pi - \mathbf{A}_\pi\theta_t\right) \tag{3}$$

and the expected update for chained TD (Equation (2)) is

$$\theta^{k+1}_{t+1} = \theta^{k+1}_t + \alpha\left(\mathbf{b}_\pi + \gamma\mathbf{Y}\theta^k_t - \mathbf{X}\theta^{k+1}_t\right)\,. \tag{4}$$

with

$$\mathbf{b}_\pi := \mathbb{E}_\mu\left[\rho_t R_t\phi(S_t)\right], \mathbf{b}_\mu := \mathbb{E}_\mu\left[R_t\phi(S_t)\right], \tag{5}$$

$$\mathbf{A}_\pi := \mathbb{E}_\mu\left[\rho_t\phi(S_t)\left(\phi(S_t)^\top - \gamma\phi(S_{t+1})^\top\right)\right] \tag{6}$$

$$= \Phi^\top\mathbf{D}_\mu(I - \gamma\mathbf{P}_\pi)\Phi = \mathbf{X} - \gamma\mathbf{Y} \tag{7}$$

$$\mathbf{X} := \mathbb{E}_\mu\left[\rho_t\phi(S_t)\phi(S_t)^\top\right] = \Phi^\top\mathbf{D}_\mu\Phi \tag{8}$$

$$\mathbf{Y} := \mathbb{E}_\mu\left[\rho_t\phi(S_t)\phi(S_{t+1})^\top\right] = \Phi^\top\mathbf{D}_\mu\mathbf{P}_\pi\Phi \tag{9}$$

where Φ is the state-feature matrix, $\mathbf{P}_\pi$ is π's transition matrix and $\mathbf{D}_\mu$ is a diagonal matrix with μ's steady-state distribution, $\mathbf{A}_\pi$ is called the *key matrix.*

3.1 Viewing Expected TD as Richardson Iteration

Expected TD (see equation (3)) can be viewed as Richardson Iteration (Richardson 1911) which is a simple and well-studied iterative algorithm that given $\mathbf{M}$ and $\mathbf{b}$ converges to $\theta^* = \mathbf{M}^{-1}\mathbf{b}$ under the condition that all eigenvalues of $\mathbf{M}$ are positive. Rather than inverting $\mathbf{A}_\pi$ expected TD learning attempts to determine the solution of $\mathbf{A}_\pi\theta_\pi = \mathbf{b}_\pi$ iteratively through Richardson Iteration and may diverge even though $\mathbf{A}_\pi$ is invertible.

Definition 1. *Given a square matrix* $\mathbf{M}$*, vector* $\mathbf{b}$ *and step-size* α *Richardson Iteration computes:*

$$\theta_{t+1} = \theta_t + \alpha\left(\mathbf{b} - \mathbf{M}\theta_t\right) \tag{10}$$

Definition 2. *We call Richardson Iteration* stable *if* $\lim_{t\to\infty}\theta_t$ *converges.*

Proposition 1. *Let* θ_1 *be any initial value,* $\mathbf{M}$ *any square matrix with only positive eigenvalues,* $\mathbf{b}$ *and vector then Richardson Iteration* θ_{t+1} *converges to* $\theta^* = \mathbf{M}^{-1}\mathbf{b}$ *for a sufficiently small step size* α*.*

Proof. Let $r_t = \theta_t - \theta^*$, then

$$\begin{aligned} r_{t+1} &= \theta_t + \alpha\left(\mathbf{b} - \mathbf{M}\theta_t\right) - \theta^* = \theta_t + \alpha\left(\mathbf{M}\theta^* - \mathbf{M}\theta_t\right) - \theta^* \\ &= \left(I - \alpha\mathbf{M}\right)r_t = \left(I - \alpha\mathbf{M}\right)^t r_0 \end{aligned} \tag{11}$$

Since $\mathbf{M}$ has positive eigenvalues we can pick α such that $I - \alpha\mathbf{M}$ satisfies $|\lambda_i| < 1.0$ for all eigenvalues λ_i. Furthermore we can diagonalize $I - \alpha\mathbf{M} = \mathbf{E}\mathbf{\Lambda}\mathbf{E}^{-1}$ such that $\left(I - \alpha\mathbf{M}\right)^k = \mathbf{E}\mathbf{\Lambda}^t\mathbf{E}^{-1}$. Since all entries of $\mathbf{\Lambda}$ have absolute value smaller than 1.0 convergence is ensured $\|\theta_t - \theta\|_2 = \|r_t\|_2 \to 0$ for $t \to \infty$ and any $\mathbf{b}$. □

3.2 Fixed Point Recursion

The expected update of sequential chained TD (4) can also be seen as Richardson Iteration. Once the k^{th}value function is estimated and θ^k is fixed, the chained TD update for the next value and its parameters θ^{k+1} converges to a fixed point θ^{k+1}_* that depends on θ^k:

$$\theta^{k+1}_*(\theta^k) := \lim_{t\to\infty}\theta^{k+1}_t = \mathbf{X}^{-1}(\gamma\mathbf{Y}\theta^k + \mathbf{b}_\pi) \tag{12}$$

convergence follows by Proposition 1 for a sufficiently small step-size α because $\mathbf{X}$ is positive-definite. Should θ^{k+1} bootstrap on the fixed point of a previous value θ^k_*, we obtain a recursion of fixed points:

$$\theta^{k+1}_* = \mathbf{X}^{-1}(\gamma\mathbf{Y}\theta^k_* + \mathbf{b}_\pi) \tag{13}$$

3.3 Bias

The established fixed point recursion (13) can be interpreted as a transformation of the unstable off-policy TD inverse problem ("determine $\theta_\pi = \mathbf{A}_\pi^{-1}\mathbf{b}_\pi$") where Richardson Iteration and hence TD diverge into a recursive sequence of stable sub-problems ("given θ^k_* determine θ^{k+1}_*") that are all stable under Richardson Iteration (see sections 3.2 and 3.4). In this section we prove under which conditions

$$\lim_{k\to\infty}\theta^k_* = \mathbf{A}_\pi^{-1}\mathbf{b}_\pi$$

i.e. that the sequence of fixed points converges to the off-policy TD solution θ_π as k increases.

Proposition 2. *Let* θ^k_* *denote the fixed point of the* k^{th} *chained value function defined as Eq.* (13)*. Its bias (distance to the TD off-policy solution* $\theta_\pi := \mathbf{A}_\pi^{-1}\mathbf{b}_\pi$*) is then given by* $\theta^k_* - \theta_\pi = \gamma^k\left(\mathbf{X}^{-1}\mathbf{Y}\right)^k\left(\theta^0 - \mathbf{A}_\pi^{-1}\mathbf{b}_\pi\right)$ *for any initial value* θ^0*.*

Proof. Given any θ^0 (e.g. without loss of generality the fixed point θ_μ of the on-policy algorithm estimating v_μ), the sequence (13) can be written in closed form as:

$$\theta_*^k = \underbrace{\sum_{i=0}^{k-1} \left(\mathbf{X}^{-1}\mathbf{Y}\gamma\right)^i}_{\mathbf{W}_k} \mathbf{X}^{-1}\mathbf{b}_\pi + \left(\mathbf{X}^{-1}\mathbf{Y}\gamma\right)^k \theta^0 \qquad (14)$$

Since $\mathbf{W}_k$ is a geometric series wrt. $\mathbf{X}^{-1}\mathbf{Y}\gamma$ it satisfies:

$$\begin{aligned}\mathbf{I} - \left(\mathbf{X}^{-1}\mathbf{Y}\gamma\right)^k &= \underbrace{\sum_{i=0}^{k-1} \left(\mathbf{X}^{-1}\mathbf{Y}\gamma\right)^i}_{\mathbf{W}_k} \left(\mathbf{I} - \mathbf{X}^{-1}\mathbf{Y}\gamma\right) \\ &= \mathbf{W}_k\mathbf{X}^{-1}\left(\mathbf{X} - \gamma\mathbf{Y}\right) \\ &= \mathbf{W}_k\mathbf{X}^{-1}\mathbf{A}_\pi\end{aligned}$$

Hence

$$\mathbf{W}_k\mathbf{X}^{-1} = \mathbf{A}_\pi^{-1} - \left(\mathbf{X}^{-1}\mathbf{Y}\gamma\right)^k \mathbf{A}_\pi^{-1} \qquad (15)$$

Plugging this into the closed form for θ_*^k from equation (14):

$$\begin{aligned}\theta_*^k &= \mathbf{W}_k\mathbf{X}^{-1}\mathbf{b}_\pi + \left(\mathbf{X}^{-1}\mathbf{Y}\gamma\right)^k \theta^0 \\ &= \mathbf{A}_\pi^{-1}\mathbf{b}_\pi - \left(\mathbf{X}^{-1}\mathbf{Y}\gamma\right)^k \mathbf{A}_\pi^{-1}\mathbf{b}_\pi + \left(\mathbf{X}^{-1}\mathbf{Y}\gamma\right)^k \theta^0 \\ &= \mathbf{A}_\pi^{-1}\mathbf{b}_\pi + \underbrace{\gamma^k \left(\mathbf{X}^{-1}\mathbf{Y}\right)^k \left(\theta^0 - \mathbf{A}_\pi^{-1}\mathbf{b}_\pi\right)}_{\textbf{Bias wrt. } \boldsymbol{\theta_\pi}} \qquad (16)\end{aligned}$$

□

Proposition 3. *Let θ_*^k denote the fixed point of the k^{th} chained value function defined as Eq.* (13)*. The fixed point of $\theta_*^\infty := \lim_{k\to\infty} \theta_*^k$ is equal to $\theta_\pi := \mathbf{A}_\pi^{-1}\mathbf{b}_\pi$ (i.e. $\theta_*^\infty = \theta_\pi$) for any initial value θ^0 if $\rho\left(\mathbf{X}^{-1}\mathbf{Y}\gamma\right) < 1$.*

Proof. $\rho\left(\mathbf{X}^{-1}\mathbf{Y}\gamma\right) < 1 \implies \lim_{k\to\infty} \|\left(\mathbf{X}^{-1}\mathbf{Y}\gamma\right)^k\|_2 = 0$ hence the bias vanishes as $k \to \infty$. □

While we will see in the next section that chained TD is always convergent for any fixed k, Proposition 2 allows us to analyze its distance to θ_π. For a fixed k the distance depends on θ^0. The distance can be greatly reduced should θ^0 already be close to the solution $\mathbf{A}_\pi^{-1}\mathbf{b}_\pi$. Hence a heuristic choice of θ^0 may be beneficial and without loss of generality we chose to use the behaviour value v_μ i.e. $\theta^0 = \mathbf{A}_\mu^{-1}\mathbf{b}_\mu$ which is always convergent independently of π and has recently been advocated with a single greedyfication step for offline RL (Gulcehre et al. 2021; Brandfonbrener et al. 2021).

Besides being always convergent (see next section), chained TD can even be unbiased wrt. θ_π if $\rho\left(\mathbf{X}^{-1}\mathbf{Y}\gamma\right) < 1$. In that case the bias in Eq. (16) reduces exponentially with k. Interestingly this is generally true, but not always. In Figure 1 we investigate how often the provably convergent chained TD is unbiased on random MDPs and observe that it is nearly always the case. On the other hand off-policy TD on the same MDPs diverges in roughly 20% of the cases.

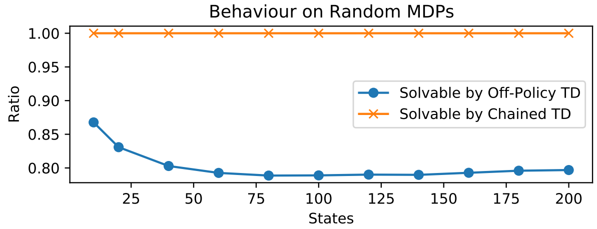

Figure 1: Off-policy TD sometimes diverges and is unable to obtain its own solution (fixed point). On the other hand chained TD always converges and often solves the off-policy TD problem to arbitrary precision for sufficiently large k – even in cases where off-policy TD diverges such as Baird's MDP. The difference between off-policy TD and chained TD becomes most apparent by looking at their worst case scenarios: Off-policy TD diverges for MDPs such as Baird's or the two-state MDP (Tsitsiklis and Van Roy 1997; Sutton, Mahmood, and White 2016). Chained TD obtains the target value in both MDPs. The latter can be modified such that chained TD becomes biased (see Section 3 of the appendix). However chained TD remains convergent i.e never diverges as we have proved. One may now ask how often each algorithm is able to solve the TD problem. We investigate this numerically by checking their relevant matrices $\mathbf{A}_\pi$ and $\mathbf{X}^{-1}\mathbf{Y}$ on random MDPs (sampling entries in Φ normal, rows of $\mathbf{P}_\pi$ and the diagonal of $\mathbf{D}_\mu$ uniformly and re-normalizing to sum to 1) with $\gamma = 0.99$ and as many features as states. We observe that chained TD solves the TD problem in nearly all cases while off-policy TD diverges in about 20% of the cases. Note that chained TD remains stable even when it does not solve the problem. Hence one may argue that the worst case scenario of chained TD is favourable.

3.4 Convergence

The previous section showed when the unstable off-policy-TD inverse problem $\theta_\pi = \mathbf{A}_\pi^{-1}\mathbf{b}_\pi$ can be decomposed into a recursive sequence of sub-problems ("given θ_*^k determine θ_*^{k+1}") that approach the off-policy TD solution ($\lim_{k\to\infty} \theta_*^k = \theta_\pi$): We will now show that each θ_*^{k+1} can be estimated through TD learning. To do this we prove that the corresponding Richardson Iterations converge. Later we will show that all θ_*^k can be determined concurrently, hence we do not need to wait until θ_t^k has converged before updating θ_t^{k+1}.

Sequential Estimation We call *Sequential Estimation* the process where each value function v^k bootstraps off the previous value function v^{k-1} only when the latter has converged. The resulting θ_*^{k-1} is then fixed and used as a TD bootstrap target in Equation (4) to estimate the next θ_*^k. Convergence can be proved by induction. Given a convergent initial value e.g. $\theta_*^0 := \theta_\mu$ or previous solution θ_*^{k-1} it remains to show that the induction step Equation (4) converges with now fixed bootstrap target θ_*^{k-1}. This update converges to θ_*^k by Proposition 1 even for unstable $\mathbf{A}_\pi$ because $\mathbf{X}$ is positive definite. Hence sequential estimation is convergent. In Figure 2 (left) we estimate a sequence of value functions with their expected

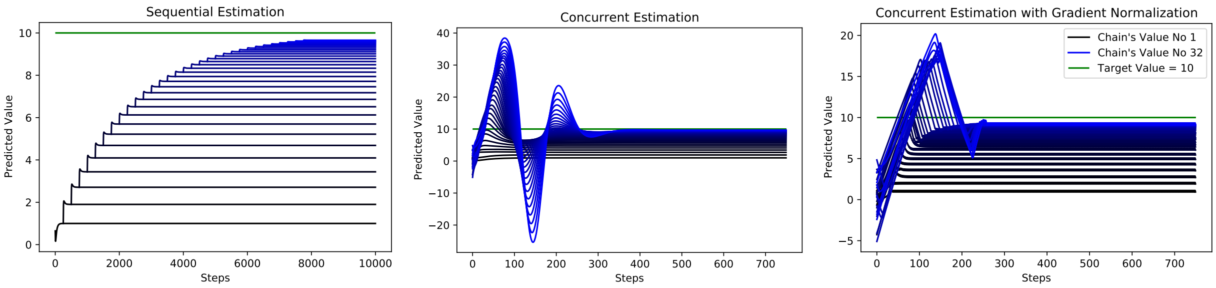

Figure 2: Various implementations (all with step-size $\alpha = 0.1$) of chained off-policy TD on Baird-Reward with discount 0.9 evaluated at state 8. Note that the target value at any state is $1/(1-\gamma) = 10$ and that all three displayed implementations approach the target-value as k increases. **Left:** Sequential Estimation. **Center:** Observe how Concurrent Estimation converges to the same correct results as Sequential Estimation with faster pace but with oscillations prior to reaching the target value. **Right:** Concurrent Estimation with gradient normalization. Note that the oscillations are reduced and that the predictions approach the target value.

update for $T = 250$ steps each and can observe convergence to the off-policy target value. For sequential estimation we use a strictly optional hot-start heuristic where after each 250 update steps we initialize the next θ_t^{k+1} with the previous solution θ^k to accelerate convergence.

Proposition 4. *Expected sequential estimation of Chained TD is convergent.*

Proof. Iterating Eq. (4) converges due to Proposition 1 for sufficiently small α because $\mathbf{X}$ is positive definite. □

Concurrent Estimation We call *Concurrent Estimation* the process where all value functions in the chain are updated simultaneously at each time step. In contrast to sequential training we do not assume that the previous value function in the chain has converged. This estimation may for example be more convenient for online learning, but requires a new proof of convergence. The proof works as follows: We will show that the matrix $\mathbf{M}$ (see (18)) – corresponding to the joint TD update of all parameters – has solely positive eigenvalues. Then viewing this expected concurrent update (see (17)) as Richardson Iteration implies the existence of a unique solution and convergence for a suitable step-size α.

In Figure 2 (center) we train a sequence of value functions with their expected concurrent update and observe convergence in accordance with the proposition below. We also observe oscillations in the value predictions in early training. This effect vanishes eventually as the parameters converge.

Nevertheless such oscillations may be inconvenient and their mitigation provides an interesting direction for future research. We present a simple mitigation technique of gradient normalization to reduce the pre-convergence oscillation magnitude in Figure 2 (right).

The Expected Concurrent Update of Chained TD The expected update of all chain parameters $\{\theta^k\}_{k\in\mathbb{Z}.k\leq K}$ can be written as a joint update in matrix form using one block structured update matrix $\mathbf{M}$.

$$\underbrace{\begin{bmatrix}\theta^0\\ \theta^1\\ \vdots\\ \theta^K\end{bmatrix}_{t+1}}_{\boldsymbol{\theta_{t+1}}} = \begin{bmatrix}\theta^0\\ \theta^1\\ \vdots\\ \theta^K\end{bmatrix}_t + \alpha\left(\underbrace{\begin{bmatrix}\mathbf{b}_\mu\\ \mathbf{b}_\pi\\ \vdots\\ \mathbf{b}_\pi\end{bmatrix}}_{\mathbf{b}^\dagger} - \mathbf{M}\underbrace{\begin{bmatrix}\theta^0\\ \theta^1\\ \vdots\\ \theta^K\end{bmatrix}_t}_{\boldsymbol{\theta_t}}\right) \tag{17}$$

with

$$\mathbf{M} := \begin{bmatrix}\mathbf{A}_\mu & & & \dots & \mathbf{0}\\ -\gamma\mathbf{Y} & \mathbf{X} & & & \vdots\\ & \ddots & \ddots & & \\ \vdots & & -\gamma\mathbf{Y} & \mathbf{X} & \\ \mathbf{0} & \dots & & -\gamma\mathbf{Y} & \mathbf{X}\end{bmatrix} \tag{18}$$

Fixed Point and Convergence of the Concurrent TD Update The formulation above allows us employ Richardson Iteration to analyze the convergence properties of all simultaneously changing parameters by investigating $\mathbf{M}$. As we will see $\mathbf{M}$ has only positive eigenvalues such that convergence to the unique solution $\boldsymbol{\theta_*} = \mathbf{M}^{-1}\mathbf{b}^\dagger$ follows. Contrary to GTD2 and TDC a single step-size suffices.

Proposition 5. $\mathbf{M}$ *has only positive eigenvalues.*

Proof. We make use of the fact that the eigenvalues of a triangular block matrix are the union of eigenvalues of the diagonal blocks. The diagonal blocks are $\mathbf{A}_\mu$ and $\mathbf{X}$. Since both are positive definite $\mathbf{M}$ has positive eigenvalues. □

Proposition 6. *The expected concurrent update has the same unique fixed point as the sequential update:* $\boldsymbol{\theta_*} = [\theta_\mu, \theta_*^1, \cdots, \theta_*^K]$.

Proof. From Proposition 5 it follows that $\mathbf{M}$ is invertible hence $\boldsymbol{\theta_*} = \mathbf{M}^{-1}\mathbf{b}^\dagger$ is the unique fixed point of the joint update. Block-wise solving $\mathbf{M}^{-1}\mathbf{b}^\dagger$ leads to an identical recursion as Equation (13) – the sequential fixed points. □

Proposition 7. *Expected concurrent chained TD is convergent. The expected update converges to the fixed point $\boldsymbol{\theta}_* = [\theta_\mu, \theta_*^1, \cdots, \theta_*^K]$ given a suitably small step-size.*

Proof. Convergence to $\boldsymbol{\theta}_* = \mathbf{M}^{-1}\mathbf{b}^\dagger$ follows from Proposition 5 (key matrix $\mathbf{M}$ has positive eigenvalues) and Proposition 1 (positive eigenvalues imply convergence). Then $\boldsymbol{\theta}_* = [\theta_\mu, \theta_*^1, \cdots, \theta_*^K]$ by Proposition 6. □

4 Empirical Study

In the previous sections we have shown that the expected update of chained TD is guaranteed to converge for sequential and concurrent parameter updates. Furthermore we have shown that it is unbiased wrt. θ_π under mild assumptions. In this section we empirically study how the corresponding stochastic update for chained TD converges on a selection of MDPs and observe favourable results.

We compare to regular off-policy TD and Emphatic Temporal Differences (ETD), and two forms of Gradient Temporal Difference Learning (GTD2 and TDC). All but the foremost are proven to be stable and have different trade-offs in practice. In our study we observe that ETD, GTD2 and TDC can suffer more from variance - and may even diverge for that reason - than chained TD if the discount is large $\gamma = 0.99$. However they converge faster if the discount is small $\gamma = 0.9$.

4.1 Methodology

While our method could also be applied offline, here we consider online off-policy learning where the stochastic update samples one transition at at time according to μ and then updates all parameters using temporal difference learning to estimate v_π. For chained-TD we bootstrap from the previous value function in the chain, while the first chain estimates v_μ with TD(0).

We consider three MDPs all with small discount of $\gamma = 0.9$ and large discount $\gamma = 0.99$ and evaluate algorithms according to the following experimental protocol: We evaluate the product of all relevant hyper-parameters for $100,000$ transitions and select the result with the lowest mean squared error averaged over the final 50% of transitions and over 10 seeds. We then select the best hyper-parameters and rerun the experiment with 100 new seeds. As hyper-parameters we consider all step-sizes α form the range $S = \{2^{-i/3} | i \in \{1, \ldots, 40\}\}$ (i.e. logarithmically spaced between 9.6×10^{-5} and 0.5), for GTD2 and TDC we also consider all secondary step-sizes β form the same range, for chained TD we consider chains of length 256 and evaluate the performance of only 9 indices $k \in I = \{2^i | i \in \{0, \ldots, 8\}\}$. This can be seen as a more efficient concurrent equivalent of experimenting with 9 different chain length separately. For sequential chained TD we split the training into windows of $T \in \{25, 50, 100, 200\}$ steps during which only one θ^k is estimated and all others kept unchanged. To prevent pollution from accidentally good initial values we initialize all parameters from a Gaussian distribution with $\sigma = 100$ such that errors at $t = 0$ are high.

4.2 Diagnostic Markov Decision Processes

Baird's MDP With and Without Rewards Baird's MDP is a classic example that demonstrates the divergence of off-policy TD with linear function approximation and has been used to evaluate the convergence of novel approaches. Originally proposed with a discount of $\gamma = 0.99$ it is often used with $\gamma = 0.9$, which results in lower variance updates. We consider both discounts. Furthermore we introduce a version of Baird's MDP with rewards as the rewards of the classic MDP are all 0. By introducing rewards we are able to investigate the bias of various convergent algorithms. To see why this interesting consider divergent off-policy TD with a large l2 regularization on θ. If the regularization is large enough it will push all parameters to 0, hence the value prediction will be 0 and match the target value of 0. This would be a stable but biased prediction if $v_\pi \neq 0$. To measure the bias we introduce rewards such that $v_\pi = \frac{1}{1-\gamma}$ (i.e 10 or 100) and $v_\mu = 0$ by rewarding each "solid" action with 1 and each "dashed" action with $-\frac{1}{6}$. We refer to this MDP as the *Baird-Reward MDP*.

The Threestate MDP Inspired by the Twostate MDP (Tsitsiklis and Van Roy 1997; Sutton, Mahmood, and White 2016) that demonstrates the divergence of off-policy TD concisely without rewards and with only two states, we propose the *Threestate MDP* with one middle state and two border states and two actions: "left" with -1 reward and "right" with 1 reward, leading to the corresponding neighbouring states or remaining if there is no further state in that direction. The starting state distribution is uniform. As with Baird-Reward introducing rewards permits us to measure the bias and convergence speed of various off-policy value predictors. We define $\Phi = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 2 & 2 & 1 \end{bmatrix}$ with full rank such that any state-value combination can be represented by a linear function. Hence any observed bias is entirely due to the evaluated algorithm. The target policy is "right" at all states while the behaviour is uniform. Again we consider $\gamma = 0.9$ and $\gamma = 0.99$ and observe that $v_\pi = \frac{1}{1-\gamma}$ and $v_\mu = 0$.

4.3 Experimental Results

Insights into 1-Step TD Estimators In Table 1 we evaluate popular TD off-policy value estimators on three MDPs each with two discounts ($\gamma = 0.9$ and $\gamma = 0.99$) and can observe that the larger discount is more challenging: Only sequential chained TD obtains an RMSE close to 0 on all MDPs and discounts.

Furthermore we provide learning curves for Baird-Reward with discount $\gamma = 0.99$ in Figure 3. Learning curves corresponding to all entries in the table can be found in the appendix.

At first we note that naive TD estimation (without off-policy correction) of v_μ is stable but its bias wrt. v_π is noticeable in MDPs with rewards (Baird-Reward and Threestate). It is desirable that an off-policy estimator is at least better than this naive baseline. However on Bairds MDP without rewards it inadvertently predicts the correct value, hence we invite the reader to focus on Baird-Reward and Threestate.

RMSE for MDP	Baird	Baird-Reward	Threestate	Baird	Baird-Reward	Threestate
with discount		$\gamma = 0.9$			$\gamma = 0.99$	
with reward	No	Yes	Yes	No	Yes	Yes
TD (no correction)	0.0	10.0	10.1	0.0	99.3	102.8
Off-Policy TD	div	div	div	div	div	div
ETD	0.0	div	0.0	136.7	div	div
GTD2	0.2	0.1	0.0	12.5	83.4	139.6
TDC	0.3	0.3	0.0	13.3	87.0	43.6
Concurrent Chained TD	0.0	0.4	0.1	0.0	72.6	77.9
Sequential Chained TD	0.0	0.0	0.0	0.0	0.0	0.2

Table 1: Evaluation of various 1-step TD algorithms on several MDPs. Observe that MDPs with large discount and rewards (Baird-Reward and Threestate) are the most challenging and that only sequentially chained TD learning obtains RMSE close to 0. Results with RMSE larger than 150 are considered divergent.

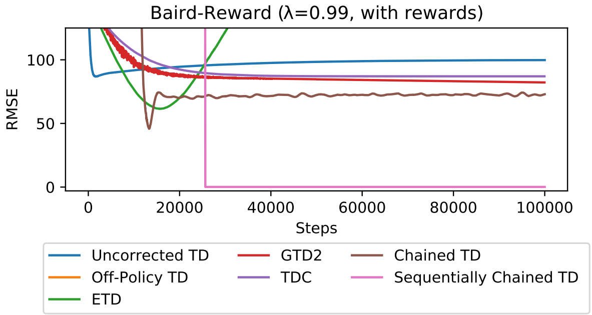

Figure 3: Learning process of the 1-step TD algorithms corresponding to Table 1 on Baird's MDP with rewards. Observe that chained TD learning reduces the RMSE most with only sequential chained TD learning reducing the error entirely. Off-policy TD has diverged. Sequential is at zero.

Next we observe that off-policy TD indeed either diverges or obtains a large error where divergence could be slowed down by a low learning rate.

ETD, GTD2 and TDC mostly fare well where the discount is small $\gamma = 0.9$. For $\gamma = 0.99$ ETD diverges on the MDPs with rewards. GTD2 and TDC obtain errors on Threestate of 139.6 and 43.6 respectively, on Baird-Reward they reduce the RMSE to 83.4 and 87.0.

Concurrent chained TD converges to the true value for small discounts $\gamma = 0.9$ and Baird irrespective of discount, while for large discount reducing the error to 72.6 and 77.9 on the challenging Baird-Reward and Threestate MDPs. Finally we observe that sequential chained TD converges close to the true value for all considered MDPs and discounts.

Chained N-step Estimators The principle of chaining value functions can also be applied to n-step estimators. N-step estimators predict the value of taking n steps with target policy and then following the policy corresponding to the bootstrap target. Chaining k such estimators results in a total prediction of $m = k \times n$ steps following π. This allows to predict v^m with fewer value functions.

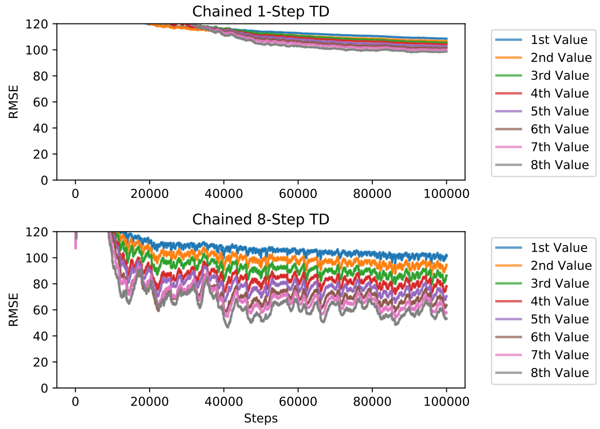

Figure 4: Convergence behaviour with increasing k for chains with chained 1-step (**top**) vs. chained 8-step off-policy TD (**bottom**). We present the RMSE of first eight k^{th}-values that are learned concurrently i.e. each bootstrapping off the previous value prediction. Observe how this leads to a sequence of increasingly better predictions. Finally note that the RMSE of the 8^{th} 8-step value prediction is lower on Threestate than the concurrent chained TD presented in Table 1 which only contains 1-step algorithms.

In Figure 4 we confirm this fact empirically on the Threestate MDP with $\gamma = 0.99$. One can see that a $(m = 8)$-step chain of length $k = 8$ attains a much lower RMSE than a $(m = 1)$-step chain of the same length. This suggests that n-step estimators may permit the use of shorter chains. Using importance sampling estimators to reduce the total length of the chain comes at the cost of increased variance. On the other hand it may come at the benefits of faster convergence and lower bias. Overall there is a bias, variance and computational complexity trade-off and n-step estimators allow to trade this off through the choice of n and k.

5 Conclusion

We present a novel family of off-policy value prediction algorithms that is convergent by construction. It works through chaining estimators that themselves do not need to be convergent. In particular we prove convergence of sequential and concurrent chained TD, which comes with the intuitive interpretation of estimating the value when following π for k steps and then following μ indefinitely.

Furthermore we provide an analytic formula for the bias of chained TD which can be used to derive three insights:

- Sequential chained TD is equivalent to TD with target networks that are switched slowly (i.e. once the current objective has converged) allowing us to compute the bias of such target-network TD and note $\rho\left(\mathbf{X}^{-1}\mathbf{Y}\gamma\right) < 1$ as the precise condition for its convergence.
- Sequential and concurrent chained TD are always convergent but may be biased, while off-policy TD may diverge and yield unbounded values when computing θ_π.
- Chained TD is unbiased wrt. θ_π in the theoretical limit of using infinitely many value functions if $\rho\left(\mathbf{X}^{-1}\mathbf{Y}\gamma\right) < 1$ e.g. on Baird's MDP where off-policy TD diverges.

Future work may be directed to investigate chaining other updates e.g. chained V-trace (Espeholt et al. 2018), chained Expected SARSA (van Seijen et al. 2009), chained Retrace (Munos et al. 2016) and to investigate the bias vs. variance trade-off of those chained estimators. For example better multi-step off-policy returns may lead to faster convergence. Chaining importance-sampling-free Q-learning can be used to estimate values off-policy even if no action probabilities were recorded. This may be useful to learn when the behaviour policy is unknown, e.g., from human demonstrations. Finally, for concurrent chaining, where all value functions in the chain are learned at the same time, the choice of which to select for acting may be taken at run-time, and potentially learnt, for example via bandits (Badia et al. 2020) or meta-gradients (Sutton 1992; Xu, van Hasselt, and Silver 2018).

Acknowledgements

We would like to thank Tom Zahavy and the anonymous AAAI 2022 reviewers for their valuable feedback.

References

Badia, A. P.; Sprechmann, P.; Vitvitskyi, A.; Guo, D.; Piot, B.; Kapturowski, S.; Tieleman, O.; Arjovsky, M.; Pritzel, A.; Bolt, A.; and Blundell, C. 2020. Never Give Up: Learning Directed Exploration Strategies. In *International Conference on Learning Representations*.

Baird, L. 1995. Residual Algorithms: Reinforcement learning with function approximation. In *Machine Learning: Proceedings of the Twelfth International Conference*, 30–37.

Brandfonbrener, D.; Whitney, W. F.; Ranganath, R.; and Bruna, J. 2021. Offline RL Without Off-Policy Evaluation. In *Thirty-Fifth Conference on Neural Information Processing Systems*.

De Asis, K.; Chan, A.; Pitis, S.; Sutton, R.; and Graves, D. 2020. Fixed-Horizon Temporal Difference Methods for Stable Reinforcement Learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04): 3741–3748.

Espeholt, L.; Soyer, H.; Munos, R.; Simonyan, K.; Mnih, V.; Ward, T.; Doron, Y.; Firoiu, V.; Harley, T.; Dunning, I.; Legg, S.; and Kavukcuoglu, K. 2018. IMPALA: Scalable Distributed Deep-RL with Importance Weighted Actor-Learner Architectures. In *Arxiv*.

Gulcehre, C.; Colmenarejo, S. G.; ziyu wang; Sygnowski, J.; Paine, T.; Zolna, K.; Chen, Y.; Hoffman, M.; Pascanu, R.; and de Freitas, N. 2021. Addressing Extrapolation Error in Deep Offline Reinforcement Learning.

Hackman, L. M. 2013. *Faster Gradient-TD Algorithms*. Master's thesis, University of Alberta.

Hauskrecht, M.; and Fraser, H. 2000. Planning treatment of ischemic heart disease with partially observable Markov decision processes. *Artificial Intelligence in Medicine*, 18(3): 221–244.

Hester, T.; Vecerik, M.; Pietquin, O.; Lanctot, M.; Schaul, T.; Piot, B.; Horgan, D.; Quan, J.; Sendonaris, A.; Osband, I.; Dulac-Arnold, G.; Agapiou, J.; Leibo, J.; and Gruslys, A. 2018. Deep Q-learning From Demonstrations.

Jaderberg, M.; Mnih, V.; Czarnecki, W. M.; Schaul, T.; Leibo, J. Z.; Silver, D.; and Kavukcuoglu, K. 2017. Reinforcement learning with unsupervised auxiliary tasks. *International Conference on Learning Representations*.

Maei, H. R. 2011. *Gradient temporal-difference learning algorithms*. Ph.D. thesis, University of Alberta.

Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A. A.; Veness, J.; Bellemare, M. G.; Graves, A.; Riedmiller, M.; Fidjeland, A. K.; Ostrovski, G.; Petersen, S.; Beattie, C.; Sadik, A.; Antonoglou, I.; King, H.; Kumaran, D.; Wierstra, D.; Legg, S.; and Hassabis, D. 2015. Human-level control through deep reinforcement learning. *Nature*, 518(7540): 529–533.

Munos, R.; Stepleton, T.; Harutyunyan, A.; and Bellemare, M. 2016. Safe and Efficient Off-Policy Reinforcement Learning. In Lee, D.; Sugiyama, M.; Luxburg, U.; Guyon, I.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc.

Richardson, L. F. 1911. The Approximate Arithmetical Solution by Finite Differences of Physical Problems Involving Differential Equations, with an Application to the Stresses in a Masonry Dam. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 210: 307–357.

Sutton, R. S. 1992. Adapting bias by gradient descent: An incremental version of delta-bar-delta. In *Proceedings of the Tenth National Conference on Artificial Intelligence*, 171–176. MIT Press.

Sutton, R. S.; and Barto, A. G. 2018. *Reinforcement Learning: An Introduction*. The MIT press, Cambridge MA.

Sutton, R. S.; Maei, H. R.; Precup, D.; Bhatnagar, S.; Silver, D.; Szepesvári, C.; and Wiewiora, E. 2009. Fast gradient-descent methods for temporal-difference learning with linear function approximation. In *Proceedings of the 26th Annual International Conference on Machine Learning (ICML 2009)*, 993–1000. ACM.

Sutton, R. S.; Mahmood, A. R.; and White, M. 2016. An Emphatic Approach to the Problem of Off-policy Temporal-Difference Learning. *Journal of Machine Learning Research*, 17(73): 1–29.

Sutton, R. S.; Modayil, J.; Delp, M.; Degris, T.; Pilarski, P. M.; White, A.; and Precup, D. 2011. Horde: A scalable real-time architecture for learning knowledge from unsupervised sensorimotor interaction. In *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, 761–768. International Foundation for Autonomous Agents and Multiagent Systems.

Tsitsiklis, J. N.; and Van Roy, B. 1997. An analysis of temporal-difference learning with function approximation. *IEEE Transactions on Automatic Control*, 42(5): 674–690.

van Hasselt, H.; Doron, Y.; Strub, F.; Hessel, M.; Sonnerat, N.; and Modayil, J. 2018. Deep Reinforcement Learning and the Deadly Triad. *CoRR*, abs/1812.02648.

van Hasselt, H.; Mahmood, A. R.; and Sutton, R. S. 2014. Off-policy TD(λ) with a true online equivalence. In *Uncertainty in Artificial Intelligence*.

van Seijen, H.; van Hasselt, H.; Whiteson, S.; and Wiering, M. 2009. A theoretical and empirical analysis of Expected Sarsa. In *Proceedings of IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning*, 177–184.

Xu, Z.; van Hasselt, H. P.; and Silver, D. 2018. Meta-Gradient Reinforcement Learning. In Bengio, S.; Wallach, H.; Larochelle, H.; Grauman, K.; Cesa-Bianchi, N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.