Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16)

Online Bellman Residual and Temporal Difference Algorithms with Predictive Error Guarantees

Wen Sun and J. Andrew Bagnell
Robotics Institute, Carnegie Mellon University, Pittsburgh, USA
{wensun, dbagnell}@cs.cmu.edu

Abstract

We establish connections from optimizing Bellman Residual and Temporal Difference Loss to worst-case long-term predictive error. In the online learning framework, learning takes place over a sequence of trials with the goal of predicting a future discounted sum of rewards. Our first analysis shows that, together with a stability assumption, any no-regret online learning algorithm that minimizes Bellman error ensures small prediction error. Our second analysis shows that applying the family of online mirror descent algorithms on temporal difference loss also ensures small prediction error. No statistical assumptions are made on the sequence of observations, which could be non-Markovian or even adversarial. Our approach thus establishes a broad new family of provably sound algorithms and provides a generalization of previous worst-case results for minimizing predictive error. We investigate the potential advantages of some of this family both theoretically and empirically on benchmark problems.

1 Introduction

Reinforcement learning (RL) is an online paradigm for optimal sequential decision making where a agent interacts with environments, takes actions, receives reward and tries to maximize its *long-term reward*, a discounted sum of all the rewards that will be received from now on. An important part of RL is policy evaluation, the problem of evaluating the expected long-term rewards of a fixed policy. *Temporal Difference* (TD) learning [Sutton, 1988] is perhaps the best known family of algorithms for policy evaluation. It has been observed that when combined with function approximation, TD may diverge and lead to poor prediction. The *Residual Gradient* (RG) was proposed [Baird, 1995] to address these concerns. RG attempts to minimize the *Bellman Error* (BE) (see definition in Sec. 2), typically with linear function approximation, using stochastic gradient descent. Comparison between the family of TD algorithms and RG has received tremendous attention, although most of the analyses heavily rely on certain stochastic assumptions of the environment such as that the sequence of observations are Markovian or from a static Markov Decision Process (MDP). For instance [Schoknecht and Merke, 2003] showed that TD converges provably faster than RG if the value functions are presented by tabular form. [Scherrer, 2010] shows that Bellman Residual minimization enjoys a guaranteed performance while TD does not in general when states are sampled from arbitrary distributions (off-policy) that may not correspond to trajectories taken by the system.

[Schapire and Warmuth, 1996] and [Li, 2008] provided worst-case analysis of long-term predictive error for variants of the linear TD and RG under a non-probabilistic online learning setting. Their results rely on the spectral analysis of a matrix that is related to **specific** update rules of the TD and RG algorithms under linear function approximation. Unfortunately, this approach makes it more difficult to extend their worst-case (assumption free) analysis to broader families of algorithms and representations that target Bellman and Temporal Difference errors.

Following [Schapire and Warmuth, 1996] and [Li, 2008]'s online learning framework, we present two simple, general connections between long-term predictive error and no-regret online learning that attempts to minimize BE and TD. The central idea is that methods such as TD and RG should be fundamentally understood as online algorithms as opposed to standard gradient methods, and that one cannot simultaneously make consistent predictions in the sense of TD and BE while doing a poor job in terms of long-run predictions. Similar to [Schapire and Warmuth, 1996] and [Li, 2008], our analysis does not rely on any statistical assumptions about the underlying system. This allows us to analyze difficult scenarios such as MDP with transition probabilities changing over time or even with each transition chosen entirely adversarial.

The main contribution of the paper is the analysis of the connections between online long-term reward prediction and no-regret online learning. Particularly, the first analysis on BE shows that any no-regret and *stable* [Ross and Bagnell, 2011] online learning algorithms, when targeting optimizing BE, ensure small prediction error. The second analysis focuses on TD and shows that when applying the family of *Online Mirror Descent* (OMD) on TD, we can also achieve small prediction error. We additionally show that Implicit Online Learning is another proper algorithm that can be used for optimizing TD to achieve small prediction error. These two analysis consequently suggests a broad new family of algorithms.

Particularly, our analysis on BE generalizes the RG algorithm from [Baird, 1995] in a sense that RG is a specific example of our family of algorithms that runs Online Gradient Descent (OGD) [Zinkevich, 2003] on a sequence of BE loss functions. For TD, our analysis generalizes TD$^*(0)$ from [Schapire and Warmuth, 1996] by showing that running OGD—a special form of OMD, reveals the update rule of TD$^*(0)$.

2 Preliminaries

We consider the sequential online learning model presented in [Schapire and Warmuth, 1996; Li, 2008] where no statistical assumptions about the sequence of observations are made. The sequence of the observations can either be Markovian as typically assumed in RL problem settings or even adversarial. We define the observation at time step t as $\mathbf{x}_t \in \mathbb{R}^n$, which usually represents the features of the environment at t. Throughout the paper, we assume that feature vector $\mathbf{x}$ is bounded as $\|\mathbf{x}\|_2 \leq X$. The corresponding reward at step t is defined as $r_t \in \mathbb{R}$, where we assume that reward is always bounded $|r| \leq R \in \mathbb{R}^+$. Given a sequence of observations $\{\mathbf{x}_t\}$ and a sequence of rewards $\{r_t\}$, the long-term reward at t is defined as $y_t = \sum_{k=t}^{\infty} \gamma^{k-t} r_k$, where $\gamma \in [0, 1)$ is a discounted factor. Given a function space $\mathcal{F}$ the learner chooses a predictor f at each time step from $\mathcal{F}$ for predicting long-term rewards. Throughout this paper, we assume that any prediction made by a predictor f at a state $\mathbf{x}$ is upper bounded as $|f(\mathbf{x})| \leq P \in \mathbb{R}^+$, for any $f \in \mathcal{F}$ and $\mathbf{x}$.

At time step $t = 0$, the learner receives $\mathbf{x}_0$, initializes a predictor $f_0 \in \mathcal{F}$ and makes prediction $\hat{y}_0$ of y_0 as $f_0(\mathbf{x}_0)$. Rounds of learning then proceeds as follows: the learner makes a prediction $\hat{y}_t$ of y_t at step t as $f_t(\mathbf{x}_t)$; the learner then observes a reward r_t and the next state $\mathbf{x}_{t+1}$; the learner updates its predictor to f_{t+1}. This interaction repeats and is terminated after T steps. Throughout this paper, we call this problem setting as *online prediction of long-term reward*.

We first define the *signed Bellman Error* at step t for predictor f_t as $b_t = f_t(\mathbf{x}_t) - r_t - \gamma f_t(\mathbf{x}_{t+1})$, which measures effectively how self consistent f_t is in its predictions between time step t and $t + 1$. We define the corresponding *Bellman Loss* at time step t with respect to predictor f as:

$$\ell_t^b(f) := (f(\mathbf{x}_t) - r_t - \gamma f(\mathbf{x}_{t+1}))^2. \tag{1}$$

We also define *signed Temporal Difference Error* (signed TD error) at step t for predictor f_t as $d_t = f_t(\mathbf{x}_t) - r_t - \gamma f_{t+1}(\mathbf{x}_{t+1})$. We define *TD* Loss* at step t as:

$$\ell_t^d(f) := (f(\mathbf{x}_t) - r_t - \gamma f_{t+1}(\mathbf{x}_{t+1}))^2. \tag{2}$$

The *Signed Prediction Error* of long-term reward at t for f_t is defined as $e_t = f_t(\mathbf{x}_t) - y_t$ and $e_t^* = f^*(\mathbf{x}_t) - y_t$ for f^* accordingly. We will typically be interested in bounding the *Prediction Error* (PE) e_t^2 of a given algorithm in terms of the best possible PE. To lighten notation in the following sections, all sums over time indices implicitly run from 0 to $T - 1$ unless explicitly noted otherwise.

3 Online Learning for Long-Term Reward Prediction

In this section, we first propose a new perspective of RG algorithm and TD* algorithm: we show that RG and TD* both could be understood as running Online Gradient Descent on Bellman loss ℓ_t^b and TD* loss ℓ_t^d, respectively.

At every time step t, after receiving the Bellman loss $\ell_t^b(f)$, let us apply OGD on $\ell_t^b(f)$:

$$f_{t+1} = f_t - \mu_t b_t (\nabla_f f_t(\mathbf{x}_t) - \gamma \nabla_f f_t(\mathbf{x}_{t+1})), \tag{3}$$

where we denote $\nabla_f f(\mathbf{x})$ as the functional gradient of the evaluation functional $f(\mathbf{x})$ at function f.[1]

Now for linear function approximation where $f(\mathbf{x})$ is represented as $\mathbf{w}^T\mathbf{x}$, the update step in Eq. 3 exactly reveals the RG algorithm proposed by [Baird, 1995].

Now, let us apply OGD to the TD* loss $\ell_t^d(f)$, we get the following update step:

$$f_{t+1} = f_t - \mu_t d_t \nabla_f f_t(\mathbf{x}_t). \tag{4}$$

Note that the above update rule is *implicit* in a sense that the Right Hand Side (RHS) and the Left Hand Side (LHS) both have f_{t+1} (d_t has f_{t+1}). To get the explicit update rule for f_{t+1}, one needs to solve f_{t+1} from Eq. 4. If we substitute the linear function approximation $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$ into Eq. 4 and solve for $\mathbf{w}_{t+1}$, one can exactly reveal the TD$^*(0)$ update rule proposed in [Schapire and Warmuth, 1996].[2]

Online Gradient Descent is one of the popular no-regret online learning algorithms. The above perspective suggests that RG and TD could be understood as applying a special no-regret online algorithm—OGD, to Bellman loss and TD* loss. A natural question that one would like to know is that whether any other no-regret online algorithms, such as Online Newton step [Hazan *et al.*, 2006], Online Frank Wolf [Hazan and Kale, 2012] and implicit online learning [Kulis *et al.*, 2010], can be applied to Bellman loss ℓ_t^b and TD* loss ℓ_t^d, and achieve similar guarantees on PE.

3.1 Optimizing Bellman Loss

In this section, we establish a connection between optimizing Bellman loss and worst case long-term predictive error. Particularly, we show that optimizing Bellman loss with any *stable* and *no-regret* online algorithms ensures small prediction error for long-term reward prediction.

We first define the stability condition:

Definition 3.1 Online Stability*: For the generated sequence of predictors f_t, we say the algorithm is online stable if:*

$$\lim_{T\to\infty} \frac{1}{T} \sum (f_t(\mathbf{x}_{t+1}) - f_{t+1}(\mathbf{x}_{t+1}))^2 = 0. \tag{5}$$

Intuitively, the online stability means that on average the difference between successive predictors is eventually small. That is, the difference between $f_t(\mathbf{x}_{t+1})$ and $f_{t+1}(\mathbf{x}_{t+1})$ is small on average. Online stability is a general condition and does not severely limit the scope of the online learning algorithms. For instance, when f is linear, the definition of stability of online learning in [Saha *et al.*, 2012] (see Eq. 3

[1] We assume the function $\nabla_f f(\mathbf{x})$ belongs to $\mathcal{F}$. This is true for function classes such as Reproducing Kernel Hilbert Space (RKHS).

[2] This is why we call $\ell_t^d(f)$ TD* loss, since with OGD, it reveals the TD$^*(0)$ algorithm, not the TD algorithm. However, nearly identical results can be established for TD loss which replaces f_{t+1} by f_t in ℓ_t^d, and classic TD can be recovered by OGD on TD loss.

in [Saha *et al.*, 2012]) and [Ross and Bagnell, 2011] implies our form of online stability. In fact, we can show that many popular no-regret online learning algorithms including OGD, ONS, OWF, implicit online learning, and FTRL satisfy our online stability condition. We refer reader to [Sun and Bagnell, 2015] for the detailed study of the online stability condition for the above mentioned no-regret online algorithms.

Define $\epsilon_t = f_t(x_{t+1}) - f_{t+1}(x_{t+1})$, with the online stability condition, we now ready to state the main theorem:

Theorem 3.2 *Assume a sequence of predictors $\{f_t\}$ is generated by running some online algorithm on the sequence of Bellman loss $\{\ell_t^b\}$. For any predictor $f^* \in \mathcal{F}$, the sum of prediction errors $\sum e_t^2$ can be upper bounded as:*

$$(1-\gamma)^2 \sum e_t^2 \leq 2\sum (b_t^2 - b_t^{*2}) + 2\gamma^2 \sum \epsilon_t^2 + 2(1+\gamma)^2 \sum e_t^{*2} + M, \quad (6)$$

where

$$M = 2(\gamma+\gamma^2)(e_0^{*2} - e_T^{*2}) - (\gamma^2-\gamma)(e_T^2 - e_0^2).$$

By running a no-regret and online stable algorithm, as $T \to \infty$, the average prediction error is then asymptotically upper bounded by a constant factor of the best possible prediction error in the function class:

$$\lim_{T\to\infty} \frac{\sum e_t^2}{T} \leq \frac{2(1+\gamma)^2}{(1-\gamma)^2} \frac{\sum e_t^{*2}}{T}. \quad (7)$$

The proof of the theorem only consists of easy application of telescoping tricks and Cauchy-Schwart inequalities. We refer readers to [Sun and Bagnell, 2015] for the detailed proof of the above theorem. We emphasize that the above analysis is independent of the particular form of function approximation.

When $e_t^* = 0, \forall t$, from Theorem. 3.2, it is easy to see that no-regret rate of $(1/T)\sum(b_t^2 - b_t^{*2})$ and the online stability rate of $(1/T)\sum \epsilon_t^2$ together determine the rate of the convergence of $(1/T)\sum e_t^2$. When $T \to \infty$ and $\gamma \to 1$ (specifically when $\gamma \geq (1/\sqrt{2})$), our upper bound analysis in Eq. 7 is asymptotically tighter than the upper bound in [Li, 2008] (Eq. 12) provided for RG. Since a large number of popular no-regret online algorithms satisfy the online stability condition, our theorem essentially expends the family of algorithms that can be used to learn predictors of long-term rewards.

We emphasize here that stability of online algorithms is essential for our results– the no-regret property can be shown by counter-example to be *insufficient* to achieve low predictive error [Sun and Bagnell, 2015].

3.2 Optimizing TD* Loss

The analysis in Sec. 3.1 is general enough such that almost any existing no-regret online learning algorithm can be used for optimizing Bellman loss and ensures small prediction error on long-term rewards. Though we wish such a nice generalization also exists for TD, we could not establish it. Instead we show that a broad family of online learning algorithms—Online Mirror Descent (OMD), when applied to TD* loss, ensures small prediction error similar in form to [Schapire and Warmuth, 1996]. We also show that implicit online gradient descent, a special form of implicit online learning, can also be used for optimizing TD* loss. The proofs of the theorems presented in this section are in the appendix.[3]

Online Mirror Descent for TD* loss

Let us define $R(f)$ as a regularization and assume that $R(f)$ is a both smooth and strongly convex function with respect to f with norm $\|\cdot\|$, defined by the inner product associated with $\mathcal{F}$ as $\|f\|^2 = \langle f, f\rangle$. A function $R(f)$ is α-smooth and β-strongly convex if and only if:

$$\frac{\beta}{2}\|f_t - f_{t+1}\|^2 \leq R(f_t) - R(f_{t+1}) - \nabla R(f_{t+1})(f_t - f_{t+1}) \leq \frac{\alpha}{2}\|f_t - f_{t+1}\|^2. \quad (8)$$

Without loss of generality, we assume that $R(f)$ is 1-strongly convex (otherwise simply scale it) and α-smooth function with respect to f with norm $\|\cdot\|$. For instance, when f is linear, $\|\mathbf{w}\|^2/2$ is 1-strongly convex and 1-smooth. When applying OMD on TD* loss, we have the following update rule, which we denote as **OMD-TD***:

$$f_t = \arg\min_f \langle f, \theta_t\rangle + \frac{1}{\mu} R(f); \quad (9)$$

$$\theta_{t+1} = \theta_t + (\hat{y}_t - r_t - \gamma\hat{y}_{t+1})\nabla_f f_t(\mathbf{x}_t). \quad (10)$$

Note that when we compute f_t using Eq. 9, the RHS of Eq. 9 actually implicitly depends on $\hat{y}_t$, which is equal to $f_t(\mathbf{x}_t)$ and hence depends on f_t. Here, we assume that though f_t appears on both sides of Eq. 9, we can still solve for f_t from Eq. 9 as TD$^*(0)$ does. In practice, whether or not we can solve f_t from Eq. 9 could depend on the form of $R(f)$. For instance, when $R(f) = \|f\|^2/2$ and f belongs to a *Reproducing Kernel Hilbert Space* (RKHS) (e.g., linear function $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$), we can achieve closed-form update of f_t. In fact, when $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$, $R(\mathbf{w}) = \|\mathbf{w}\|^2$, it is easy to show the update rule from Eq. 9 reveals the TD$^*(0)$ algorithm.

The following theorem shows optimizing TD* loss with OMD ensures small long-term prediction error:

Theorem 3.3 *With $\mu = O(\frac{1}{\sqrt{T}})$ and $\mathcal{F}$ being a RKHS, OMD-TD* (Eq. 9 and 10) has the following bound:*

$$\sum e_t^2 \leq \frac{2+2\gamma^2}{(1-\gamma)^2}\sum e_t^{*2} + O(\sqrt{T}). \quad (11)$$

For the average prediction error $\sum e_t^2/T$, we have:

$$\lim_{T\to\infty} \frac{\sum e_t^2}{T} \leq \frac{2+2\gamma^2}{(1-\gamma)^2}\frac{\sum e_t^{*2}}{T}. \quad (12)$$

Implicit Online Learning for TD* Loss

The OMD framework generalizes quite a few popular online algorithms such as Online Gradient Descent, Normalized Exponential Gradient (normalized EG), OGD with lazy projection and p-norm algorithm [Shalev-Shwartz, 2011]. However, OMD is conceptually different from another family of online algorithms—*Implicit Online Learning* [Kulis *et al.*, 2010]. Implicit online learning algorithms usually are more stable and robust compared to algorithms with explicit update rules.

[3] Available at http://www.cs.cmu.edu/~wensun

The idea of implicit update has been applied to classic TD [Tamar *et al.*, 2014], where the authors show the algorithm with implicit update is more stable than classic TD in a sense that it is not sensitive to learning step size.

Briefly, given the sequence of loss $\ell_t(f)$, implicit online learning updates f as $f_{t+1} = \arg\min_f \ell_t(f) + \frac{1}{\mu_t} D_R(f, f_t)$, where $D_R(f, f_t)$ is the Bregman divergence generated from regularization R. For special case where f is in RKHS, TD* loss $\ell_t^d(f)$ is actually a quadratic loss with respect to f. Hence, we propose to apply the implicit Online Gradient Descent—one special form of implicit online learning, to TD* loss. Set $R(f) = \|f\|^2/2$, we have the following update rule:

$$f_{t+1} = \arg\min_f \ell_t^d(f) + \frac{1}{\mu_t}\|f_t - f\|_2, \tag{13}$$

Note that the above update rule is implicit since $\hat{y}_{t+1}$ (buried in ℓ_t^d) depends f_{t+1}. Depending on the form of f, we can achieve closed-form solution for f_{t+1} from Eq. 13.

Below, we demonstrate a closed-form update rule for linear function $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$ with $R(\mathbf{w}) = \|\mathbf{w}\|^2$. Replace f with $\mathbf{w}$ in Eq. 13, take the derivative with respect to $\mathbf{w}$, set it to zero, and solve for $\mathbf{w}_{t+1}$, we will get:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \frac{\mu_t}{1 + \mu_t\|\mathbf{x}_t\|_2^2}(\mathbf{w}_t^T\mathbf{x}_t - r_t - \gamma\hat{y}_{t+1})\mathbf{x}_t. \tag{14}$$

Note that $\hat{y}_{t+1}$ implicitly depends on $\mathbf{w}_{t+1}$. To solve for $\mathbf{w}_{t+1}$, we first dot product $\mathbf{x}_{t+1}$ on both sides of the above equation (the LHS becomes $\hat{y}_{t+1}$), solve for $\hat{y}_{t+1}$ and then substitute $\hat{y}_{t+1}$ back to the equation and solve for $\mathbf{w}_{t+1}$. This gives us the following **Implicit-TD*** update step:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \frac{\mu}{1 + \mu\mathbf{x}_t^T(\mathbf{x}_t - \gamma\mathbf{x}_{t+1})} b_t\mathbf{x}_t, \tag{15}$$

where $b_t = (\mathbf{w}_t^T\mathbf{x}_t - r_t - \gamma\mathbf{w}_t^T\mathbf{x}_{t+1})$. The corresponding update rule for RKHS with kernel $K(\cdot,\cdot)$ is:

$$f_{t+1} = f_t - \frac{\mu}{1 + \mu K(\mathbf{x}_t, \mathbf{x}_t - \gamma\mathbf{x}_{t+1})} b_t K(\mathbf{x}_t, \cdot), \tag{16}$$

where $b_t = (f_t(\mathbf{x}_t) - r_t - \gamma f_t(\mathbf{x}_{t+1}))$.

Implicit-TD* has the following upper bound on PE:

Theorem 3.4 *With $\mu = O(\frac{1}{\sqrt{T}})$ and $\mathcal{F}$ being a RKHS, Implicit-TD* (Eq. 15 and 16) has the following bound:*

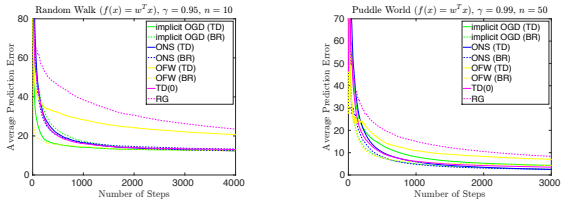
$$\sum e_t^2 \le \frac{(1+\gamma)^2(2+2\gamma^2)}{(1-\gamma)^2}\sum e_t^{*2} + O(\sqrt{T}). \tag{17}$$

For the average prediction error $\sum e_t^2/T$, we have:

$$\lim_{T\to\infty} \frac{\sum e_t^2}{T} \le \frac{(1+\gamma)^2(2+2\gamma^2)}{(1-\gamma)^2}\frac{\sum e_t^{*2}}{T} \tag{18}$$

3.3 Discussion

The bound of OMD-TD* is the tightest compared to Implicit-TD* and RG. Though our OMD-TD* bound is not as tight as the one from [Schapire and Warmuth, 1996], our analysis is more general. Our bound of RG is asymptotically tighter than the one from [Li, 2008] when $\gamma \to 1$. Experimentally we find that Implicit-TD* performs really well, which indicates that our worst-case bound for Implicit-TD* may be not tight.



(a) Random Walk (b) Puddle World

Figure 1: Convergence of prediction error. We applied a set of online algorithms on Bellman loss $\{\ell_t^b(\mathbf{w})\}$ (dot line) and TD*-loss functions $\{\ell_t^d(\mathbf{w})\}$ (solid line) for Random walk (left) and Puddle World (right).

4 Experiments

We applied several online learning algorithms to two simulated policy evaluation problems: (1) Random Walk with a ring chain, which is a variant of the Hall problem introduced in [Baird, 1995], (2) PuddleWorld adopted from [Sutton and Barto, 1998]. We tested several popular no-regret and stable online learning algorithms, including *implicit online gradient descent* (implicit OGD), *online Newton step* (ONS) [Hazan *et al.*, 2006], *online Frank Wolf* (OFW) [Hazan and Kale, 2012] and classic online gradient descent [Zinkevich, 2003], on both TD* loss and Bellman loss.

Fig. 1 shows the convergence of average prediction error with respect to number of time steps. We note that ONS and implicit OGD give good convergence speed in general. Throughout the experiments, we found that implicit OGD works well for both TD* loss and Bellman loss. Our experimental results also show that our approaches have the possibility to achieve smaller prediction error than TD(0) (e.g., Fig. 1b). Note that when optimizing TD* loss, ONS and OFW actually achieve good performance, though our analysis on TD* loss currently does not support ONS or OFW.

The experiment results for RKHS can be found at [Sun and Bagnell, 2015], where we also demonstrated these algorithms on a simulated helicopter hover domain [Coates *et al.*, 2008].

5 Conclusion

We introduced a new perspective for RG and TD—they could be understood as running special no-regret online algorithm on Bellman loss and TD* loss, respectively. This new perspective enables us to derive two generalizations, one for RG and one for TD* in the online setting, where no statistical assumptions are placed on the observations. Particularly, we show that any no-regret and stable online algorithms, when applied to Bellman loss, ensures small prediction error. For TD, we connect TD* to two family of online algorithms—Online Mirror Descent and Implicit Online Learning, and we show that optimizing TD* loss with OMD and implicit OGD guarantees small prediction error. The remaining open problem is that whether there exists a more general connection between TD* loss and no-regret online algorithms: when optimizing TD* loss, are the no-regret property and stability sufficient to achieve low prediction error?

References

[Baird, 1995] Leemon Baird. Residual algorithms: Reinforcement learning with function approximation. In *Proceedings of the 12th International Conference on Machine Learning (ICML 1995)*, pages 30–37, 1995.

[Coates *et al.*, 2008] Adam Coates, Pieter Abbeel, and Andrew Y Ng. Learning for control from multiple demonstrations. In *Proceedings of the 25th international conference on Machine learning (ICML 2008)*, pages 144–151, 2008.

[Hazan and Kale, 2012] Elad Hazan and Satyen Kale. Projection-free Online Learning. In *29th International Conference on Machine Learning (ICML 2012)*, pages 521–528, 2012.

[Hazan *et al.*, 2006] Elad Hazan, Amit Agarwal, and Satyen Kale. Logarithmic regret algorithms for online convex optimization. In *Proceedings of the 19th annual conference on Computa- tional Learning Theory (COLT 2006)*, pages 169–192, 2006.

[Kulis *et al.*, 2010] Brian Kulis, Peter L Bartlett, Bartlett Eecs, and Berkeley Edu. Implicit Online Learning. In *Proceedings of the 27th international conference on Machine learning (ICML 2010)*, pages 575–582, 2010.

[Li, 2008] Lihong Li. A worst-case comparison between temporal difference and residual gradient with linear function approximation. In *Proceedings of the 25th international conference on Machine learning (ICML 2008)*, pages 560–567, 2008.

[Ross and Bagnell, 2011] Stephane Ross and J. Andrew Bagnell. Stability Conditions for Online Learnability. *arXiv:1108.3154*, 2011.

[Saha *et al.*, 2012] Ankan Saha, Prateek Jain, and Ambuj Tewari. The Interplay Between Stability and Regret in Online Learning. *arXiv preprint arXiv:1211.6158*, pages 1–19, 2012.

[Schapire and Warmuth, 1996] Robert E. Schapire and Manfred K. Warmuth. On the worst-case analysis of temporal-difference learning algorithms. *Machine Learning*, 22(1):95–121, 1996.

[Scherrer, 2010] Bruno Scherrer. Should one compute the Temporal Difference fix point or minimize the Bellman Residual? The unified oblique projection view. *International Conference on Machine Learning (ICML 2010)*, 2010.

[Schoknecht and Merke, 2003] Ralf Schoknecht and Artur Merke. TD(0) Converges Provably Faster than the Residual Gradient Algorithm. In *International Conference on Machine Learning (ICML 2003)*, pages 680–687, 2003.

[Shalev-Shwartz, 2011] Shai Shalev-Shwartz. Online Learning and Online Convex Optimization. *Foundations and Trends in Machine Learning*, 4(2):107–194, 2011.

[Sun and Bagnell, 2015] Wen Sun and J. Andrew (Drew) Bagnell. Online Bellman Residual Algorithms with Predictive Error Guarantees. In *The 31st Conference on Uncertainty in Artificial Intelligence (UAI 2015)*, July 2015.

[Sutton and Barto, 1998] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.

[Sutton, 1988] R S Sutton. Learning to Predict by the Methods of Temporal Difference. *Machine Learning*, pages 9–44, 1988.

[Tamar *et al.*, 2014] Aviv Tamar, Panos Toulis, Shie Mannor, and Edoardo M. Airoldi. Implicit Temporal Differences. *arXiv:1412.6734*, pages 1–6, 2014.

[Zinkevich, 2003] Martin Zinkevich. Online Convex Programming and Generalized Infinitesimal Gradient Ascent. In *International Conference on Machine Learning (ICML 2003)*, pages 421–422, 2003.