Temporal Difference Methods for the Variance of the Reward To Go

Aviv Tamar AVIVT@TX.TECHNION.AC.IL
Dotan Di Castro DOT@TX.TECHNION.AC.IL
Shie Mannor SHIE@EE.TECHNION.AC.IL
Department of Electrical Engineering, The Technion - Israel Institute of Technology, Haifa, Israel 32000

Abstract

In this paper we extend temporal difference policy evaluation algorithms to performance criteria that include the variance of the cumulative reward. Such criteria are useful for risk management, and are important in domains such as finance and process control. We propose variants of both TD(0) and LSTD(λ) with linear function approximation, prove their convergence, and demonstrate their utility in a 4-dimensional continuous state space problem.

1. Introduction

In sequential decision making within the Markov Decision Process (MDP) framework, policy evaluation refers to the process of mapping each state of the system to some statistical property of its long-term outcome, most commonly its *expected reward to go*. In the fields of Reinforcement Learning (RL; Bertsekas & Tsitsiklis, 1996; Sutton & Barto, 1998) and planning in MDPs (Puterman, 1994), policy evaluation is a fundamental step in many policy improvement algorithms. Yet in domains where policies are mostly hand designed, for example in clinical decision making, policy evaluation is also important, for a prudent choice of strategy must depend on it (Shortreed et al., 2011).

A principal challenge in policy evaluation arises when the state space is large, or continuous, necessitating some means of *approximation* for the process to be tractable. This difficulty is even more pronounced when a model of the process is not available, and the evaluation has to be *estimated* from a limited amount of samples. Fortunately, for the case of the expected reward to go, also known as the value function and denoted by J, the sequential nature of the problem may be exploited to overcome these difficulties. Temporal Difference methods (TD; Sutton, 1988) employ *function approximation* to represent J in a lower dimensional subspace, and tune the approximation parameters efficiently from data. Enjoying both theoretical guarantees (Bertsekas, 2012; Lazaric et al., 2010) and empirical success (Tesauro, 1995), these methods are considered the state of the art in policy evaluation.

However, when it comes to evaluating additional statistics of the reward to go, such as its variance, little is known. This is due to the fact that the expectation plays a key role in the Bellman equation, which drives TD algorithms.

Yet, the incentives to evaluate such statistics are extensive. In the context of RL and planning, incorporating such statistics into the performance evaluation criteria leads to *risk sensitive* optimization, a topic that has gained significant interest recently (Filar et al., 1995; Mihatsch & Neuneier, 2002; Geibel & Wysotzki, 2005; Mannor & Tsitsiklis, 2011). In a more general context, uncertainty in a policy's long-term outcome is critical for decision making in many areas, such as financial, process control, and clinical domains. In these domains, considering the variance of the total reward is particularly important, as it is both common-practice and intuitive to understand (Sharpe, 1966; Shortreed et al., 2011).

In this paper we present a TD framework for estimating the *variance of the reward to go*, denoted by V, using function approximation, in problems where a model is not available. To our knowledge, this is the first work that addresses the challenge of large state spaces, by considering an approximation scheme for V. Our approach is based on the following observation: the second moment of the reward to go, denoted by M, together with the value function J, obey a linear 'Bellman-like' equation. By extending TD methods to jointly estimate J and M with linear function approximation, we obtain a solution for estimating the

Proceedings of the 30^{th} International Conference on Machine Learning, Atlanta, Georgia, USA, 2013. JMLR: W&CP volume 28.

variance, using the relation $V = M - J^2$.

We propose both a variant of Least Squares Temporal Difference (LSTD) (Boyan, 2002) and of TD(0) (Sutton & Barto, 1998) for jointly estimating J and M with a linear function approximation. For these algorithms, we provide convergence guarantees and error bounds. In addition, we introduce a novel method for enforcing the approximate variance to be positive, through a constrained TD equation. An empirical evaluation on a challenging continuous maze problem demonstrates the applicability of our approach to large domains, and highlights the importance of the variance function in understanding the risk of a policy.

A previous study by Sato et al. (2001) suggested TD equations for J and V, without function approximation. Their approach relied on a non-linear equation for V, and it is not clear how it may be extended to handle large state spaces. More recently, Morimura et al. (2012) proposed TD learning rules for a parametric distribution of the return, albeit without function approximation nor formal guarantees. In the Bayesian GPTD framework of Engel et al. (2005), the reward-to-go is assumed to have a Gaussian posterior distribution, and its mean and variance are estimated. However, the resulting variance is a product of both stochastic transitions and model uncertainty, and is thus different than the variance considered here.

2. Framework and Background

We consider a Stochastic Shortest Path (SSP) problem[1,2] (Bertsekas, 2012), where the environment is modeled by an MDP in discrete time with a finite state space $X \triangleq \{1, \ldots, n\}$ and a terminal state x^*. A fixed policy π determines, for each $x \in X$, a stochastic transition to a subsequent state $x' \in \{X \cup x^*\}$ with probability $P(x'|x)$. We consider a deterministic and bounded reward function $r : X \to \mathbb{R}$, and assume zero reward at the terminal state. We denote by x_k the state at time k, where $k = 0, 1, 2, \ldots$.

A policy is said to be *proper* (Bertsekas, 2012) if there is a positive probability that the terminal state x^* will be reached after at most n transitions, from any initial state. In this paper we make the following assumption

Assumption 1. *The policy π is proper.*

[1]This is also known as an episodic setting.

[2]The popular infinite horizon discounted setting is actually simpler than the SSP considered here, as the discount factor simplifies the verification of the contraction properties presented in the sequel. Therefore, all of our results may easily be extended to that setting as well, with even simpler proofs.

Let $\tau \triangleq \min\{k > 0 | x_k = x^*\}$ denote the first visit time to the terminal state, and let the random variable B denote the accumulated reward along the trajectory until that time

$$B \triangleq \sum_{k=0}^{\tau-1} r(x_k).$$

In this work, we are interested in the mean-variance tradeoff in B, represented by the *value function*

$$J(x) \triangleq \mathbb{E}\left[B | x_0 = x\right], \quad x \in X,$$

and the *variance of the reward to go*

$$V(x) \triangleq \mathrm{Var}\left[B | x_0 = x\right], \quad x \in X.$$

We will find it convenient to define also the *second moment of the reward to go*

$$M(x) \triangleq \mathbb{E}\left[B^2 | x_0 = x\right], \quad x \in X.$$

Our goal is to estimate $J(x)$ and $V(x)$ from trajectories obtained by simulating the MDP with policy π.

3. Approximation of the Variance of the Reward To Go

In this section we derive a projected equation method for approximating $J(x)$ and $M(x)$ using linear function approximation. The estimation of $V(x)$ will then follow from the relation $V(x) = M(x) - J(x)^2$.

Our starting point is a system of equations for $J(x)$ and $M(x)$, first derived by Sobel (1982) for a discounted infinite horizon case, and extended here to the SSP case. The equation for J is the well known Bellman equation for a fixed policy, and independent of the equation for M.

Proposition 2. *The following equations hold for $x \in X$*

$$J(x) = r(x) + \sum_{x' \in X} P(x'|x) J(x'), \tag{1}$$

$$M(x) = r(x)^2 + 2r(x) \sum_{x' \in X} P(x'|x) J(x') + \sum_{x' \in X} P(x'|x) M(x').$$

Furthermore, under Assumption 1 a unique solution to (1) exists.

A straightforward proof is given in Appendix A.

At this point the reader may wonder why an equation for V is not presented. While such an equation may be derived, as was done by Tamar et al. (2012), it is not linear. The linearity of (1) in J and M is the key to our approach. As we show in the next subsection,

the solution to (1) may be expressed as the fixed point of a linear mapping in the joint space of J and M. We will then show that a projection of this mapping onto a linear feature space is contracting, thus allowing us to use existing TD theory to derive estimation algorithms for J and M.

3.1. A Projected Fixed Point Equation in the Joint Space of J and M

For the sequel, we introduce the following vector notations. We denote by $P \in \mathbb{R}^{n\times n}$ and $r \in \mathbb{R}^n$ the SSP transition matrix and reward vector, i.e., $P_{x,x'} = P(x'|x)$ and $r_x = r(x)$, where $x, x' \in X$. Also, we define the diagonal matrix $R \triangleq diag(r)$.

For a vector $z \in \mathbb{R}^{2n}$ we let $z_J \in \mathbb{R}^n$ and $z_M \in \mathbb{R}^n$ denote its leading and ending n components, respectively. Thus, such a vector belongs to the joint space of J and M.

We define the mapping $T : \mathbb{R}^{2n} \to \mathbb{R}^{2n}$ by

$$\begin{aligned}[Tz]_J &= r + Pz_J,\\ [Tz]_M &= Rr + 2RPz_J + Pz_M.\end{aligned} \tag{2}$$

It may easily be verified that a fixed point of T is a solution to (1), and by Proposition 2 such a fixed point exists and is unique.

When the state space X is large, a direct solution of (1) is not feasible, even if P may be accurately obtained. A popular approach in this case is to approximate $J(x)$ by restricting it to a lower dimensional subspace, and use simulation based TD algorithms to adjust the approximation parameters (Bertsekas, 2012). In this paper we extend this approach to the approximation of $M(x)$ as well.

We consider a linear approximation architecture of the form

$$\tilde{J}(x) = \phi_J(x)^\top w_J, \quad \tilde{M}(x) = \phi_M(x)^\top w_M,$$

where $w_J \in \mathbb{R}^l$ and $w_M \in \mathbb{R}^m$ are the approximation parameter vectors, $\phi_J(x) \in \mathbb{R}^l$ and $\phi_M(x) \in \mathbb{R}^m$ are state dependent features, and $(\cdot)^\top$ denotes the transpose of a vector. The low dimensional subspaces are therefore

$$S_J = \{\Phi_J w | w \in \mathbb{R}^l\}, \quad S_M = \{\Phi_M w | w \in \mathbb{R}^m\},$$

where Φ_J and Φ_M are matrices whose rows are $\phi_J(x)^\top$ and $\phi_M(x)^\top$, respectively. We make the following standard independence assumption on the features

Assumption 3. *The matrix Φ_J has rank l and the matrix Φ_M has rank m.*

As outlined earlier, our goal is to estimate w_J and w_M from simulated trajectories of the MDP. Thus, it is constructive to consider projections onto S_J and S_M with respect to a norm that is weighted according to the state occupancy in these trajectories.

For a trajectory $x_0, \ldots, x_{\tau-1}$, where x_0 is drawn from a fixed distribution $\zeta_0(x)$, and the states evolve according to the MDP with policy π, define the state occupancy probabilities

$$q_t(x) = P(x_t = x), \quad x \in X, \quad t = 0, 1, \ldots$$

and let

$$q(x) = \sum_{t=0}^{\infty} q_t(x), \quad x \in X$$
$$Q \triangleq diag(q).$$

We make the following assumption on the policy π and initial distribution ζ_0

Assumption 4. *Each state has a positive probability of being visited, namely, $q(x) > 0$ for all $x \in X$.*

For vectors in $\mathbb{R}^n$, we introduce the weighted Euclidean norm

$$\|y\|_q = \sqrt{\sum_{i=1}^{n} q(i)\,(y(i))^2}, \quad y \in \mathbb{R}^n,$$

and we denote by Π_J and Π_M the projections from $\mathbb{R}^n$ onto the subspaces S_J and S_M, respectively, with respect to this norm. For $z \in \mathbb{R}^{2n}$ we denote by Π the projection of z_J onto S_J and z_M onto S_M, namely [3]

$$\Pi = \begin{pmatrix} \Pi_J & 0 \\ 0 & \Pi_M \end{pmatrix}. \tag{3}$$

We are now ready to fully describe our approximation scheme. We consider the *projected* fixed point equation

$$z = \Pi T z, \tag{4}$$

and, letting z^* denote its solution, propose the approximate value function $\tilde{J} = z^*_J \in S_J$ and second moment function $\tilde{M} = z^*_M \in S_M$.

We proceed to derive some properties of the projected fixed point equation (4). We begin by stating a well known result regarding the contraction properties of the *projected Bellman operator* $\Pi_J T_J$, where $T_J y = r + Py$. A proof can be found at (Bertsekas, 2012), proposition 7.1.1.

[3] The projection operators Π_J and Π_M are linear, and may be written explicitly as $\Pi_J = \Phi_J(\Phi_J^\top Q \Phi_J)^{-1}\Phi_J^\top Q$, and similarly for Π_M.

Lemma 5. *Let Assumptions 1, 3, and 4 hold. Then, there exists some norm $\|\cdot\|_J$ and some $\beta_J < 1$ such that*

$$\|\Pi_J P y\|_J \leq \beta_J \|y\|_J, \quad \forall y \in \mathbb{R}^n.$$

Similarly, there exists some norm $\|\cdot\|_M$ and some $\beta_M < 1$ such that

$$\|\Pi_M P y\|_M \leq \beta_M \|y\|_M, \quad \forall y \in \mathbb{R}^n.$$

Next, we define a weighted norm on $\mathbb{R}^{2n}$

Definition 6. *For a vector $z \in \mathbb{R}^{2n}$ and a scalar $0 < \alpha < 1$, the α-weighted norm is*

$$\|z\|_\alpha = \alpha \|z_J\|_J + (1-\alpha)\|z_M\|_M, \tag{5}$$

where $\|\cdot\|_J$ and $\|\cdot\|_M$ are defined in Lemma 5.

Our main result of this section is given in the following proposition, where we show that the projected operator ΠT is a contraction with respect to the α-weighted norm.

Proposition 7. *Let Assumptions 1, 3, and 4 hold. Then, there exists some $0 < \alpha < 1$ and some $\beta < 1$ such that ΠT is a β-contraction with respect to the α-weighted norm, i.e.,*

$$\|\Pi T z_1 - \Pi T z_2\|_\alpha \leq \beta \|z_1 - z_2\|_\alpha, \quad \forall z_1, z_2 \in \mathbb{R}^{2n}.$$

Proof. First, using (2) and (3) we have that $\|\Pi T z_1 - \Pi T z_2\|_\alpha = \|\Pi \mathcal{P}(z_1 - z_2)\|_\alpha$, where

$$\Pi \mathcal{P} = \begin{pmatrix} \Pi_J P & 0 \\ 2\Pi_M R P & \Pi_M P \end{pmatrix}.$$

Thus, it suffices to show that for all $z \in \mathbb{R}^{2n}$

$$\|\Pi \mathcal{P} z\|_\alpha \leq \beta \|z\|_\alpha.$$

We will now show that $\|\Pi \mathcal{P} z\|_\alpha$ may be separated into two terms which may be bounded by Lemma 5, and an additional cross term. By balancing α and β, this term may be contained to yield the required contraction.

We have

$$\begin{aligned}
\|\Pi \mathcal{P} z\|_\alpha =& \alpha \|\Pi_J P z_J\|_J \\
&+ (1-\alpha)\|2\Pi_M R P z_J + \Pi_M P z_M\|_M \\
\leq& \alpha \|\Pi_J P z_J\|_J + (1-\alpha)\|\Pi_M P z_M\|_M \\
&+ (1-\alpha)\|2\Pi_M R P z_J\|_M \\
\leq& \alpha\beta_J \|z_J\|_J + (1-\alpha)\beta_M \|z_M\|_M \\
&+ (1-\alpha)\|2\Pi_M R P z_J\|_M,
\end{aligned} \tag{6}$$

where the equality is by definition of the α weighted norm (5), the first inequality is from the triangle inequality, and the second inequality is by Lemma 5.

Now, we claim that there exists some finite C such that

$$\|2\Pi_M R P y\|_M \leq C \|y\|_J, \quad \forall y \in \mathbb{R}^n. \tag{7}$$

To see this, note that since $\mathbb{R}^n$ is a finite dimensional real vector space, all vector norms are equivalent (Horn & Johnson, 1985) therefore there exist finite C_1 and C_2 such that for all $y \in \mathbb{R}^n$

$$C_1 \|2\Pi_M R P y\|_2 \leq \|2\Pi_M R P y\|_M \leq C_2 \|2\Pi_M R P y\|_2,$$

where $\|\cdot\|_2$ denotes the Euclidean norm. Let λ denote the spectral norm of the matrix $2\Pi_M R P$, which is finite since all the matrix elements are finite. We have that

$$\|2\Pi_M R P y\|_2 \leq \lambda \|y\|_2, \quad \forall y \in \mathbb{R}^n.$$

Using again the fact that all vector norms are equivalent, there exists a finite C_3 such that

$$\|y\|_2 \leq C_3 \|y\|_J, \quad \forall y \in \mathbb{R}^n.$$

Setting $C = C_2 \lambda C_3$ we get the desired bound. Let $\tilde{\beta} = \max\{\beta_J, \beta_M\} < 1$, and choose $\epsilon > 0$ such that

$$\tilde{\beta} + \epsilon < 1.$$

Now, choose α such that $\alpha = \frac{C}{\epsilon + C}$. We have that

$$(1-\alpha)C = \alpha\epsilon,$$

and plugging this into (7) yields

$$(1-\alpha)\|2\Pi_M R P y\|_M \leq \alpha\epsilon \|y\|_J. \tag{8}$$

We now return to (6), where we have

$$\begin{aligned}
&\alpha\beta_J \|z_J\|_J + (1-\alpha)\beta_M \|z_M\|_M + (1-\alpha)\|2\Pi_M R P z_J\|_M \\
\leq& \alpha\beta_J \|z_J\|_J + (1-\alpha)\beta_M \|z_M\|_M + \alpha\epsilon \|z_J\|_J \\
\leq& (\tilde{\beta} + \epsilon)\left(\alpha \|z_J\|_J + (1-\alpha)\|z_M\|_M\right),
\end{aligned}$$

where the first inequality is by (8), and the second is by the definition of $\tilde{\beta}$. We have thus shown that

$$\|\Pi \mathcal{P} z\|_\alpha \leq (\tilde{\beta} + \epsilon)\|z\|_\alpha.$$

Finally, choose $\beta = \tilde{\beta} + \epsilon$. □

Proposition 7 guarantees that the projected operator ΠT has a unique fixed point. Let us denote this fixed point by z^*, and let w_J^*, w_M^* denote the corresponding weights, which are unique due to Assumption 3

$$\begin{aligned}
\Pi T z^* &= z^*, \\
z_J^* &= \Phi_J w_J^*, \\
z_M^* &= \Phi_M w_M^*.
\end{aligned} \tag{9}$$

In the next proposition we provide a bound on the approximation error. The proof is in Appendix B.

Proposition 8. *Let Assumptions 1, 3, and 4 hold. Denote by $z_{true} \in \mathbb{R}^{2n}$ the true value and second moment functions, i.e., $[z_{true}]_J = J$, and $[z_{true}]_M = M$. Then,*

$$\|z_{true} - z^*\|_\alpha \leq \frac{1}{1-\beta}\|z_{true} - \Pi z_{true}\|_\alpha,$$

with α and β defined in Proposition 7.

4. Simulation Based Estimation Algorithms

In this section we propose algorithms that estimate $\tilde{J}$ and $\tilde{M}$ from sampled trajectories of the MDP, based on the approximation architecture of the previous section.

We begin by writing the projected equation (9) in matrix form. First, let us write the equation explicitly as

$$\begin{aligned} \Pi_J\left(r + P\Phi_J w_J^*\right) &= \Phi_J w_J^*, \\ \Pi_M\left(Rr + 2RP\Phi_J w_J^* + P\Phi_M w_M^*\right) &= \Phi_M w_M^*. \end{aligned} \tag{10}$$

Projecting a vector y onto Φw satisfies the following orthogonality condition

$$\Phi^\top Q(y - \Phi w) = 0,$$

we therefore have

$$\begin{aligned} \Phi_J^\top Q\left(\Phi_J w_J^* - (r + P\Phi_J w_J^*)\right) &= 0, \\ \Phi_M^\top Q\left(\Phi_M w_M^* - (Rr + 2RP\Phi_J w_J^* + P\Phi_M w_M^*)\right) &= 0, \end{aligned}$$

which can be written as

$$Aw_J^* = b, \quad Cw_M^* = d, \tag{11}$$

with

$$\begin{aligned} A &= \Phi_J^\top Q\left(I - P\right)\Phi_J, \quad b = \Phi_J^\top Qr, \\ C &= \Phi_M^\top Q\left(I - P\right)\Phi_M, \quad d = \Phi_M^\top QR\left(r + 2P\Phi_J A^{-1}b\right), \end{aligned} \tag{12}$$

and the matrices A and C are invertible since Proposition 7 guarantees a unique solution to (9) and Assumption 3 guarantees the unique weights of its projection.

4.1. A Least Squares TD Algorithm

Our first simulation-based algorithm is an extension of the Least Squares Temporal Difference (LSTD) algorithm (Boyan, 2002). We simulate N trajectories of the MDP with the policy π and initial state distribution ζ_0. Let $x_0^k, x_1^k, \ldots, x_{\tau^k-1}^k$ and τ^k, where $k = 0, 1, \ldots, N$, denote the state sequence and visit times to the terminal state within these trajectories, respectively. We now use these trajectories to form the following estimates of the terms in (12)

$$\begin{aligned} A_N &= \mathbb{E}_N\left[\sum_{t=0}^{\tau-1}\phi_J(x_t)\left(\phi_J(x_t) - \phi_J(x_{t+1})\right)^\top\right], \\ b_N &= \mathbb{E}_N\left[\sum_{t=0}^{\tau-1}\phi_J(x_t)r(x_t)\right], \\ C_N &= \mathbb{E}_N\left[\sum_{t=0}^{\tau-1}\phi_M(x_t)\left(\phi_M(x_t) - \phi_M(x_{t+1})\right)^\top\right], \\ d_N &= \mathbb{E}_N\left[\sum_{t=0}^{\tau-1}\phi_M(x_t)r(x_t)\left(r(x_t) + 2\phi_J(x_{t+1})^\top A_N^{-1}b_N\right)\right], \end{aligned} \tag{13}$$

where $\mathbb{E}_N$ denotes an empirical average over trajectories, i.e., $\mathbb{E}_N\left[f(x,\tau)\right] = \frac{1}{N}\sum_{k=1}^N f(x^k, \tau^k)$. The LSTD approximation is given by

$$\hat{w}_J^* = A_N^{-1}b_N, \quad \hat{w}_M^* = C_N^{-1}d_N.$$

The next theorem shows that LSTD converges.

Theorem 9. *Let Assumptions 1, 3, and 4 hold. Then $\hat{w}_J^* \to w_J^*$ and $\hat{w}_M^* \to w_M^*$ as $N \to \infty$ with probability 1.*

The proof involves a straightforward application of the law of large numbers and is described in Appendix C. Convergence rates for regular LSTD were derived by Konda (2002) and Lazaric et al. (2010), and may be extended to the algorithm presented here. This issue is deferred to the full version of this paper.

4.2. An Online TD(0) Algorithm

Our second estimation algorithm is an extension of the well known TD(0) algorithm (Sutton & Barto, 1998). Again, we simulate trajectories of the MDP corresponding to the policy π and initial state distribution ζ_0, and we iteratively update our estimates at every visit to the terminal state[4]. For some $0 \leq t < \tau^k$ and weights w_J, w_M, we introduce the TD terms

$$\begin{aligned} \delta_J^k(t, w_J, w_M) =& r(x_t^k) + \left(\phi_J(x_{t+1}^k)^\top - \phi_J(x_t^k)^\top\right)w_J, \\ \delta_M^k(t, w_J, w_M) =& r^2(x_t^k) + 2r(x_t^k)\phi_J(x_{t+1}^k)^\top w_J \\ &+ \left(\phi_M(x_{t+1}^k)^\top - \phi_M(x_t^k)^\top\right)w_M. \end{aligned}$$

Note that δ_J^k is the standard TD error (Sutton & Barto, 1998). For the intuition behind δ_M^k, observe that M in (1) is equivalent to the value function of an MDP with stochastic reward $r(x)^2 + 2r(x)J(x')$, where $x' \sim P(x'|x)$. δ_M^k is then the equivalent TD error, with $\phi_J(x')^\top w_J$ substituting $J(x')$. The TD(0) algorithm

[4] An extension to an algorithm that updates at every state transition is possible, but we do not pursue such here.

is given by

$$\hat{w}_{J;k+1} = \hat{w}_{J;k} + \xi_k \sum_{t=0}^{\tau^k-1} \phi_J(x_t)\delta_J^k(t, \hat{w}_{J;k}, \hat{w}_{M;k}),$$

$$\hat{w}_{M;k+1} = \hat{w}_{M;k} + \xi_k \sum_{t=0}^{\tau^k-1} \phi_M(x_t)\delta_M^k(t, \hat{w}_{J;k}, \hat{w}_{M;k}),$$

where $\{\xi_k\}$ are positive step sizes.

The next theorem shows that TD(0) converges.

Theorem 10. *Let Assumptions 1, 3, and 4 hold, and let the step sizes satisfy*

$$\sum_{k=0}^{\infty} \xi_k = \infty, \quad \sum_{k=0}^{\infty} \xi_k^2 < \infty.$$

Then $\hat{w}_{J;k} \to w_J^*$ and $\hat{w}_{M;k} \to w_M^*$ as $k \to \infty$ with probability 1.

The proof, provided in Appendix D, is based on representing the algorithm as a stochastic approximation, and using a result of Borkar (2008) to show that the iterates asymptotically track a certain ordinary differential equation (ODE). This ODE is then shown to have a unique asymptotically stable equilibrium exactly at w_J^*, w_M^*. Convergence rates for TD(0) may be derived along the lines of Konda (2002), with the details deferred to the full version of this paper.

4.3. Multistep LSTD(λ) Algorithms

A common method in value function approximation is to replace the single step mapping T_J with a multistep version of the form

$$T_J^{(\lambda)} = (1-\lambda)\sum_{l=0}^{\infty} \lambda^l T_J^{l+1}$$

with $0 < \lambda < 1$. The projected equation (10) then becomes $\Pi_J T_J^{(\lambda)}\left(\Phi_J w_J^{*(\lambda)}\right) = \Phi_J w_J^{*(\lambda)}$. Similarly, we may write a multistep equation for M

$$\Pi_M T_M^{(\lambda)}\left(\Phi_M w_M^{*(\lambda)}\right) = \Phi_M w_M^{*(\lambda)}, \tag{14}$$

where

$$T_M^{(\lambda)} = (1-\lambda)\sum_{l=0}^{\infty} \lambda^l T_{M^*}^{l+1},$$

and

$$T_{M^*}(y) = Rr + 2RP\Phi_J w_J^{*(\lambda)} + Py.$$

Note the difference between T_{M^*} and $[T]_M$ defined earlier; We are no longer working on the joint space of J and M but instead we have an independent equation for approximating J, and its solution $w_J^{*(\lambda)}$ is part of Equation (14) for approximating M. By Proposition 7.1.1 of Bertsekas (2012) both $\Pi_J T_J^{(\lambda)}$ and $\Pi_M T_M^{(\lambda)}$ are contractions with respect to the weighted norm $\|\cdot\|_q$, therefore both multistep projected equations admit a unique solution. In a similar manner to the single step version, the projected equations may be written in matrix form

$$A^{(\lambda)} w_J^{*(\lambda)} = b^{(\lambda)}, \quad C^{(\lambda)} w_M^{*(\lambda)} = d^{(\lambda)}, \tag{15}$$

where

$$A^{(\lambda)} = \Phi_J^\top Q\left(I - P^{(\lambda)}\right)\Phi_J, \quad b^{(\lambda)} = \Phi_J^\top Q(I-\lambda P)^{-1} r,$$

$$C^{(\lambda)} = \Phi_M^\top Q\left(I - P^{(\lambda)}\right)\Phi_M,$$

$$d^{(\lambda)} = \Phi_M^\top Q(I-\lambda P)^{-1} R\left(r + 2P\Phi_J w_J^{*(\lambda)}\right),$$

and $P^{(\lambda)} = (1-\lambda)\sum_{l=0}^{\infty}\lambda^l P^{l+1}$.

Simulation based estimates $A_N^{(\lambda)}$ and $b_N^{(\lambda)}$ of the expressions above may be obtained by using eligibility traces, as described by Bertsekas (2012), and the LSTD(λ) approximation is then given by $\hat{w}_J^{*(\lambda)} = (A_N^{(\lambda)})^{-1} b_N^{(\lambda)}$. By substituting $w_J^{*(\lambda)}$ with $\hat{w}_J^{*(\lambda)}$ in the expression for $d^{(\lambda)}$, a similar procedure may be used to derive estimates $C_N^{(\lambda)}$ and $d_N^{(\lambda)}$, and to obtain the LSTD(λ) approximation $\hat{w}_M^{*(\lambda)} = (C_N^{(\lambda)})^{-1} d_N^{(\lambda)}$. A convergence result similar to Theorem 9 may also be obtained. Due to the similarity to the LSTD procedure in (13), the exact details are omitted.

5. Non Negative Approximate Variance by Constrained Projection

The TD algorithms of the preceding section approximated J and M by the solution to the fixed point equation (9). While Proposition 8 shows that the approximation errors of $\tilde{J}$ and $\tilde{M}$ are bounded, it does not guarantee that the approximated variance $\tilde{V}$, given by $\tilde{M} - \tilde{J}^2$, is non-negative for all states. A trivial remedy is to set all negative values of $\tilde{V}$ to zero; however, by such we lose all information in these states. In this section we propose an alternative method, based on modifying the fixed point equation (9) to include constraints for variance non-negativeness. We thus obtain a different approximation architecture, in which a non-negative variance is inherent. We now present the constrained equation and discuss how its solution may be computed.

First, let us write the multistep equation for the second moment weights (14) with the projection operator as an explicit minimization

$$w_M^{*(\lambda)} = \arg\min_w \|\Phi_M w - \left(\tilde{r} + \tilde{\Phi} w_M^{*(\lambda)}\right)\|_q,$$

with

$$\tilde{\Phi} = P^{(\lambda)}\Phi_M, \quad \tilde{r} = (I - \lambda P)^{-1}\left(Rr + 2RP\Phi_J w_J^{*(\lambda)}\right).$$

Observe that a non-negative variance in some state x may be written as a *linear* inequality in $w_M^{*(\lambda)}$

$$\phi_M(x)^\top w_M^{*(\lambda)} - (\phi_J(x)^\top w_J^{*(\lambda)})^2 \geq 0.$$

We now propose to add such inequality constraints to the projection operator. Let $\{x_1, \ldots, x_s\}$ denote a set of states in which we demand that the variance be non-negative. Let $H \in \mathbb{R}^{s\times m}$ denote a matrix with the features $-\phi_M^\top(x_i)$ as its rows, and let $g \in \mathbb{R}^s$ denote a vector with elements $-(\phi_J(x_i)^\top w_J^{*(\lambda)})^2$. We write the non-negative-variance projected equation for the second moment as

$$w_M^+ = \begin{cases} \arg\min_w & \|\Phi_M w - \left(\tilde{r} + \tilde{\Phi} w_M^+\right)\|_q \\ \text{s.t.} & Hw \leq g \end{cases} \quad (16)$$

Here, w_M^+ denotes the weights of $\tilde{M}$ in the modified approximation architecture. We now discuss whether a solution to (16) exists, and how it may be obtained.

Let us assume that the constraints in (16) admit a feasible solution:

Assumption 11. *There exists w such that $Hw < g$.*

Note that a trivial way to satisfy Assumption 11 is to have some feature vector that is positive for all states.

Equation (16) is a form of projected equation studied by Bertsekas (2011), the solution of which exists, and may be obtained by the following iterative procedure

$$w_{k+1} = \Pi_{\Xi,\hat{W}_M}[w_k - \gamma\Xi^{-1}(C^{(\lambda)}w_k - d^{(\lambda)})], \quad (17)$$

where Ξ is an arbitrary positive definite matrix, and $\Pi_{\Xi,\hat{W}_M}$ denotes a projection onto the convex set $\hat{W}_M = \{w|Hw \leq g\}$ with respect to the Ξ weighted Euclidean norm. The following lemma, which is based on a convergence result of Bertsekas (2011), guarantees that algorithm (17) converges.

Lemma 12. *Assume $\lambda > 0$, and let Assumption 11 hold. Then (16) admits a unique solution w_M^+, and there exists $\bar{\gamma} > 0$ such that $\forall\gamma \in (0, \bar{\gamma})$ and $\forall w_0 \in \mathbb{R}^m$ the algorithm (17) converges at a linear rate to w_M^+.*

Proof. This is a direct application of the convergence result of Bertsekas (2011). The only nontrivial assumption that needs to be verified is that $T_M^{(\lambda)}$ is a contraction in the $\|\cdot\|_q$ norm (Proposition 1 in Bertsekas, 2011). For $\lambda > 0$ Proposition 7.1.1. of Bertsekas (2012) guarantees that $T_M^{(\lambda)}$ is indeed contracting in the $\|\cdot\|_q$ norm. □

Generally, $C^{(\lambda)}$, $d^{(\lambda)}$, and $w_J^{*(\lambda)}$ are not known in advance, and should be replaced in (17) with their simulation based estimates, $C_N^{(\lambda)}$, $d_N^{(\lambda)}$, and $\hat{w}_J^{*(\lambda)}$, proposed in the previous section. The convergence of these estimates, together with the result of Lemma 12, lead to the following convergence result, which is given without proof.

Theorem 13. *Consider the algorithm in (17) with $C^{(\lambda)}$, $d^{(\lambda)}$, and $w_J^{*(\lambda)}$ replaced by $C_N^{(\lambda)}$, $d_N^{(\lambda)}$, and $\hat{w}_J^{*(\lambda)}$, respectively, and with $k(N)$ replacing k for a specific N. Also, let the assumptions in Lemma 12 hold, and let $\gamma \in (0, \bar{\gamma})$, with $\bar{\gamma}$ defined in Lemma 12. Then $w_{k(N)} \to w_M^+$ as $N \to \infty$ and $k \to \infty$ almost surely.*

An in-depth study of the approximation architecture (16) is deferred to the full version of this paper. However, an illustration on a toy problem is provided in Appendix E.

6. Experiments

In this section we present numerical simulations of policy evaluation on a challenging continuous maze domain. The goal of this presentation is threefold; first, we show that the variance of the reward-to-go may be estimated successfully on a large state space. Second, the intuitive maze domain highlights the insight that may be gleaned from this variance, and third, we show that in terms of sample efficiency, our LSTD(λ) algorithm significantly outperforms the current state-of-the-art. We begin by describing the domain and then present our policy evaluation results.

The Pinball Domain (Konidaris & Barto, 2009) is a continuous 2-dimensional maze where a small ball needs to be maneuvered between obstacles to reach some target area, as depicted in Figure 1A. The ball is controlled by applying a constant force in one of the 4 directions at each time step, which causes acceleration in the respective direction. In addition, the ball's velocity is susceptible to additive Gaussian noise (zero mean, standard deviation 0.03) and friction (drag coefficient 0.995). The obstacles are sharply shaped, and collisions are fully elastic. The state of the ball is thus 4-dimensional $(x, y, \dot{x}, \dot{y})$, and the action set is discrete, with 4 available controls. The reward is -1 for all states until reaching the target. A Java implementation of the pinball domain used by Konidaris & Barto (2009) is available on-line[5] and was used for our simulations as well, with the addition of noise to the velocity.

A near-optimal policy π was obtained using SARSA (Sutton & Barto, 1998) with radial basis function fea-

[5] http://people.csail.mit.edu/gdk/software.html

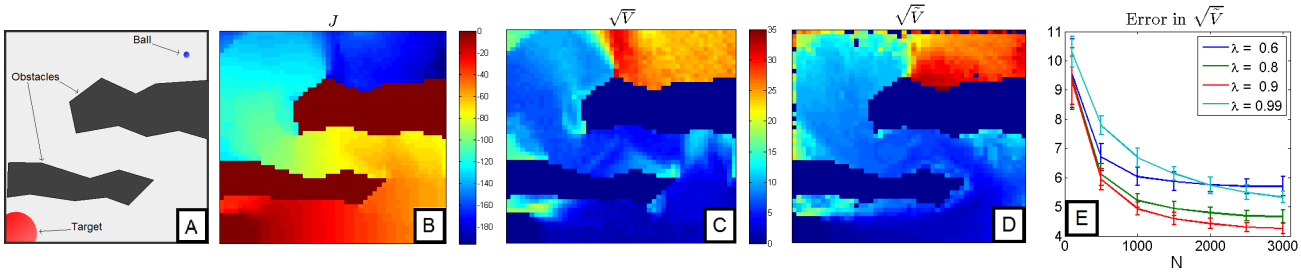

Figure 1. Experimental evaluation. A: The pinball domain. B,C: The 'true' value function J (left color bar) and standard deviation of the reward to go $\sqrt{V}$ (right color bar), estimated by monte carlo. D: Approximate standard deviation $\sqrt{\tilde{V}}$, using LSTD(0.9); same color bar as in (C). E: RMS error of $\sqrt{\tilde{V}}$ vs. number of trajectories N. Standard deviation error-bars from 10 runs are shown.

tures. The value J and standard deviation of the reward-to-go $\sqrt{V}$ for this policy are plotted in Figure 1(B;C), for 1816 equally spaced states between the obstacles with zero velocity. These plots were obtained by Monte Carlo (MC) estimation of the mean and variance, using over 2 million trajectories starting from these states. To our knowledge, MC is the current state-of-the-art technique for obtaining such variance estimates. As should be expected, the value is approximately a linear function of the distance to the target. In contrast, the standard deviation is clearly not linear in the distance, and in some places not even monotone. Furthermore, we see that an area in the top part of the maze before the first turn is very risky, even more than the farthest point from the target. We stress that this information cannot be gleaned from inspecting the value function alone.

Figure 1D shows the approximate standard deviation $\sqrt{\tilde{V}}$ obtained by the LSTD(λ) algorithm of Section 4.3. We used uniform tile features for $\tilde{J}$ and $\tilde{M}$ (50×50 non-overlapping tiles in x and y without dependence on velocity, for the same resolution as the MC estimate), and set $\lambda = 0.9$. To emphasize the efficiency of our method, we used only one sample trajectory per each state in the MC evaluation – a total of $N = 1816$ trajectories, with uniformly distributed initial states. Clearly, a single sample for each evaluation point is insufficient for a meaningful MC variance estimate. However, by exploiting relations *between* states (1), LSTD provides a reasonable approximation.

We further explore LSTD(λ) in Figure 1E, where we show the RMS error of $\sqrt{\tilde{V}}$ (compared to the MC estimate) for different values of λ and N. As in regular LSTD, λ trades off estimation bias and variance.

7. Conclusion

This work presented a novel framework for policy evaluation in RL with respect to the variance of the reward to go. We presented both formal guarantees and empirical evidence that this approach is useful in problems with a large state space. To the best of our knowledge, such problems are beyond the capabilities of previous approaches.

A requirement of variance evaluation is that it be non-negative. We approached this issue by adding constraints to the second moment approximation. An alternative is through the choice of features. Interestingly, in our experiments we found that using non-overlapping tile features produces a non-negative approximate variance. For this choice of features (identical for J and M), we can show that the *direct* approximation is always non-negative, i.e., $\Pi M - (\Pi J)^2 \geq 0$, where the square is element-wise. Whether this holds also for the fixed-point approximation, and if there are other features with this property, is an open question.

We conclude with a discussion on policy optimization with respect to a mean-variance tradeoff. While a naive variance-penalized policy iteration algorithm may be easily conceived, its usefulness should be questioned, as it was shown to be problematic for the standard deviation adjusted reward (Sobel, 1982) and the variance constrained reward (Mannor & Tsitsiklis, 2011). Perhaps a wiser approach would be to consider gradient based updates. Tamar et al. (2012) proposed policy gradient algorithms for a class of variance related criteria, and showed their convergence to local optima. These algorithms may be extended to use the variance function in an actor-critic type scheme. Such a study is left for future research.

Acknowledgments

The research leading to these results has received funding from the European Research Counsel under the European Union's Seventh Framework Program (FP7/2007-2013) / ERC Grant Agreement No 306638.

References

Bertsekas, D. P. Temporal difference methods for general projected equations. *IEEE Trans. Auto. Control*, 56(9):2128–2139, 2011.

Bertsekas, D. P. *Dynamic Programming and Optimal Control, Vol II*. Athena Scientific, fourth edition, 2012.

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

Borkar, V. S. *Stochastic Approximation: A Dynamical Systems Viewpoint*. Cambridge Univ Press, 2008.

Boyan, J. A. Technical update: Least-squares temporal difference learning. *Machine Learning*, 49(2): 233–246, 2002.

Engel, Y., Mannor, S., and Meir, R. Reinforcement learning with Gaussian processes. In *ICML*, 2005.

Filar, J. A., Krass, D., and Ross, K. W. Percentile performance criteria for limiting average Markov decision processes. *IEEE Trans. Auto. Control*, 40(1): 2–10, 1995.

Geibel, P. and Wysotzki, F. Risk-sensitive reinforcement learning applied to control under constraints. *JAIR*, 24(1):81–108, 2005.

Horn, R. A. and Johnson, C. R. *Matrix Analysis*. Cambridge University Press, 1985.

Konda, V. *Actor-Critic Algorithms*. PhD thesis, Dept. Comput. Sci. Elect. Eng., MIT, Cambridge, MA, 2002.

Konidaris, G. D. and Barto, A. G. Skill discovery in continuous reinforcement learning domains using skill chaining. In *NIPS*, 2009.

Lazaric, A., Ghavamzadeh, M., and Munos, R. Finite-sample analysis of LSTD. In *ICML*, 2010.

Mannor, S. and Tsitsiklis, J. N. Mean-variance optimization in Markov decision processes. In *ICML*, 2011.

Mihatsch, O. and Neuneier, R. Risk-sensitive reinforcement learning. *Machine Learning*, 49(2):267–290, 2002.

Morimura, T., Sugiyama, M., Kashima, H., Hachiya, H., and Tanaka, T. Parametric return density estimation for reinforcement learning. *arXiv preprint arXiv:1203.3497*, 2012.

Puterman, M. L. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, Inc., 1994.

Sato, M., Kimura, H., and Kobayashi, S. TD algorithm for the variance of return and mean-variance reinforcement learning. *Transactions of the Japanese Society for Artificial Intelligence*, 16:353–362, 2001.

Sharpe, W. F. Mutual fund performance. *The Journal of Business*, 39(1):119–138, 1966.

Shortreed, S. M., Laber, E., Lizotte, D. J., Stroup, T. S., Pineau, J., and Murphy, S. A. Informing sequential clinical decision-making through reinforcement learning: an empirical study. *Machine learning*, 84(1):109–136, 2011.

Sobel, M. J. The variance of discounted Markov decision processes. *J. Applied Probability*, pp. 794–802, 1982.

Sutton, R. S. Learning to predict by the methods of temporal differences. *Machine Learning*, 3(1):9–44, 1988.

Sutton, R. S. and Barto, A. G. *Reinforcement Learning*. MIT Press, 1998.

Tamar, A., Di Castro, D., and Mannor, S. Policy gradients with variance related risk criteria. In *ICML*, 2012.

Tesauro, G. Temporal difference learning and TD-gammon. *Communications of the ACM*, 38(3):58–68, 1995.