December 2003 **Report LIDS-P-2573**

IMPROVED TEMPORAL DIFFERENCE METHODS WITH LINEAR FUNCTION APPROXIMATION[1]

by

D. P. Bertsekas,[2] V. S. Borkar,[3] and A. Nedić[4]

Abstract

We consider temporal difference algorithms within the context of infinite-horizon finite-state dynamic programming problems with discounted cost, and linear cost function approximation. We show, under standard assumptions, that a least squares-based temporal difference method, proposed by Nedić and Bertsekas [NeB03], converges with a stepsize equal to 1. To our knowledge, this is the first iterative temporal difference method that converges without requiring a diminishing stepsize. We discuss the connections of the method with Sutton's TD(λ) and with various versions of least squares-based value iteration, and we show via analysis and experiment that the method is substantially and often dramatically faster than TD(λ), as well as simpler and more reliable. We also discuss the relation of our method with the LSTD method of Boyan [Boy02], and Bradtke and Barto [BrB96].

[1] Research supported by NSF Grant ECS-0218328 and Grant III.5(157)/99-ET from the Dept. of Science and Technology, Government of India. Thanks are due to Janey Yu for her assistance with the computational experimentation.

[2] Lab. for Information and Decision Systems, M.I.T., Cambridge, MA., 02139

[3] School of Technology and Computer Science, Tata Institute of Fundamental Research, Homi Bhabha Road, Mumbai 400005, India.

[4] Alphatech, Inc., Burlington, MA.

1. INTRODUCTION

In this paper, we analyze methods for approximate evaluation of the cost-to-go function of a stationary Markov chain within the framework of infinite-horizon discounted dynamic programming. We denote the states by $1, \ldots, n$, the transition probabilities by p_{ij}, $i, j = 1, \ldots, n$, and the corresponding costs by $\alpha^t g(i, j)$, where α is a discount factor with $0 < \alpha < 1$. We want to evaluate the long-term expected cost corresponding to each initial state i, given by

$$J(i) = E\left[\sum_{t=0}^{\infty} \alpha^t g(i_t, i_{t+1}) \;\Big|\; i_0 = i\right], \qquad \forall\, i = 1, \ldots, n,$$

where i_t denotes the state at time t. This problem arises as a subproblem in the policy iteration method of dynamic programming, and its variations, such as modified policy iteration, optimistic policy iteration, and λ-policy iteration (see Bertsekas and Tsitsiklis [BeT96], Bertsekas [Ber01], and Puterman [Put94] for extensive discussions of these methods).

The cost function $J(i)$ is approximated by a linear function of the form

$$\tilde{J}(i, r) = \phi(i)'r, \qquad \forall\, i = 1, \ldots, n,$$

where $\phi(i)$ is an s-dimensional feature vector, associated with the state i, with components $\phi_1(i), \ldots, \phi_s(i)$, while r is a weight vector with components $r(1), \ldots, r(s)$. (Throughout the paper, vectors are viewed as column vectors, and a prime denotes transposition.)

Our standing assumptions are:

(a) The Markov chain has steady-state probabilities $\pi(1), \ldots, \pi(n)$ which are positive, i.e.,

$$\lim_{t \to \infty} P[i_t = j \mid i_0 = i] = \pi(j) > 0, \qquad \forall\, i, j.$$

(b) The matrix Φ given by

$$\Phi = \begin{bmatrix} -\ \phi(1)'\ - \\ \vdots \\ -\ \phi(n)'\ - \end{bmatrix}$$

has rank s.

The TD(λ) method with function approximation was originally proposed by Sutton [Sut88], and its convergence has been analyzed by several authors, including Dayan [Day92], Gurvits, Lin, and Hanson [GLH94], Pineda [Pin97], Tsitsiklis and Van Roy [TsV97], and Van Roy [Van98]. We follow the line of analysis and Tsitsiklis and Van Roy, who have also considered a discounted problem under the preceding assumptions on the existence of steady-state probabilities and rank of Φ.

The algorithm, described in several references, including the books by Bertsekas and Tsitsiklis [BeT96], and Sutton and Barto [SuB98], generates an infinitely long trajectory of the Markov chain $(i_0, i_1, \ldots)$ using a simulator, and at time t iteratively updates the current estimate r_t using an iteration that depends on a fixed scalar $\lambda \in [0, 1]$, and on the temporal differences

$$d_t(i_k, i_{k+1}) = g(i_k, i_{k+1}) + \alpha\phi(i_{k+1})'r_t - \phi(i_k)'r_t, \qquad \forall\ t = 0, 1, \ldots,\ \forall\ k \leq t.$$

Tsitsiklis and Van Roy [TsV97] have introduced the linear system of equations

$$Ar + b = 0,$$

where A and b are given by

$$A = \Phi' D(\alpha P - I) \sum_{m=0}^{\infty} (\alpha\lambda P)^m \Phi, \qquad b = \Phi' D \sum_{m=0}^{\infty} (\alpha\lambda P)^m \bar{g}, \tag{1.1}$$

P is the transition probability matrix of the Markov chain, D is the diagonal matrix with diagonal entries $\pi(i)$, $i = 1, \ldots, n$,

$$D = \begin{pmatrix} \pi(1) & 0 & \cdots & 0 \\ 0 & \pi(2) & \cdots & 0 \\ & & \cdots & \\ 0 & 0 & \cdots & \pi(n) \end{pmatrix}, \tag{1.2}$$

and $\overline{g}$ is the vector with components $\overline{g}(i) = \sum_{j=1}^{n} p_{ij} g(i, j)$. They have shown that TD(λ) converges to the unique solution $r^* = -A^{-1}b$ of the system $Ar + b = 0$, and that the error between the corresponding approximation Φr^* and the true cost-to-go vector J satisfies

$$\|\Phi r^* - J\|_D \leq \frac{1 - \alpha\lambda}{1 - \alpha} \|\Pi J - J\|_D,$$

where $\|\cdot\|_D$ is the weighted norm corresponding to the matrix D (i.e., $\|x\|_D = \sqrt{x'Dx}$), and Π is the matrix given by $\Pi = \Phi(\Phi' D \Phi)^{-1} \Phi' D$. (Note that $\Pi J - J$ is the difference between J and its projection, with respect to the weighted norm, on the range of the feature matrix Φ.)

The essence of the Tsitsiklis and Van Roy analysis is to write the TD(λ) algorithm as

$$r_{t+1} = r_t + \gamma_t(Ar_t + b) + \gamma_t(\Xi_t r_t + \xi_t), \qquad t = 0, 1, \ldots, \tag{1.3}$$

where γ_t is a positive stepsize, and Ξ_t and ξ_t are some sequences of random matrices and vectors, respectively, that depend only on the simulated trajectory (so they are independent of r_t), and asymptotically have zero mean. A key to the convergence proof is that the matrix A is negative definite, so it has eigenvalues with negative real parts, which implies in turn that the matrix $I + \gamma_t A$ has eigenvalues within the unit circle for sufficiently small γ_t. However, in TD(λ) it is

essential that the stepsize γ_t be diminishing to 0, both because a small γ_t is needed to keep the eigenvalues of $I + \gamma_t A$ within the unit circle, and also because Ξ_t and ξ_t do not converge to 0.

In this paper, we focus on the λ-least squares policy evaluation method (λ-LSPE for short), proposed and analyzed by Nedić and Bertsekas [NeB03]. This algorithm was motivated as a simulation-based implementation of the λ-policy iteration method, proposed by Bertsekas and Ioffe [BeI96] (also described in Bertsekas and Tsitsiklis [BeT96], Section 2.3.1). In fact the method of this paper was also stated (without convergence analysis), and was used with considerable success by Bertsekas and Ioffe [BeI96] [see also Bertsekas and Tsitsiklis [BeT96], Eq. (8.6)] to train a tetris playing program – a challenging large-scale problem that TD(λ) failed to solve. In this paper, rather than focusing on the connection with λ-policy iteration, we emphasize a connection with (multistep) value iteration (see Section 4).

The λ-LSPE method, similar to TD(λ), generates an infinitely long trajectory $(i_0, i_1, \ldots)$ using a simulator. At each time t, it finds the solution $\tilde{r}_t$ of a least squares problem,

$$\tilde{r}_t = \arg\min_r \sum_{m=0}^{t} \left(\phi(i_m)'r - \phi(i_m)'r_t - \sum_{k=m}^{t} (\alpha\lambda)^{k-m} d_t(i_k, i_{k+1}) \right)^2, \tag{1.4}$$

and computes the new vector r_{t+1} according to

$$r_{t+1} = r_t + \gamma(\tilde{r}_t - r_t), \tag{1.5}$$

where γ is a positive stepsize. The initial weight vector r_0 is chosen independently of the trajectory $(i_0, i_1, \ldots)$.

It can be argued that λ-LSPE is a "scaled" version of TD(λ). In particular, from the analysis of Nedić and Bertsekas ([NeB03], p. 101; see also Section 3), it follows that the method takes the form

$$r_{t+1} = r_t + \gamma(\Phi' D \Phi)^{-1}(Ar_t + b) + \gamma(Z_t r_t + \zeta_t), \qquad t = 0, 1, \ldots, \tag{1.6}$$

where γ is a positive stepsize, and Z_t and ζ_t are some sequences of random matrices and vectors, respectively, that converge to 0 with probability 1. It was shown in [NeB03] that when the stepsize is diminishing rather than being constant, the method converges with probability 1 to the same limit as TD(λ), the unique solution r^* of the system $Ar + b = 0$ (convergence for a constant stepsize was conjectured but not proved).

One of the principal results of this paper is that the scaling matrix $(\Phi' D \Phi)^{-1}$ is "close" enough to $-A^{-1}$ so that, based also on the negative definiteness of A, the stepsize $\gamma = 1$ leads to convergence for all $\lambda \in [0, 1]$, i.e., the matrix $I + (\Phi' D \Phi)^{-1} A$ has eigenvalues that are within

the unit circle of the complex plane. In fact, we can see that A may be written in the alternative form

$$A = \Phi' D(M - I)\Phi, \qquad M = (1 - \lambda) \sum_{m=0}^{\infty} \lambda^m (\alpha P)^{m+1},$$

so that for $\lambda = 1$, the eigenvalues of $I + (\Phi' D\Phi)^{-1} A$ are all equal to 0. We will also show that as λ decreases towards 0, the region where the eigenvalues of $I + (\Phi' D\Phi)^{-1} A$ lie expands, but stays within the interior of the unit circle.

By comparing the iterations (1.3) and (1.6), we see that TD(λ) and λ-LSPE have a common structure – a deterministic linear iteration plus noise that tends to 0 with probability 1. However, the convergence rate of the deterministic linear iteration is geometric in the case of λ-LSPE, while it is slower than geometric in the case of TD(λ), because the stepsize γ_t must be diminishing. This indicates that λ-LSPE has a significant rate of convergence advantage over TD(λ). At the same time, with a recursive Kalman filter-like implementation discussed in [NeB03], λ-LSPE does not require much more overhead per iteration than TD(λ) [the associated matrix inversion at each iteration requires only $O(s^2)$ computation using the results of the inversion at the preceding iteration, where s is the dimension of r].

For some further insight on the relation of λ-LSPE with $\gamma = 1$ and TD(λ), let us focus on the case where $\lambda = 0$. TD(0) has the form

$$r_{t+1} = r_t + \gamma_t \phi(i_t) d_t(i_t, i_{t+1}), \tag{1.7}$$

while 0-LSPE has the form

$$r_{t+1} = \arg\min_r \sum_{m=0}^{t} \big(\phi(i_m)' r - \phi(i_m)' r_t - d_t(i_m, i_{m+1})\big)^2 \tag{1.8}$$

[cf. Eq. (1.4)]. We note that the gradient of the least squares sum above is

$$-2 \sum_{m=0}^{t} \phi(i_m) d_t(i_m, i_{m+1}).$$

Asymptotically, in steady-state, the expected values of all the terms in this sum are equal, and each is proportional to the expected value of the term $\phi(i_t) d_t(i_t, i_{t+1})$ in the TD(0) iteration (1.7). Thus, TD(0) *updates r_t along the gradient of the least squares sum of 0-LSPE, plus stochastic noise that asymptotically has zero mean.* This interpretation also holds for other values of $\lambda \neq 0$, as will be discussed in Section 4.

Another class of temporal difference methods, parameterized by $\lambda \in [0, 1]$, has been introduced by Boyan [Boy02], following the work by Bradtke and Barto [BrB96] who considered the case $\lambda = 0$. These methods, known as Least Squares TD (LSTD), also employ least squares and

have guaranteed convergence to the same limit as TD(λ) and λ-LSPE, as shown by Bradtke and Barto [BrB96] for the case $\lambda = 0$, and by Nedić and Bertsekas [BeN03] for the case $\lambda \in (0, 1]$. Konda [Kon02] has derived the asymptotic mean squared error of a class of recursive and non-recursive temporal difference methods [including TD(λ) and LSTD, but not including LSPE], and has found that LSTD has optimal asymptotic convergence rate within this class. The LSTD method is not iterative, but instead it evaluates the simulation-based estimates A_t and b_t of $(t+1)A$ and $(t+1)b$, given by

$$A_t = \sum_{m=0}^{t} z_m\big(\alpha\phi(i_{m+1})' - \phi(i_m)'\big), \qquad b_t = \sum_{m=0}^{t} z_m g(i_m, i_{m+1}), \qquad z_m = \sum_{k=0}^{m} (\alpha\lambda)^{m-k}\phi(i_k),$$

(see Section 3), and estimates the solution r^* of the system $Ar + b = 0$ by

$$\hat{r}_{t+1} = -A_t^{-1} b_t.$$

We argue in Section 5 that LSTD and λ-LSPE have comparable asymptotic performance, although there are significant differences in the early iterations. In fact, the iterates of LSTD and λ-LSPE converge to each other faster than they converge to r^*. Some insight into the comparability of the two methods can be obtained by verifying that the LSTD estimate $\hat{r}_{t+1}$ is also the unique vector $\hat{r}$ satisfying

$$\hat{r} = \arg\min_{r} \sum_{m=0}^{t} \left(\phi(i_m)'r - \phi(i_m)'\hat{r} - \sum_{k=m}^{t} (\alpha\lambda)^{k-m} \hat{d}(i_k, i_{k+1}; \hat{r}) \right)^2, \tag{1.9}$$

where

$$\hat{d}(i_k, i_{k+1}; \hat{r}) = g(i_k, i_{k+1}) + \alpha\phi(i_{k+1})'\hat{r} - \phi(i_k)'\hat{r}.$$

While finding $\hat{r}$ that satisfies Eq. (1.9) is not a least squares problem, its similarity with the least squares problem solved by LSPE [cf. Eq. (1.4)] is evident.

We note, however, that LSTD and LSPE may differ substantially in the early iterations. Furthermore, LSTD is a pure simulation method that cannot take advantage of a good initial choice r_0. This is a significant factor in favor of λ-LSPE in a major context, namely optimistic policy iteration [BeT96], where the policy used is changed (using a policy improvement mechanism) after a few simulated transitions. Then, the use of the latest estimate of r to start the iterations corresponding to a new policy, as well as a small stepsize (to damp oscillatory behavior following a change to a new policy) is essential for good overall performance.

The algorithms and analysis of the present paper, in conjunction with existing research, support a fairly comprehensive view of temporal difference methods with linear function approximation. The highlights of this view are as follows:

(1) Temporal difference methods fundamentally emulate value iteration methods that aim to solve a Bellman equation that corresponds to a multiple-transition version of the given Markov chain, and depends on λ (see Section 4).

(2) The emulation of the kth value iteration is approximate through linear function approximation, and solution of the least squares approximation problem (1.4) that involves the simulation data $(i_0, i_1, \ldots, i_t)$ up to time t.

(3) The least squares problem (1.4) is fully solved at time t by λ-LSPE, but is solved only approximately, by a single gradient iteration (plus zero-mean noise), by TD(λ) (see Section 4).

(4) LSPE and LSTD have similar asymptotic performance, but may differ substantially in the early iterations. Furthermore, LSPE can take advantage of good initial estimates of r^*, while LSTD, as presently known, cannot.

The paper is organized as follows. In Section 2, we derive a basic lemma regarding the location of the eigenvalues of the matrix $I + (\Phi' D \Phi)^{-1} A$. In Section 3, we use this lemma to show convergence of λ-LSPE with probability 1 for any stepsize γ in a range that includes $\gamma = 1$. In Section 4, we derive the connection of λ-LSPE with various forms of approximate value iteration. Based on this connection, we discuss how our line of analysis extends to other types of dynamic programming problems. In Section 5, we discuss the relation between λ-LSPE and LSTD. Finally, in Section 6 we present computational results showing that λ-LSPE is dramatically faster than TD(λ), and also simpler because it does not require any parameter tuning for the stepsize selection method.

2. PRELIMINARY ANALYSIS

In this section we prove some lemmas relating to the transition probability matrix P, the feature matrix Φ, and the associated matrices D and A of Eqs. (1.2) and (1.1). We denote by R and C the set of real and complex numbers, respectively, and by R^n and C^n the spaces of n-dimensional vectors with real and with complex components, respectively. The complex conjugate of a complex number z is denoted $\hat{z}$. The complex conjugate of a vector $z \in C^n$, is the vector whose components are the complex conjugates of the components of z, and is denoted $\hat{z}$. The modulus $\sqrt{\hat{z}z}$ of a complex number z is denoted by $|z|$. We consider two norms on C^n, the

standard norm, defined by

$$\|z\| = (\hat{z}'z)^{1/2} = \left(\sum_{i=1}^{n} |z_i|^2 \right)^{1/2}, \qquad \forall\ z = (z_1, \ldots, z_n) \in C^n,$$

and the weighted norm, defined by

$$\|z\|_D = (\hat{z}'Dz)^{1/2} = \left(\sum_{i=1}^{n} \pi(i)|z_i|^2 \right)^{1/2}, \qquad \forall\ z = (z_1, \ldots, z_n) \in C^n.$$

The following lemma extends, from $\Re^n$ to C^n, a basic result of Tsitsiklis and Van Roy [TsV97].

Lemma 2.1: For all $z \in C^n$, we have $\|Pz\|_D \leq \|z\|_D$.

Proof: For any $z = (z_1, \ldots, z_n) \in C^n$, we have, using the defining property $\sum_{i=1}^{n} \pi(i)p_{ij} = \pi(j)$ of the steady-state probabilities,

$$\begin{aligned}
\|Pz\|_D^2 &= \hat{z}'P'DPz \\
&= \sum_{i=1}^{n} \pi(i) \left(\sum_{j=1}^{n} p_{ij}\hat{z}_j \right) \left(\sum_{j=1}^{n} p_{ij}z_j \right) \\
&\leq \sum_{i=1}^{n} \pi(i) \left(\sum_{j=1}^{n} p_{ij}|z_j| \right)^2 \\
&\leq \sum_{i=1}^{n} \pi(i) \sum_{j=1}^{n} p_{ij}|z_j|^2 \\
&= \sum_{j=1}^{n} \sum_{i=1}^{n} \pi(i)p_{ij}|z_j|^2 \\
&= \sum_{j=1}^{n} \pi(j)|z_j|^2 \\
&= \|z\|_D^2,
\end{aligned}$$

where the first inequality follows since $\hat{x}y + x\hat{y} \leq 2|x|\,|y|$ for any two complex numbers x and y, and the second inequality follows by applying Jensen's inequality. **Q.E.D.**

The next lemma is the key to the convergence proof of the next section.

Lemma 2.2: The eigenvalues of the matrix $I + (\Phi'D\Phi)^{-1}A$ lie within the circle of radius $\alpha(1-\lambda)/(1-\alpha\lambda)$.

Proof: We have

$$A = \Phi'D(M - I)\Phi,$$

where

$$M = (1-\lambda) \sum_{m=0}^{\infty} \lambda^m (\alpha P)^{m+1},$$

so that

$$(\Phi' D\Phi)^{-1}A = (\Phi' D\Phi)^{-1}\Phi' DM\Phi - I.$$

Hence

$$I + (\Phi' D\Phi)^{-1}A = (\Phi' D\Phi)^{-1}\Phi' DM\Phi.$$

Let β be an eigenvalue of $I + (\Phi' D\Phi)^{-1}A$ and let z be a corresponding eigenvector, so that

$$(\Phi' D\Phi)^{-1}\Phi' DM\Phi z = \beta z.$$

Letting

$$W = \sqrt{D}\Phi,$$

we have

$$(W'W)^{-1}W'\sqrt{D}M\Phi z = \beta z,$$

from which, by left-multiplying with W, we obtain

$$W(W'W)^{-1}W'\sqrt{D}M\Phi z = \beta W z. \tag{2.1}$$

The norm of the right-hand side of Eq. (2.1) is

$$\|\beta W z\| = |\beta|\,\|Wz\| = |\beta|\,\sqrt{\hat{z}\Phi' D\Phi z} = |\beta|\,\|\Phi z\|_D. \tag{2.2}$$

To estimate the norm of the left-hand side of Eq. (2.1), first note that

$$\|W(W'W)^{-1}W'\sqrt{D}M\Phi z\| \le \|W(W'W)^{-1}W'\|\,\|\sqrt{D}M\Phi z\| = \|W(W'W)^{-1}W'\|\,\|M\Phi z\|_D,$$

and then note also that $W(W'W)^{-1}W'$ is a projection matrix [i.e., for $x \in \Re^n$, $W(W'W)^{-1}W'x$ is the projection of x on the subspace spanned by the columns of W], so that $\|W(W'W)^{-1}W'x\| \le \|x\|$, from which

$$\|W(W'W)^{-1}W'\| \le 1.$$

Thus we have

$$\begin{aligned}
\|W(W'W)^{-1}W'\sqrt{D}M\Phi z\| &\le \|M\Phi z\|_D \\
&= \left\| (1-\lambda)\sum_{m=0}^{\infty} \lambda^m \alpha^{m+1} P^{m+1}\Phi z \right\|_D \\
&\le (1-\lambda)\sum_{m=0}^{\infty} \lambda^m \alpha^{m+1} \|P^{m+1}\Phi z\|_D \\
&\le (1-\lambda)\sum_{m=0}^{\infty} \lambda^m \alpha^{m+1} \|\Phi z\|_D \\
&= \frac{\alpha(1-\lambda)}{1-\alpha\lambda}\|\Phi z\|_D,
\end{aligned} \tag{2.3}$$

where the last inequality follows by repeated use of Lemma 2.1. By comparing Eqs. (2.3) and (2.2), and by taking into account that $\Phi z \neq 0$ (since Φ has full rank), we see that

$$|\beta| \leq \frac{\alpha(1-\lambda)}{1-\alpha\lambda}.$$

Q.E.D.

3. CONVERGENCE ANALYSIS

We will now use Lemma 2.2 to prove the convergence of λ-LSPE. It is shown in Nedić and Bertsekas [NeB03] that the method is given by

$$r_{t+1} = r_t + \gamma B_t^{-1}(A_t r_t + b_t), \qquad \forall\ t, \tag{3.1}$$

where

$$B_t = \sum_{m=0}^{t} \phi(i_m)\phi(i_m)', \qquad A_t = \sum_{m=0}^{t} z_m\big(\alpha\phi(i_{m+1})' - \phi(i_m)'\big), \tag{3.2}$$

$$b_t = \sum_{m=0}^{t} z_m g(i_m, i_{m+1}), \qquad z_m = \sum_{k=0}^{m} (\alpha\lambda)^{m-k}\phi(i_k). \tag{3.3}$$

[Note that if in the early iterations, $\sum_{m=0}^{t} \phi(i_m)\phi(i_m)'$ is not invertible, we may add to it a small positive multiple of the identity, or alternatively we may replace inverse by pseudoinverse. Such modifications are inconsequential and will be ignored in the subsequent analysis; see also [NeB03].] We can rewrite Eq. (3.1) as

$$r_{t+1} = r_t + \gamma \overline{B}_t^{-1}(\overline{A}_t r_t + \overline{b}_t), \qquad \forall\ t,$$

where

$$\overline{B}_t = \frac{B_t}{t+1}, \qquad \overline{A}_t = \frac{A_t}{t+1}, \qquad \overline{b}_t = \frac{b_t}{t+1}.$$

Using the analysis of [NeB03] (see the proof of Prop. 3.1, p. 108), it follows that with probability 1, we have

$$\overline{B}_t \to B, \qquad \overline{A}_t \to A, \qquad \overline{b}_t \to b,$$

where

$$B = \Phi' D \Phi,$$

and A and b are given by Eq. (1.1).

Thus, we may write iteration (3.1) as

$$r_{t+1} = r_t + \gamma(\Phi' D \Phi)^{-1}(A r_t + b) + \gamma(Z_t r_t + \zeta_t), \qquad t = 0, 1, \ldots, \tag{3.4}$$

where

$$Z_t = \overline{B}_t^{-1}\overline{A}_t - B^{-1}A, \qquad \zeta_t = \overline{B}_t^{-1}\overline{b}_t - B^{-1}b.$$

Furthermore, with probability 1, we have

$$Z_t \to 0, \qquad \zeta_t \to 0.$$

We are now ready to prove our convergence result.

Proposition 3.1: The sequence generated by the λ-LSPE method converges to $r^* = -A^{-1}b$ with probability 1, provided that the constant stepsize γ satisfies

$$0 < \gamma < \frac{2 - 2\alpha\lambda}{1 + \alpha - 2\alpha\lambda}.$$

Proof: If we write the matrix $I + \gamma(\Phi' D \Phi)^{-1}A$ as

$$(1-\gamma)I + \gamma\big(I + (\Phi' D \Phi)^{-1}A\big),$$

we see, using Lemma 2.2, that its eigenvalues lie within the circle that is centered at $1-\gamma$ and has radius

$$\frac{\gamma\alpha(1-\lambda)}{1-\alpha\lambda}.$$

It follows by a simple geometrical argument that this circle is strictly contained within the unit circle if and only if γ lies in the range between 0 and $(2-2\alpha\lambda)/(1+\alpha-2\alpha\lambda)$. Thus for each γ within this range, the spectral radius of $I + \gamma(\Phi' D\Phi)^{-1}A$ is less than 1, and there exists a norm $\|\cdot\|_w$ over $\Re^n$ and an $\epsilon > 0$ (depending on γ) such that

$$\|I + \gamma(\Phi' D\Phi)^{-1}A\|_w < 1 - \epsilon.$$

Using the equation $b = -Ar^*$, we can write the iteration (3.4) as

$$r_{t+1} - r^* = \big(I + \gamma(\Phi' D\Phi)^{-1}A + \gamma Z_t\big)(r_t - r^*) + \gamma(Z_t r^* + \zeta_t), \qquad t = 0, 1, \ldots.$$

For any simulated trajectory such that $Z_t \to 0$ and $\zeta_t \to 0$, there exists an index $\overline{t}$ such that

$$\|I + \gamma(\Phi' D\Phi)^{-1}A + \gamma Z_t\|_w < 1 - \epsilon, \qquad \forall\ t \geq \overline{t}.$$

Thus, for sufficiently large t, we have

$$\|r_{t+1} - r^*\|_w \leq (1-\epsilon)\|r_t - r^*\|_w + \gamma\|Z_t r^* + \zeta_t\|_w.$$

Since $Z_t r^* + \zeta_t \to 0$, it follows that $r_t - r^* \to 0$. Since the set of simulated trajectories such that $Z_t \to 0$ and $\zeta_t \to 0$ is a set of probability 1, it follows that $r_t \to r^*$ with probability 1. **Q.E.D.**

Note that as λ decreases, the range of stepsizes γ that lead to convergence is reduced. However, this range always contains the stepsize $\gamma = 1$.

4. RELATIONS BETWEEN λ-LSPE AND VALUE ITERATION

In this section, we will discuss a number of value iteration ideas, which underlie the structure of λ-LSPE. These connections become most apparent when the stepsize is constant and equal to 1 ($\gamma \equiv 1$), which we will assume in our discussion.

The Case $\lambda = 0$

The classical value iteration method for solving the given policy evaluation problem is

$$J_{t+1}(i) = \sum_{j=1}^{n} p_{ij}\big(g(i,j) + \alpha J_t(j)\big), \qquad i = 1, \ldots, n, \tag{4.1}$$

and by standard dynamic programming results, it converges to the cost-to-go function $J(i)$. We will show that approximate versions of this method are connected with three methods that are relevant to our discussion: TD(0), 0-LSPE, and the deterministic portion of the 0-LSPE iteration (3.4).

Indeed, a version of value iteration that uses linear function approximation of the form $J_t(i) \approx \phi(i)'r_t$ is to recursively select r_{t+1} so that $\phi(i)'r_{t+1}$ is uniformly (for all states i) "close" to $\sum_{j=1}^{n} p_{ij}\big(g(i,j) + \alpha J_t(j)\big)$; for example by solving a corresponding least squares problem

$$r_{t+1} = \arg\min_{r} \sum_{i=1}^{n} w(i) \left(\phi(i)'r - \sum_{j=1}^{n} p_{ij}\big(g(i,j) + \alpha\phi(j)'r_t\big) \right)^2, \qquad t = 0, 1, \ldots, \tag{4.2}$$

where $w(i)$, $i = 1, \ldots, n$, are some positive weights. This method is considered in Section 6.5.3 of Bertsekas and Tsitsiklis [BeT96], where it is pointed out that divergence is possible if the weights $w(i)$ are not properly chosen; for example if $w(i) = 1$ for all i. It can be seen that the TD(0) iteration (1.7) may be viewed as a one-sample approximation of the special case of iteration (4.2) where the weights are chosen as $w(i) = \pi(i)$, for all i, as discussed in Section 6.5.4 of [BeT96]. Furthermore, Tsitsiklis and Van Roy [TsV97] show that for TD(0) convergence, it is essential that state samples are collected in accordance with the steady-state probabilities $\pi(i)$. By using the definition of temporal difference to write the 0-LSPE iteration (1.8) as

$$r_{t+1} = \arg\min_{r} \sum_{m=0}^{t} \big(\phi(i_m)'r - g(i_m, i_{m+1}) - \alpha\phi(i_{m+1})'r_t\big)^2, \tag{4.3}$$

we can similarly interpret it as a multiple-sample approximation of iteration (4.2) with weights $w(i) = \pi(i)$. Of course, when $w(i) = \pi(i)$, the iteration (4.2) is not implementable since the $\pi(i)$ are unknown, and the only way to approximate it is through the on-line type of state sampling used in 0-LSPE and TD(0).

These interpretations suggest that the approximate value iteration method (4.2) should converge when the weights are chosen as $w(i) = \pi(i)$. Indeed for these weights, the method takes the form

$$r_{t+1} = \arg\min_r \|\Phi r - P(g + \alpha\Phi r_t)\|_D^2, \tag{4.4}$$

which after some calculation, is written as

$$r_{t+1} = r_t + (\Phi' D \Phi)^{-1}(Ar_t + b), \qquad t = 0, 1, \ldots, \tag{4.5}$$

where A and b are given by Eq. (1.1), for the case where $\lambda = 0$. In other words *the deterministic linear iteration portion of the 0-LSPE method with $\gamma = 1$ is equivalent to the approximate value iteration (4.2) with weights $w(i) = \pi(i)$*. Thus, *we can view 0-LSPE as the approximate value iteration method (4.2), plus noise that asymptotically tends to 0*.

Note that the approximate value iteration method (4.4) can be interpreted as a mapping from the feature subspace

$$S = \{\Phi r \mid r \in \Re^s\}$$

to itself: it maps the vector Φr_t to its value iterate $P(g + \alpha\Phi r_t)$, and then projects [with respect to the norm $\|\cdot\|_D$ corresponding to the steady-state probabilities/weights $\pi(i)$] the result on S, as discussed by Tsitsiklis and Van Roy [TsV97], who give an example of divergence when nonlinear function approximation is used. Related issues are discussed by de Farias and Van Roy [FaV00], who consider approximate value iteration with linear function approximation, but multiple policies.

Figure 4.1 illustrates the approximate value iteration method (4.4) together with 0-LSPE, which is the same iteration plus asymptotically vanishing simulation error.

Connection with Multistep Value Iteration

In the case where $\lambda \in (0, 1)$, a similar connection with approximate value iteration can be derived, except that *each value iteration involves multiple state transitions* (see also the corresponding discussion by Bertsekas and Ioffe [BeI96], and also Bertsekas and Tsitsiklis [BeT96], Section 2.3). In particular, for $M \geq 1$, let us consider the M-transition Bellman's equation

$$J(i) = E\left[\alpha^M J(i_M) + \sum_{k=0}^{M-1} \alpha^k g(i_k, i_{k+1})) \;\Big|\; i_0 = i\right], \qquad i = 1, \ldots, n. \tag{4.6}$$

This equation has the cost-to-go function J as its unique solution, and in fact may be viewed as Bellman's equation for a modified policy evaluation problem, involving a Markov chain where each transition corresponds to M transitions of the original, and the cost is calculated using a

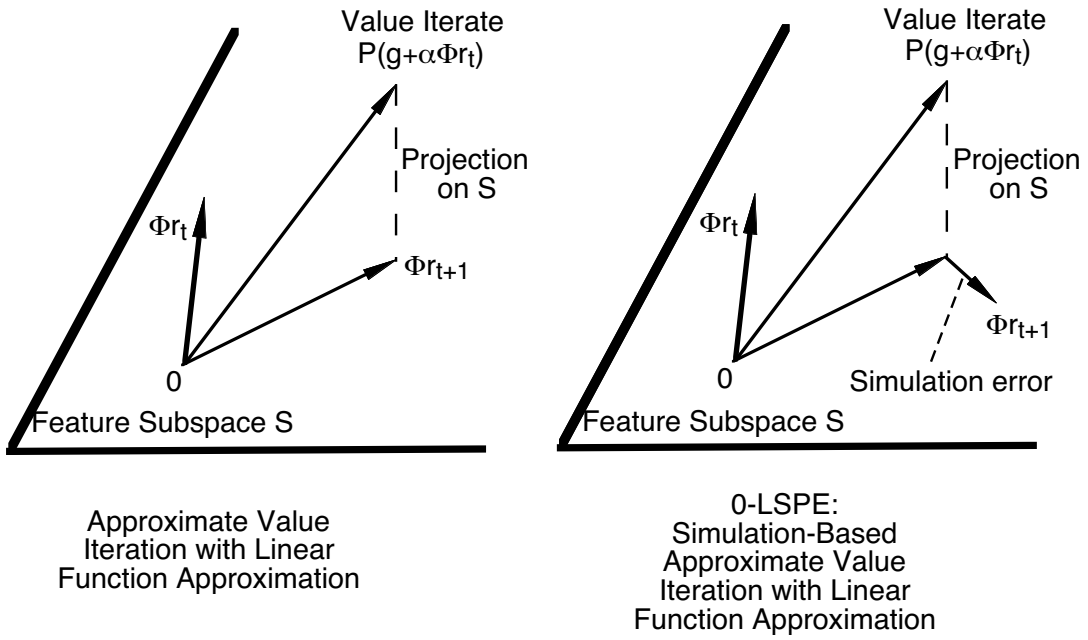

Figure 4.1. Geometric interpretation of 0-LSPE as the sum of the approximate value iterate (4.4) plus asymptotically vanishing simulation error.

discount factor α^M and a cost per (M-transition) stage equal to $\sum_{k=0}^{M-1} \alpha^k g(i_k, i_{k+1})$. The value iteration method corresponding to this modified problem is

$$J_{t+1}(i) = E\left[\alpha^M J_t(i_M) + \sum_{k=0}^{M-1} \alpha^k g(i_k, i_{k+1}) \;\Big|\; i_0 = i\right], \qquad i = 1, \ldots, n,$$

and can be seen to be equivalent to M iterations of the value iteration method (4.1) for the original problem. The corresponding simulation-based least-squares implementation is

$$r_{t+1} = \arg\min_r \sum_{m=0}^{t} \left(\phi(i_m)'r - \alpha^M \phi(i_{m+M})' r_t - \sum_{k=0}^{M-1} \alpha^k g(i_{m+k}, i_{m+k+1})\right)^2, \qquad t = 0, 1, \ldots,$$

or equivalently, using the definition of temporal difference,

$$r_{t+1} = \arg\min_r \sum_{m=0}^{t} \left(\phi(i_m)'r - \phi(i_m)' r_t - \sum_{k=m}^{m+M-1} \alpha^{k-m} d_t(i_k, i_{k+1})\right)^2, \qquad t = 0, 1, \ldots. \quad (4.7)$$

This method, which is identical to 0-LSPE for the modified policy evaluation problem described above, may be viewed as intermediate between 0-LSPE and 1-LSPE for the original policy evaluation problem; compare with the form (1.4) of λ-LSPE for $\lambda = 0$ and $\lambda = 1$.

Let us also mention the incremental gradient version of the iteration (4.7), given by

$$r_{t+1} = r_t + \gamma_t \, \phi(i_t) \sum_{k=t}^{t+M-1} \alpha^{k-t} d_t(i_k, i_{k+1}), \qquad t = 0, 1, \ldots. \quad (4.8)$$

This method, which is identical to TD(0) for the modified (M-step) policy evaluation problem described above, may be viewed as intermediate between TD(0) and TD(1) [it is closest to TD(0) for small M, and to TD(1) for large M]. Note that temporal differences do not play a fundamental role in the above iterations; they just provide a convenient shorthand notation that simplifies the formulas.

The Case $0 < \lambda < 1$

The M-transition Bellman's equation (4.6) holds for a fixed M, but it is also possible to consider a version of Bellman's equation where M is random and geometrically distributed with parameter λ, i.e.,

$$\text{Prob}(M = m) = (1-\lambda)\lambda^{m-1}, \qquad m = 1, 2, \ldots$$

This equation is obtained by multiplying both sides of Eq. (4.6) with $(1-\lambda)\lambda^{m-1}$, for each m, and adding over m:

$$J(i) = \sum_{m=1}^{\infty} (1-\lambda)\lambda^{m-1} E\left[\alpha^m J(i_m) + \sum_{k=0}^{m-1} \alpha^k g(i_k, i_{k+1})) \;\middle|\; i_0 = i\right], \qquad i = 1, \ldots, n. \tag{4.9}$$

Tsitsiklis and Van Roy [TsV97] provide an interpretation of TD(λ) as a gradient-like method for minimizing a weighted quadratic function of the error in satisfying this equation.

We may view Eq. (4.9) as Bellman's equation for a modified policy evaluation problem. The value iteration method corresponding to this modified problem is

$$J_{t+1}(i) = \sum_{m=1}^{\infty} (1-\lambda)\lambda^{m-1} E\left[\alpha^m J_t(i_m) + \sum_{k=0}^{m-1} \alpha^k g(i_k, i_{k+1})) \;\middle|\; i_0 = i\right], \qquad i = 1, \ldots, n,$$

which can be written as

$$\begin{aligned} J_{t+1}(i) &= J_t(i) + (1-\lambda) \sum_{m=1}^{\infty} \sum_{k=0}^{m-1} \lambda^{m-1} \alpha^k E\big[g(i_k, i_{k+1}) + \alpha J_t(i_{k+1}) - J_t(i_k) \mid i_0 = i\big] \\ &= J_t(i) + (1-\lambda) \sum_{k=0}^{\infty} \left(\sum_{m=k+1}^{\infty} \lambda^{m-1} \right) \alpha^k E\big[g(i_k, i_{k+1}) + \alpha J_t(i_{k+1}) - J_t(i_k) \mid i_0 = i\big] \end{aligned}$$

and finally,

$$J_{t+1}(i) = J_t(i) + \sum_{k=0}^{\infty} (\alpha\lambda)^k E\big[g(i_k, i_{k+1}) + \alpha J_t(i_{k+1}) - J_t(i_k) \mid i_0 = i\big], \qquad i = 1, \ldots, n.$$

By using the linear function approximation $\phi(i)'r_t$ for the costs $J_t(i)$, and by replacing the terms $g(i_k, i_{k+1}) + \alpha J_t(i_{k+1}) - J_t(i_k)$ in the above iteration with temporal differences

$$d_t(i_k, i_{k+1}) = g(i_k, i_{k+1}) + \alpha\phi(i_{k+1})'r_t - \phi(i_k)'r_t,$$

we obtain the simulation-based least-squares implementation

$$r_{t+1} = \arg\min_r \sum_{m=0}^{t} \left(\phi(i_m)'r - \phi(i_m)'r_t - \sum_{k=m}^{t} (\alpha\lambda)^{k-m} d_t(i_k, i_{k+1}) \right)^2, \tag{4.10}$$

which is in fact λ-LSPE with stepsize $\gamma = 1$.

Let us now discuss the relation of λ-LSPE with $\gamma = 1$ and TD(λ). We note that the gradient of the least squares sum of λ-LSPE is

$$-2 \sum_{m=0}^{t} \phi(i_m) \sum_{k=m}^{t} (\alpha\lambda)^{k-m} d_t(i_k, i_{k+1}).$$

This gradient after some calculation, can be written as

$$-2\big(z_0 d_t(i_0, i_1) + \cdots + z_t d_t(i_t, i_{t+1})\big), \tag{4.11}$$

where

$$z_k = \sum_{m=0}^{k} (\alpha\lambda)^{k-m} \phi(i_m), \qquad k = 0, \ldots, t,$$

[cf. Eq. (3.3)]. On the other hand, TD(λ) has the form

$$r_{t+1} = r_t + \gamma_t z_t d_t(i_t, i_{t+1}).$$

Asymptotically, in steady-state, the expected values of all the terms $z_m d_t(i_m, i_{m+1})$ in the gradient sum (4.11) are equal, and each is proportional to the expected value of the term $z_t d_t(i_t, i_{t+1})$ in the TD(λ) iteration. Thus, TD(λ) *updates r_t along the gradient of the least squares sum of λ-LSPE, plus stochastic noise that asymptotically has zero mean.*

In conclusion, for all $\lambda < 1$, we can view λ-LSPE with $\gamma = 1$ as a least squares-based approximate value iteration with linear function approximation. However, each value iteration implicitly involves a random number of transitions with geometric distribution that depends on λ. The limit r^* depends on λ because the underlying Bellman's equation also depends on λ. Furthermore, TD(λ) and λ-LSPE may be viewed as stochastic gradient and Kalman filtering algorithms, respectively, for solving the least squares problem associated with approximate value iteration.

Generalizations Based on Other Types of Value Iteration

The connection with value iteration described above provides a guideline for developing other least squares-based approximation methods, relating to different types of dynamic programming problems, such as stochastic shortest path, average cost, and semi-Markov decision problems,

or to variants of value iteration such as for example Gauss-Seidel methods. To this end, we generalize the key idea of the convergence analysis of Sections 2 and 3. A proof of the following proposition is embodied in the argument of the proof of Prop. 6.9 of Bertsekas and Tsitsiklis [BeT96] (which actually deals with a more general nonlinear iteration), but for completeness, we give an independent argument that uses the proof of Lemma 2.2.

Proposition 4.1: Consider a linear iteration of the form

$$x_{t+1} = Gx_t + g, \qquad t = 0, 1, \ldots, \tag{4.12}$$

where $x_t \in \Re^n$, and G and g are given $n \times n$ matrix and n-dimensional vector, respectively. Assume that D is a positive definite symmetric matrix such that

$$\|G\|_D = \max_{\substack{\|z\|_D \leq 1 \\ z \in C^n}} \|Gz\|_D < 1,$$

where $\|z\|_D = \sqrt{\hat{z}' D z}$, for all $z \in C^n$. Let Φ be an $n \times s$ matrix of rank s. Then the iteration

$$r_{t+1} = \arg\min_{r \in \Re^s} \|\Phi r - G\Phi r_t - g\|_D, \qquad t = 0, 1, \ldots \tag{4.13}$$

converges to the vector r^* satisfying

$$r^* = \arg\min_{r \in \Re^s} \|\Phi r - G\Phi r^* - g\|_D, \tag{4.14}$$

from every starting point $r_0 \in \Re^s$.

Proof: The iteration (4.13) can be written as

$$r_{t+1} = (\Phi' D \Phi)^{-1} (\Phi' D G \Phi r_t + \Phi' D g), \tag{4.15}$$

so it is sufficient to show that the matrix $(\Phi' D \Phi)^{-1} \Phi' D G \Phi$ has eigenvalues that lie within the unit circle. The proof of this follows nearly verbatim the corresponding steps of the proof of Lemma 2.2. If r^* is the limit of r_t, we have by taking limit in Eq. (4.15),

$$r^* = \big(I - (\Phi' D \Phi)^{-1} \Phi' D G\big)^{-1} (\Phi' D \Phi)^{-1} \Phi' D g.$$

It can be verified that r^* as given by the above equation, also satisfies Eq. (4.14). **Q.E.D.**

The above proposition can be used within various dynamic programming/function approximation contexts. In particular, starting with a value iteration of the form (4.12), we can consider a linear function approximation version of the form (4.13), as long as we can find a weighted Euclidean norm $\|\cdot\|_D$ such that $\|G\|_D < 1$. We may then try to devise a simulation-based method

that emulates approximately iteration (4.13), similar to λ-LSPE. This method will be an iterative stochastic algorithm, and its convergence may be established along the lines of the proof of Prop. 3.1. Thus, Prop. 4.1 provides a general framework for deriving and analyzing least-squares simulation-based methods in approximate dynamic programming. An example of such a method, indeed the direct analog of λ-LSPE for stochastic shortest path problems, was stated and used by Bertsekas and Ioffe [BeI96] to solve the tetris training problem [see also [BeT96], Eq. (8.6)].

5. RELATION BETWEEN λ-LSPE AND LSTD

We now discuss the relation between λ-LSPE and the LSTD method that estimates $r^* = -A^{-1}b$ based on the portion $(i_0, \ldots, i_t)$ of the simulation trajectory by

$$\hat{r}_{t+1} = -A_t^{-1} b_t,$$

[cf. Eqs. (3.2) and (3.3)]. Konda [Kon02] has shown that the error covariance $E\{(\hat{r}_t - r^*)(\hat{r}_t - r^*)'\}$ of LSTD goes to zero at the rate of $1/t$. Similarly, it was shown by Nedić and Bertsekas [NeB03] that the covariance of the stochastic term $Z_t r_t + \zeta_t$ in Eq. (4.5) goes to zero at the rate of $1/t$. Thus, from Eq. (4.5), we see that the error covariance $E\{(r_t - r^*)(r_t - r^*)'\}$ of λ-LSPE also goes to zero at the rate of $1/t$.

We will now argue that a stronger result holds, namely that r_t "tracks" $\hat{r}_t$ in the sense that the difference $r_t - \hat{r}_t$ converges to 0 faster than $\hat{r}_t - r^*$. Indeed, from Eqs. (3.2) and (3.3), we see that the averages $\overline{B}_t$, $\overline{A}_t$, and $\overline{b}_t$ are generated by the slow stochastic approximation-type iterations

$$\overline{B}_{t+1} = \overline{B}_t + \frac{1}{t+2}\big(\phi(i_{t+1})\phi(i_{t+1})' - \overline{B}_t\big),$$

$$\overline{A}_{t+1} = \overline{A}_t + \frac{1}{t+2}\big(z_{t+1}\big(\alpha\phi(i_{t+2})' - \phi(i_{t+1})'\big) - \overline{A}_t\big), \tag{5.1}$$

$$\overline{b}_{t+1} = \overline{b}_t + \frac{1}{t+2}\big(z_{t+1} g(i_{t+1}, i_{t+2}) - \overline{b}_{t+1}\big). \tag{5.2}$$

Thus, they converge at a slower time scale than the λ-LSPE iteration

$$r_{t+1} = r_t + \overline{B}_t^{-1}(\overline{A}_t r_t + \overline{b}_t), \tag{5.3}$$

where, for sufficiently large t, the matrix $I + \overline{B}_t^{-1}\overline{A}_t$ has eigenvalues within the unit circle, inducing much larger relative changes of r_t. This means that the λ-LSPE iteration (5.3) "sees $\overline{B}_t$, $\overline{A}_t$, and $\overline{b}_t$ as essentially constant," so that, for large t, r_{t+1} is essentially equal to the corresponding limit of iteration (5.3) with $\overline{B}_t$, $\overline{A}_t$, and $\overline{b}_t$ held fixed. This limit is $-\overline{A}_t^{-1}\overline{b}_t$ or $\hat{r}_{t+1}$. It follows that the

difference $r_t - \hat{r}_t$ converges to 0 faster than $\hat{r}_t - r^*$. The preceding argument can be made precise by appealing to the theory of two-time scale iterative methods (see e.g., Benveniste, Metivier, and Priouret [BMP90]), but a detailed analysis is beyond the scope of this paper.

Despite their similar asymptotic behavior, the methods may differ substantially in the early iterations, and it appears that the iterates of LSTD tend to fluctuate more than those of λ-LSPE. Some insight into this behavior may be obtained by noting that the λ-LSPE iteration consists of a deterministic component that converges fast, and a stochastic component that converges slowly, so in the early iterations, the deterministic component dominates the stochastic fluctuations. On the other hand, $\overline{A}_t$ and $\overline{b}_t$ are generated by the slow iterations (5.1) and (5.2), and the corresponding estimate $-\overline{A}_t^{-1}\overline{b}_t$ of LSTD fluctuates significantly in the early iterations.

Another significant factor in favor of LSPE is that LSTD cannot take advantage of a good initial choice r_0. This is important in contexts such as optimistic policy iteration, as discussed in the introduction. Figure 5.1 shows some typical computational results for two 100-state problems with four features, and the values $\lambda = 0$ and $\lambda = 1$. The four features are

$$\phi_1(i) = 1, \qquad \phi_2(i) = i, \qquad \phi_3(i) = I([81, 90]), \qquad \phi_4(i) = I([91, 100]),$$

where $I(S)$ denotes the indicator function of a set S [$I(i) = 1$ if $i \in S$, and $I(i) = 0$ if $i \notin S$].

The figure shows the sequence of the parameter values $r(1)$ over 1,000 iterations/simulated transitions, for three methods: LSTD, LSPE with a constant stepsize $\gamma = 1$, and LSPE with a time-varying stepsize given by

$$\gamma_t = \frac{t}{500 + t}.$$

While all three methods asymptotically give the same results, it appears that LSTD oscillates more that LSPE in the initial iterations. The use of the time-varying stepsize "damps" the noisy behavior in the early iterations.

6. COMPUTATIONAL COMPARISON OF λ-LSPE AND TD(λ)

We conducted some computational experimentation to compare the performance of λ-LSPE and TD(λ). Despite the fact that our test problems were small, the differences between the two methods emerged strikingly and unmistakably. The methods performed as expected from the existing theoretical analysis, and converged to the same limit. In summary, the major observed differences between the two methods are:

(1) The number of iterations (length of simulation) to converge within the same small neighborhood of r^* was dramatically smaller for λ-LSPE than for TD(λ). Interestingly, not only

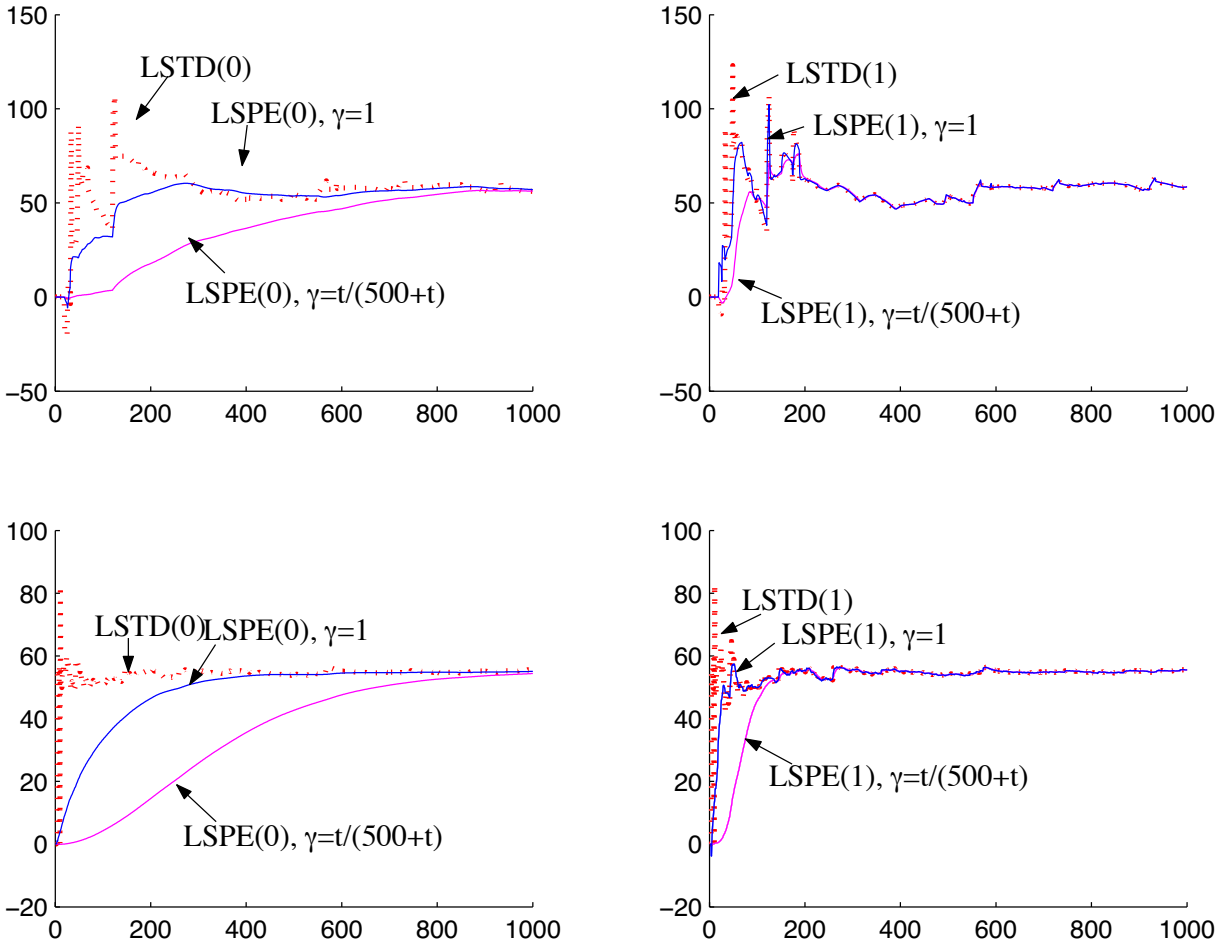

Figure 5.1. The sequence of the parameter values $r(1)$ over 1,000 iterations/simulated transitions, for three methods: LSTD, LSPE with a constant stepsize $\gamma = 1$, and LSPE with a time-varying stepsize. The top figures correspond to a "slow-mixing" Markov chain (high self-transition probabilities) of the form

$$P = 0.9 * P_{random} + 0.1I,$$

where I is the identity and P_{random} is a matrix whose row elements were generated as uniformly distributed random numbers within $[0, 1]$, and were normalized so that they add to 1. The bottom figures correspond to a "fast-mixing" Markov chain (low self-transition probabilities):

$$P = 0.1 * P_{random} + 0.9I.$$

The cost of a transition was randomly chosen within $[0, 1]$ at every state i, plus $i/30$ for self-transitions for $i \in [90, 100]$.

was the deterministic portion of the λ-LSPE iteration much faster, but the noisy portion was faster as well, for all the stepsize rules that we tried for TD(λ).

(2) While in λ-LSPE there is no need to choose any parameters (we fixed the stepsize to $\gamma = 1$), in TD(λ) the choice of the stepsize γ_t was λ-dependent, and required a lot of trial and error to obtain reasonable performance.

(3) Because of the faster convergence and greater resilience to simulation noise of λ-LSPE, it is possible to use values of λ that are closer to 1 than with TD(λ), thereby obtaining vectors Φr^* that more accurately approximate the true cost vector J.

The observed superiority of λ-LSPE over TD(λ) is based on the much faster convergence rate of its deterministic portion. On the other hand, for many problems the noisy portion of the iteration may dominate the computation, such as for example when the Markov chain is "slow-mixing," and a large number of transitions are needed for the simulation to reach all the important parts of the state space. Then, both methods may need a very long simulation trajectory in order to converge. Our experiments suggest much better performance for λ-LSPE under these circumstances as well, but were too limited to establish any kind of solid conclusion. However, in such cases, the optimality result for LSTD of Konda (see Section 1), and comparability of the behavior of LSTD and λ-LSPE, suggest a substantial superiority of λ-LSPE over TD(λ).

We will present representative results for a simple test problem with three states $i = 1, 2, 3$, and two features, corresponding to a linear approximation architecture of the form

$$\tilde{J}(i, r) = r(1) + ir(2), \qquad i = 1, 2, 3,$$

where $r(1)$ and $r(2)$ were the components of r. Because the problem is small, we can state it precisely here, so that our experiments can be replicated by others. We obtained qualitatively similar results with larger problems, involving 10 states and two features, and 100 states and four features. We also obtained similar results in limited tests involving the M-step methods (4.7) and (4.8).

We tested λ-LSPE and TD(λ) for a variety of problem data, experimental conditions, and values of λ. Figure 5.1 shows some results where the transition probability and cost matrices are given by

$$[p_{ij}] = \begin{pmatrix} 0.01 & 0.99 & 0 \\ 0.55 & 0.01 & 0.44 \\ 0 & 0.99 & 0.01 \end{pmatrix}, \qquad [g(i,j)] = \begin{pmatrix} 1 & 2 & 0 \\ 1 & 2 & -1 \\ 0 & 1 & 0 \end{pmatrix}.$$

The discount factor was $\alpha = 0.99$. The initial condition was $r_0 = (0, 0)$. The stepsize for λ-LSPE was chosen to be equal to 1 throughout. The stepsize choice for TD(λ) required quite a bit of

trial and error, aiming to balance speed of convergence and stochastic oscillatory behavior. We obtained the best results with three different stepsize rules

$$\gamma_t = \frac{16(1-\alpha\lambda)}{500(1-\alpha\lambda)+t}, \tag{6.1}$$

$$\gamma_t = \frac{16(1-\alpha\lambda)\sqrt{\log(t)}}{500(1-\alpha\lambda)+t}, \tag{6.2}$$

$$\gamma_t = \frac{16(1-\alpha\lambda)\log(t)}{500(1-\alpha\lambda)+t}. \tag{6.3}$$

Rule (6.1) led to the slowest convergence with least stochastic oscillation, while rule (6.3) led to the fastest convergence with most stochastic oscillation.

It can be seen from Fig. 5.2 that TD(λ) is not settled after 20,000 iterations/simulated transitions, and in the case where $\lambda = 1$, it does not even show signs of convergence. By contrast, λ-LSPE essentially converges within no more than 500 iterations, and with small subsequent stochastic oscillation. Generally, as λ becomes smaller, both TD(λ) and λ-LSPE converge faster at the expense of a worse bound on the error $\Phi r^* - J$. The qualitative behavior, illustrated in Fig. 5.2, was replicated for a variety of transition probability and cost matrices, initial conditions, and other experimental conditions. This behavior is consistent with the computational results of Bertsekas and Ioffe [BeI96] for the tetris training problem (see also Bertsekas and Tsitsiklis [BeT96], Section 8.3). Furthermore, in view of the similarity of performance of λ-LSPE and LSTD, our computational experience is also consistent with that of Boyan [Boy02].

7. REFERENCE

[BMP90] Benveniste, A., Metivier, M., and Priouret, P., Adaptive Algorithms and Stochastic Approximations, Springer-Verlag, N. Y., 1990.

[BeI96] Bertsekas, D. P., and Ioffe, S., "Temporal Differences-Based Policy Iteration and Applications in Neuro-Dynamic Programming," Lab. for Info. and Decision Systems Report LIDS-P-2349, MIT, Cambridge, MA, 1996.

[Ber01] Bertsekas, D. P., Dynamic Programming and Optimal Control, 2nd edition, Athena Scientific, Belmont, MA, 2001.

[BeT96] Bertsekas, D. P., and Tsitsiklis, J. N., Neuro-Dynamic Programming, Athena Scientific, Belmont, MA, 1996.

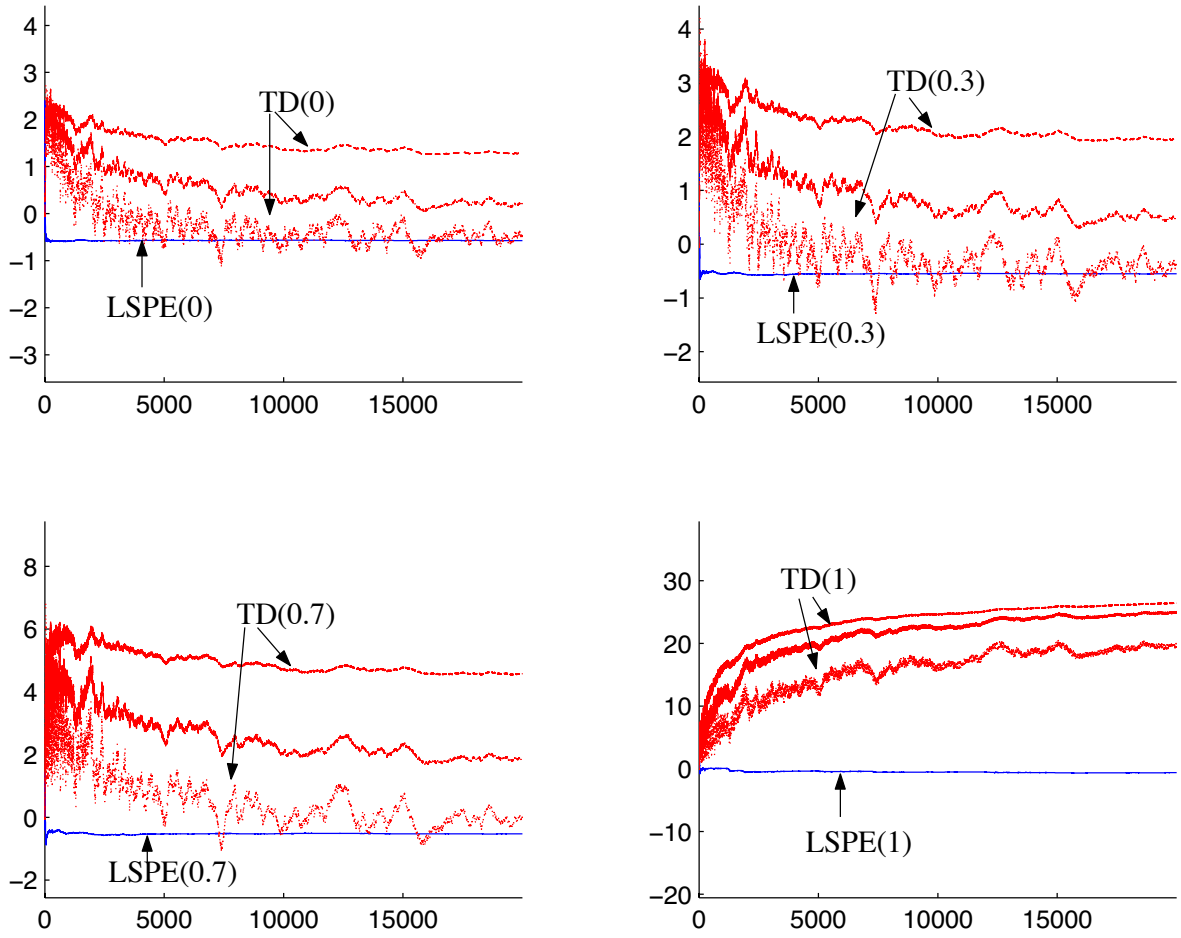

Figure 5.2. The sequence of the parameter values $r(2)$ generated by λ-LSPE and TD(λ) [using the three stepsize rules (6.1)-(6.3)] over 20,000 iterations/simulated transitions, for the four values $\lambda = 0, 0.3, 0.7, 1$. All runs used the same simulation trajectory.

[Boy02] Boyan, J. A., "Technical Update: Least-Squares Temporal Difference Learning," Machine Learning, Vol. 49, 2002, pp. 1-15.

[BrB96] Bradtke, S. J., and Barto, A. G., "Linear Least-Squares Algorithms for Temporal Difference Learning," Machine Learning, Vol. 22, 1996, pp. 33-57.

[Day92] Dayan, P. D., "The Convergence of TD(λ) for general λ," Machine Learning, Vol. 8, 1992, pp. 341-362.

[FaV00] de Farias, D. P., and Van Roy, B., " On the Existence of Fixed Points for Approximate Value Iteration and Temporal-Difference Learning ," J. of Optimization Theory and Applications, Vol. 105, 2000.

[GLH94] Gurvits, L., Lin, L. J., and Hanson, S. J., "Incremental Learning of Evaluation Functions for Absorbing Markov Chains: New Methods and Theorems," Preprint, 1994.

[Kon02] Konda, V. R., Actor-Critic Algorithms, Ph.D. Thesis, Dept. of Electrical Engineering and Computer Science, M.I.T., Cambridge, MA, 2002.

[NeB03] Nedić, A., and Bertsekas, D. P., "Least Squares Policy Evaluation Algorithms with Linear Function Approximation," Discrete Event Dynamic Systems: Theory and Applications, Vol. 13, 2003, pp. 79-110.

[Pin72] Pineda, F., "Mean-Field Analysis for Batched TD(λ)," Neural Computation, 1997, pp. 1403-1419.

[Put94] Puterman, M. L., Markov Decision Processes, John Wiley Inc., New York, 1994.

[SuB98] Sutton, R. S., and Barto, A. G., Reinforcement Learning, MIT Press, Cambridge, MA, 1998.

[Sut88] Sutton, R. S., "Learning to Predict by the Methods of Temporal Differences," Machine Learning, 3, 1988, pp. 9–44.

[TsV97] Tsitsiklis, J. N., and Van Roy, B., "An Analysis of Temporal-Difference Learning with Function Approximation," IEEE Transactions on Automatic Control, 42, 1997, pp. 674–690.