Temporal Difference Methods for General Projected Equations

Dimitri P. Bertsekas

***Abstract*—We consider projected equations for approximate solution of high-dimensional fixed point problems within low-dimensional subspaces. We introduce an analytical framework based on an equivalence with variational inequalities, and algorithms that may be implemented with low-dimensional simulation. These algorithms originated in approximate dynamic programming (DP), where they are collectively known as temporal difference (TD) methods. Even when specialized to DP, our methods include extensions/new versions of TD methods, which offer special implementation advantages and reduced overhead over the standard LSTD and LSPE methods, and can deal with near singularity in the associated matrix inversion. We develop deterministic iterative methods and their simulation-based versions, and we discuss a sharp qualitative distinction between them: the performance of the former is greatly affected by direction and feature scaling, yet the latter have the same asymptotic convergence rate regardless of scaling, because of their common simulation-induced performance bottleneck.**

***Index Terms*—Approximation methods, dynamic programming, Markov decision processes, reinforcement learning, temporal difference methods.**

I. Introduction

We consider the approximation of a fixed point of a mapping $T : \Re^n \mapsto \Re^n$ by solving the projected equation

$$x = \Pi T(x) \tag{1}$$

where Π denotes projection onto a closed convex subset $\hat{S}$ of $\Re^n$. The projection is with respect to a weighted Euclidean norm $\|\cdot\|_\Xi$, where Ξ is a positive definite symmetric matrix (i.e., $\|x\|_\Xi^2 = x'\Xi x$).[1] We assume that $\hat{S}$ is contained in a subspace S spanned by the columns of an $n \times s$ matrix Φ, which may be viewed as basis functions, suitably chosen to match the characteristics of the underlying problem:

$$S = \{\Phi r \mid r \in \Re^s\}. \tag{2}$$

Implicit here is the assumption that $s \ll n$, so we are interested in low-dimensional approximations of the high-dimensional fixed point. The convex set $\hat{S}$ may be represented as a convex subset $\hat{R} \subset \Re^s$, where

$$\hat{R} = \{r \mid \Phi r \in \hat{S}\}, \qquad \hat{S} = \Phi \hat{R} \tag{3}$$

so solving the projected equation (1) is equivalent to finding $r \in \hat{R}$ that satisfies

$$\Phi r = \Pi T(\Phi r). \tag{4}$$

Note that our choice of a fixed point format is not strictly necessary for our development, since any equation of the form $F(x) = 0$, where $F : \Re^n \mapsto \Re^n$, can be converted into the fixed point problem $x = x - F(x)$.

The approximation framework just described has a long history for the case where $\hat{S} = S$ and $\hat{R}$ is the entire space $\Re^s$. To set the stage for subsequent developments, we will describe its connection with two important contexts, *approximate DP* and *Galerkin approximation*. We will then describe a new connection with a more general context, related to *approximate solution of variational inequalities (VI)*, where $\hat{R}$ is a strict subset of $\Re^s$.

A. Approximate DP

Here T is a DP/Bellman operator, and x has the interpretation of the optimal cost vector or the cost vector of a policy. Furthermore in the literature thus far it has been assumed that $\hat{S} = S$, so r is unconstrained ($\hat{R} = \Re^s$) and the projected equation (4) is linear. An example is policy evaluation in a discounted finite-state problem where T is linear of the form $T(x) = Ax + b$, with $A = \alpha P$, where P is a given transition probability matrix corresponding to a fixed policy, b is a given cost vector of the policy, and $\alpha \in (0, 1)$ is a discount factor. Other cases where $\alpha = 1$ include the classical average cost and stochastic shortest path problems; see e.g., Bertsekas [1], Puterman [2]. An approximate/projected solution of Bellman's equation can be used to generate an (approximately) improved policy through an (approximate) policy iteration scheme. This approach is described in detail in the literature, has been extensively tested in practice, and is one of the major methods for approximate DP (see the books by Bertsekas and Tsitsiklis [3], Sutton and Barto [4], and Powell [5]; Bertsekas [1] provides a recent textbook treatment and up-to-date references).

For problems of very high dimension, classical matrix inversion methods cannot be used to solve the projected equation, and *temporal differences methods* are one of the principal alternatives; see [1], [3], [4]. These are simulation-based methods that can be divided in two categories: *iterative* and *matrix inversion* (also called equation approximation).

Manuscript received April 02, 2009; revised September 26, 2009, January 26, 2010, and June 12, 2010; accepted January 20, 2011. Date of publication February 17, 2011; date of current version September 08, 2011. This work was supported by NSF Grant ECCS-0801549. Recommended by Associate Editor C.-H. Chen.

The author is with the Laboratory for Information and Decision Systems (LIDS), Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: dimitrib@mit.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TAC.2011.2115290

[1] In our notation $\Re^s$ is the s-dimensional Euclidean space, all vectors in $\Re^s$ are viewed as column vectors, and a prime denotes transposition.

The iterative methods produce a sequence $\{r_k\}$ converging to a solution of the projected Bellman's equation $\Phi r = \Pi(A\Phi r + b)$, with Π defined by the diagonal matrix Ξ that has the steady-state distribution of P along the diagonal. They generate a sequence of indexes $\{i_0, i_1, \ldots\}$ using the Markov chain associated with P, and they use the temporal differences (TD) defined by

$$q_{k,t} = \phi(i_t)' r_k - \alpha \phi(i_{t+1})' r_k - b_{i_t}, \qquad t \le k \tag{5}$$

where $\phi(i)'$ denotes the ith row of the matrix Φ. The original method known as TD(0), due to Sutton [6], is

$$r_{k+1} = r_k - \gamma_k \phi(i_k) q_{k,k} \tag{6}$$

where γ_k is a stepsize sequence that diminishes to 0.[2] It may be viewed as a stochastic approximation/Robbins-Monro scheme for solving the equation $\Phi'\Xi(\Phi r - A\Phi r - b) = 0$ (the necessary and sufficient condition for r to solve the projected equation). Indeed, using (5), (6), it is seen that $\phi(i_k)q_{k,k}$ is a sample of the left-hand side $\Phi'\Xi(\Phi r - A\Phi r - b)$ of the equation.

Because TD(0) is often slow and unreliable (this is well-known in practice and typical of stochastic approximation schemes; see also the analysis by Konda [10]), alternative iterative methods have been proposed. One of them is the Fixed Point Kalman Filter (FPKF), proposed by Choi and Van Roy [11] and given by

$$r_{k+1} = r_k - \gamma_k D_k^{-1} \phi(i_k) q_{k,k} \tag{7}$$

where D_k is a positive definite symmetric scaling matrix, selected to speed up convergence. It is a scaled (by the matrix D_k) version of TD(0), so it may be viewed as a stochastic approximation-type method. The choice

$$D_k = \frac{1}{k+1} \sum_{t=0}^{k} \phi(i_t)\phi(i_t)' \tag{8}$$

is suggested in [11] and some favorable computational results are reported, albeit without theoretical proof of convergence rate superiority over TD(0).

An alternative to TD(0) is the Least Squares Policy Evaluation algorithm (LSPE, proposed by Bertsekas and Ioffe [12]; see also Nedić and Bertsekas, [13], Bertsekas, Borkar, and Nedić [14], Yu and Bertsekas [15], [33]):

$$r_{k+1} = r_k - \frac{1}{k+1} D_k^{-1} \sum_{t=0}^{k} \phi(i_t) q_{k,t} \tag{9}$$

where D_k is given by (8). While this method resembles the FPKF iteration (7), it is different in a fundamental way because it is not a stochastic approximation method. Instead it may be viewed as the fixed point/projected value iteration $x_{k+1} = \Pi T(x_k)$, where the mapping ΠT is approximated by simulation (see the discussion in Sections III and IV). Compared with TD(0) and FPKF, it does not require the stepsize γ_k, and uses the time average $(k+1)^{-1} \sum_{t=0}^{k} \phi(i_t) q_{k,t}$ of the TD term in its right-hand side in place of $\phi(i_k) q_{k,k}$, the latest sample of the TD term [cf. (6) and (7)]. This results in reduced simulation noise within the iteration, and much improved theoretical rate of convergence and practical reliability, as verified by computational studies and convergence rate analysis (see [10], [12], and [15], [33]).

The validity of all these iterative algorithms depends on ΠT being a contraction mapping with respect to the norm $\|\cdot\|_\Xi$, where Ξ is the diagonal matrix whose diagonal components are the steady-state probabilities of the Markov chain. When these algorithms are extended to solve nonlinear versions of Bellman's equation, they become unreliable because in the nonlinear context, ΠT need not be a contraction [3], [16] (a notable exception is optimal stopping problems, as shown by Tsitsiklis and Van Roy [17], [18]; see also Yu and Bertsekas [19]).

The alternatives to iterative methods are matrix inversion methods, a prime example of which is the Least Squares Temporal Differences method (LSTD, proposed by Bradtke and Barto [20], and followed up by Boyan [21], and Nedić and Bertsekas [13]). It writes the projected equation (4) in an equivalent linear form $Cr = d$, where C is an $s \times s$ matrix, and $d \in \Re^s$, then uses the type of simulation described earlier to compute a matrix $\hat{C} \approx C$ and a vector $\hat{d} \approx d$, and approximates the solution $C^{-1}d$ with $\hat{C}^{-1}\hat{d}$ (cf. Section IV-A). This method can also be implemented using temporal differences: the vector $\hat{C}^{-1}\hat{d}$ is the vector r_k that solves the equation $\sum_{t=0}^{k} \phi(i_t) q_{k,t} = 0$, where k is the number of samples obtained from the simulation [cf. (5), (9) and Section IV-C].

B. Galerkin Approximation

This is an older methodology, which is widely used for approximating the solution of linear operator equations, including integral and partial differential equations, and their finely discretized versions. Here we are given a fixed point problem $x = Ax + b$, where A is an $n \times n$ matrix and $b \in \Re^n$ is a vector, a subspace $S \subset \Re^n$ of the form (2), and a (possibly weighted) Euclidean projection operator Π from $\Re^n$ to S. Then we approximate a fixed point with a vector $\Phi r \in S$ that solves the projected equation $\Phi r = \Pi(A\Phi r + b)$ (see e.g., [22], [23]). Thus, the projected equation framework of approximate DP is a special case of Galerkin approximation. This connection, which is potentially significant, does not seem to have been mentioned in the literature.

Another related approach uses two subspaces, S and U, and a least squares formulation. The vector that minimizes $\|x - Ax - b\|^2$ is approximated by an $x \in S$ such that the residual $(x - Ax - b)$ is orthogonal to U (this is known as the Petrov-Galerkin condition [24]). If $U = \Xi S$, where Ξ is a positive definite symmetric matrix, then the orthogonality condition is written as $y'\Xi(x - Ax - b) = 0$ for all $y \in S$, which together with

[2] There are "λ-versions" of TD(0) and other TD methods, which use a parameter $\lambda \in (0,1)$ and aim to solve the "weighted-multistep" version of Bellman's equation, where T is replaced by

$$T^{(\lambda)} = (1-\lambda) \sum_{\ell=0}^{\infty} \lambda^\ell T^{\ell+1}.$$

The best known example is $\mathrm{TD}(\lambda)$ [6]. Our algorithms and qualitative conclusions apply to general $\lambda \in [0,1)$. For our purposes in this paper, however, we focus primarily on $\lambda = 0$, and briefly summarize the case $\lambda > 0$ in Section IV-E. For the unconstrained case ($\hat{S} = S$) and $\lambda > 0$, analogs of $\mathrm{TD}(\lambda)$, $\mathrm{LSPE}(\lambda)$, and $\mathrm{LSTD}(\lambda)$ for general projected equations and their convergence properties are discussed in [8] and [9].

the condition $x \in S$, is equivalent to the projected equation $\Phi r = \Pi(A\Phi r + b)$. Alternatively, if $U = (I - A)S$, then the orthogonality condition is written as $y'(I - A)'(x - Ax - b) = 0$ for all $y \in S$, which together with $x \in S$, is the optimality condition for minimization of $\|x - Ax - b\|^2$ over $x \in S$. The optimality condition is in turn equivalent to the projected equation $\Pi(I - A)'(x - Ax - b) = 0$, where Π denotes projection on S with respect to the standard (unweighted) Euclidean norm. This approach to deriving a projected equation can be applied to general linear least squares problems, where A is not necessarily a square matrix. It has also been applied in approximate DP under the name *Bellman error method*, for approximating the solution of the linear Bellman's equation discussed in Section I-A.

Note that the Galerkin methodology, as currently practiced in scientific computation, does not use the Monte Carlo simulation ideas that are central in approximate DP. Instead, the projected equation is solved by standard matrix inversion or iterative methods. Thus, the methodology can be applied only to problems of small dimension or to problems where the basis matrix Φ is favorably chosen, so that the linear algebra calculations to obtain and to solve the exact form of the projected equation are feasible. This motivates our extension of simulation-based approximate DP methods to more general non-DP contexts where n is extremely large and Φ cannot be chosen favorably.

C. Approximate Solution of Variational Inequalities

This context is more general than the preceding two because $\hat{R}$ may be a strict subset of $\Re^s$. In fact it is equivalent to the projected equation (1) as we will explain shortly. This equivalence has not been noticed earlier, to our knowledge, and is the starting point for the developments of this paper.

By the properties of projection, x^* satisfies $x^* = \Pi T(x^*)$ if and only if $x^* \in \hat{S}$ and the vector $x^* - T(x^*)$ forms a nonnegative inner product with all vectors $x - x^*$ with $x \in \hat{S}$, i.e.,

$$(x^* - T(x^*))' \, \Xi(x - x^*) \geq 0, \qquad \forall x \in \hat{S}. \tag{10}$$

Here Ξ is the positive definite symmetric matrix that defines the projection norm $\|\cdot\|_\Xi$ and the associated inner product $x_1'\Xi x_2$ of any two vectors x_1, x_2; see Fig. 1. We can equivalently write (10) as the VI $f(x^*)'(x - x^*) \geq 0$ for all $x \in \hat{S}$ or as the VI[3]

$$f(\Phi r^*)'\Phi(r - r^*) \geq 0, \qquad \forall r \in \hat{R} \tag{11}$$

where $f : \Re^n \mapsto \Re^n$ is the function defined by

$$f(x) = \Xi\,(x - T(x)) \tag{12}$$

and $\hat{R} = \{r | \Phi r \in \hat{S}\}$ [cf. (3)]. In conclusion, *projected equations of the form $x = \Pi T(x)$ and VIs of the form* (11), (12) *are equivalent*, so analytical and algorithmic methods for solving one of the two problems may be used to solve the other.

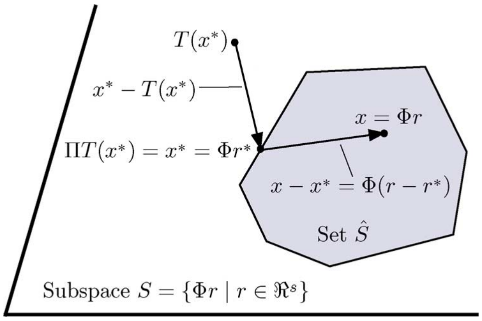
$T(x^*)$
$x^* - T(x^*)$
$x = \Phi r$
$\Pi T(x^*) = x^* = \Phi r^*$
$x - x^* = \Phi(r - r^*)$
Set $\hat{S}$
Subspace $S = \{\Phi r \mid r \in \Re^s\}$

Fig. 1. Equivalence of a projected equation $x^* = \Pi T(x^*)$ with the variational inequality $f(\Phi r^*)'\Phi(r - r^*) \geq 0, \forall r \in \hat{R}$, where $f(x) = \Xi(x - T(x))$ and $\hat{R} = \{r | \Phi r \in \hat{S}\}$. By the properties of projection, we have $x^* = \Pi T(x^*)$ if and only if $x^* \in \hat{S}$ and the inner product $(x^* - T(x^*))'\Xi(x - x^*)$ is nonnegative for all $x \in \hat{S}$.

There are several interesting problems from optimization and game theory that can be modeled by VIs (see e.g., [25], [26]), and the connection with projected equations can be used as a basis for an approximate solution approach. We discuss these connections in the report [7], which is in effect an extended version of the present paper.

D. New TD Algorithms

The starting point of this paper is a classical (deterministic) iterative projection algorithm for monotone VIs of the form (11). This algorithm has the form

$$r_{k+1} = P_{D,\hat{R}}\left[r_k - \gamma D^{-1}\Phi' f(\Phi r_k)\right] \tag{13}$$

where γ is a positive constant stepsize, D is a positive definite symmetric matrix, and $P_{D,\hat{R}}[\cdot]$ denotes projection on $\hat{R}$ with respect to the norm $\|r\|_D = \sqrt{r'Dr}$. One of the focal points of this paper is to propose and analyze a new class of TD methods that are simulation-based versions of this iteration, transcribed to the projected equation framework. When specialized to approximate DP (with simulation done in the manner described in Section I-A), our methods take the form

$$r_{k+1} = P_{D_k,\hat{R}}\left[r_k - \frac{\gamma}{k+1} D_k^{-1} \sum_{t=0}^{k} \phi(i_t) q_{k,t}\right] \tag{14}$$

where D_k is a sequence of positive definite symmetric matrices and $q_{k,t}$ is the TD of (5). This is similar to LSPE [cf. (8), (9)] but is more general in two ways:

1) The constraint set $\hat{R}$ may be a strict subset of $\Re^s$. This is useful in cases where some prior information on the fixed point of T can be translated into useful constraints on r. Also, in certain contexts one may wish to replace $\hat{R}$ by an approximation to facilitate the projection operation $P_{D_k,\hat{R}}[\cdot]$; see the discussion on constrained optimization applications in [7].
2) A general scaling matrix D_k may be used rather than the special choice (8). For example, D_k may be the identity or a diagonal approximation of the matrix (8), thereby avoiding

[3] The standard VI problem is to find a vector $r^* \in \hat{R}$ such that

$$F(r^*)'(r - r^*) \geq 0, \qquad \forall r \in \hat{R}$$

where $\hat{R}$ is a closed convex set and $F : \Re^s \mapsto \Re^s$ is a given function. The VI (11) corresponds to $F(r) = \Phi' f(\Phi r)$. The textbook by Facchinei and Pang [25] provides an extensive account of the associated theory.

the associated matrix inversion, and substantially reducing the associated overhead. Yet we will see that there is no rate of convergence penalty for doing so, with a potential net gain in algorithmic efficiency resulting.

Aside from these generalizations within the approximate DP context, our methods apply to general (nonDP-related) projected equations, and generalize similarly a corresponding LSPE-type algorithm given in [8].

E. Summary of the Paper

The paper is structured as follows. In Section II, we establish the conditions that we need for iteration (13) to be applicable to projected equations. In particular, the associated VI must have certain monotonicity properties, which are in turn related to contraction properties of the projected equation. In Section III we focus on the case where T is linear, and we apply the iteration (13) to projected equations. We interpret the role of the scaling matrix D in the context of subspace approximation and we show that it is related to *feature scaling*, i.e., alternative representations of the subspace S using different sets of basis functions.

The main algorithmic contributions of the paper are contained in Section IV. We develop new simulation-based algorithms for general projected equations, which require low (s-dimensional) calculations only, and we investigate their properties. In the process we recover the existing TD methods for approximate DP, including LSPE and LSTD. We introduce iterative regularization algorithms that work well when the projected equation is nearly singular, and/or does not involve a contraction. These algorithms provide a connecting link between iterative and matrix inversion methods. We also consider rate of convergence issues, and we derive an important qualitative result: in simulation-based implementations, the slower speed of simulation dominates, and *all simulation-based algorithms in our framework converge at the same rate asymptotically, regardless of the scaling used* (although the short-term convergence rate may be significantly affected by scaling).

As a byproduct of our analysis, we clarify the significance of rank conditions on the matrix Φ. The assumption that Φ has full rank has been universally made in previous convergence analyses of TD(0) and related methods. We show that Φ *need not have full rank for convergence of iterative TD-type methods* (unless this is required for invertibility of D_k). As a special case, we show that when Φ is rank-deficient and hence the projected equation admits multiple solutions, TD(0) converges to the projection of the initial iterate on the manifold of solutions.

II. Iterative Methods for Variational Inequalities

Given a mapping $F : \Re^s \mapsto \Re^s$, a closed convex set $\hat{R}$, and the VI

$$F(r^*)'(r - r^*) \geq 0, \qquad \forall r \in \hat{R} \tag{15}$$

let us consider the iteration (13):

$$r_{k+1} = P_{D,\hat{R}}\left[r_k - \gamma D^{-1} F(r_k)\right]$$

which can also be written as a quadratic program:

$$r_{k+1} = \arg\min_{r \in \hat{R}} \left\{ F(r_k)'(r - r_k) + \frac{1}{2\gamma}(r - r_k)' D (r - r_k) \right\}. \tag{16}$$

This iteration has a long history, and contains as a special case the class of (scaled by D) gradient projection methods for minimizing a cost function whose gradient is F over a constraint set $\hat{R}$ (see sources in nonlinear programming or [26], Ch. 3).

The properties of this method are closely linked with monotonicity properties of F (see e.g., Facchinei and Pang [25] for a detailed account). We say that F is monotone (strongly monotone) over $\hat{R}$ if for some $\beta \geq 0$ ($\beta > 0$, respectively) we have

$$\left(F(r_1) - F(r_2)\right)'(r_1 - r_2) \geq \beta \|r_1 - r_2\|^2, \quad \forall r_1, r_2 \in \hat{R}$$

(here $\|\cdot\|$ can be any norm, e.g., the standard Euclidean norm). If F is strongly monotone, the VI (15) has a unique solution r^*. If F is the gradient of a differentiable function H, then (strong) monotonicity of F over $\Re^s$ is equivalent to (strong) convexity of H over $\Re^s$.

If F is linear of the form $F(r) = Cr - d$, then F is monotone (strongly monotone) over $\Re^s$ if and only if C is a positive semidefinite (positive definite, respectively) matrix in the sense that $r'Cr \geq 0$ for all $r \in \Re^s$ ($r'Cr > 0$ for all $r \neq 0$, respectively); see [25]. When $\hat{R} = \Re^s$, the VI (15) is equivalent to the linear system $Cr = d$.

The standard convergence result for the projection method (13) (see e.g., [26], Section 3.5.3, or [25], Section 12.1.1) is that if F is Lipschitz continuous and strongly monotone over $\hat{R}$, with unique solution denoted by r^*, there exists $\bar{\gamma} > 0$ such that $r_k \to r^*$ linearly for each constant stepsize γ in the range $(0, \bar{\gamma}]$ (i.e., $\|r_k - r^*\|$ converges to 0 at least as fast as a geometric progression). The strong monotonicity assumption is essential for this—just monotonicity (i.e., $\beta = 0$) may result in divergence (see e.g., [26], p. 270).

Let now F have the special form [cf. (11)]

$$F(r) = \Phi' f(\Phi r)$$

where Φ is an $n \times s$ matrix, and $f : \Re^n \mapsto \Re^n$ is Lipschitz continuous and strongly monotone over the set $\hat{S} = \Phi \hat{R}$. Then F is Lipschitz continuous, but it may not be strongly monotone, so the solution of the corresponding VI may not be unique, and the convergence of the corresponding iteration [cf. (13)]

$$r_{k+1} = P_{D,\hat{R}}\left[r_k - \gamma D^{-1} \Phi' f(\Phi r_k)\right]$$

comes into doubt. However, despite the lack of strong monotonicity of F, it turns out that this iteration is convergent in a way similar to the case where F is strongly monotone. In particular, in a paper devoted to the case $F(r) = \Phi' f(\Phi r)$ [27], it was shown that there exists $\bar{\gamma} > 0$ such that $r_k \to r^*$ linearly for each $\gamma \in (0, \bar{\gamma}]$, where r^* is *some* solution of

$$f(\Phi r^*)' \Phi (r - r^*) \geq 0, \qquad \forall r \in \hat{R}$$

provided f is strongly monotone over $\Phi \hat{R}$ and $\hat{R}$ is a polyhedral set (the polyhedral assumption is essential).

We next show that contraction properties of T or ΠT imply that f is strongly monotone over $\hat{S}$, which is a prerequisite for the convergence of the method (13). The properties of the next two propositions can be easily inferred from existing results on variational inequalities, but for completeness we provide the proofs.

Proposition 1: Assume that T is a contraction with respect to the norm $\|\cdot\|_\Xi$ over the set $\hat{S}$. Then the function f of (12) is strongly monotone over $\hat{S}$.

Proof: Let $\alpha \in [0,1)$ be the modulus of contraction of T. For any two vectors $x_1, x_2 \in \hat{S}$

$$\begin{aligned}&(f(x_1)-f(x_2))'(x_1-x_2)\\ &\quad=(x_1-T(x_1)-x_2+T(x_2))'\,\Xi(x_1-x_2)\\ &\quad=(x_1-x_2)'\Xi(x_1-x_2)-(T(x_1)-T(x_2))'\,\Xi(x_1-x_2)\\ &\quad\geq \|x_1-x_2\|_\Xi^2-\|T(x_1)-T(x_2)\|_\Xi\,\|x_1-x_2\|_\Xi\\ &\quad\geq \|x_1-x_2\|_\Xi^2-\alpha\|x_1-x_2\|_\Xi^2\\ &\quad=(1-\alpha)\|x_1-x_2\|_\Xi^2,\end{aligned}$$

where the first inequality follows from the Cauchy-Schwarz inequality, and the second inequality follows from the contraction property of T. Since $\alpha \in [0,1)$, this shows that f is strongly monotone on $\hat{S}$. ■

In the special case where $\hat{S}=S$ (i.e., r is unconstrained) and Π is projection on the subspace S, it is sufficient that ΠT rather than T be a contraction. The origin of the following proposition can be traced to the convergence proof of TD(λ) in [17] (Lemma 9); see also [8], Prop. 5.

Proposition 2: Assume that $\hat{S}=S$ and that ΠT is a contraction with respect to the norm $\|\cdot\|_\Xi$ over the subspace S. Then the function f of (12) is strongly monotone over S.

Proof: Let $\alpha \in [0,1)$ be the modulus of contraction of ΠT, and note that we have

$$(T(x)-\Pi T(x))'\,\Xi\bar{x}=0, \qquad \forall x,\bar{x}\in S \tag{17}$$

since vectors of the form $x-\Pi x$ are orthogonal (with respect to the norm $\|\cdot\|_\Xi$) to S. We use this equation as an intermediate step in the proof of the preceding proposition to obtain the desired conclusion.

We have for any two vectors $x_1, x_2 \in S$

$$\begin{aligned}&(f(x_1)-f(x_2))'(x_1-x_2)\\ &\quad=(x_1-T(x_1)-x_2+T(x_2))'\,\Xi(x_1-x_2)\\ &\quad=(x_1-x_2)'\Xi(x_1-x_2)-(T(x_1)-T(x_2))'\,\Xi(x_1-x_2)\\ &\quad=\|x_1-x_2\|_\Xi^2-(\Pi T(x_1)-\Pi T(x_2))'\,\Xi(x_1-x_2)\\ &\quad\geq \|x_1-x_2\|_\Xi^2-\|\Pi T(x_1)-\Pi T(x_2)\|_\Xi\,\|x_1-x_2\|_\Xi\\ &\quad\geq \|x_1-x_2\|_\Xi^2-\alpha\|x_1-x_2\|_\Xi^2\\ &\quad=(1-\alpha)\|x_1-x_2\|_\Xi^2,\end{aligned}$$

where the third equation follows from (17), the first inequality follows from the Cauchy-Schwarz inequality, and the second inequality follows from the contraction property of ΠT. This shows that f is strongly monotone on S. ■

There are well-known cases in approximate DP where ΠT is a contraction with respect to $\|\cdot\|_\Xi$, with Ξ a diagonal matrix (see [1], [3], [15], [17], [28], [33]). An example is discounted or average cost DP, where $T(x)=\alpha Px+b$, with $\alpha\in(0,1]$, P is a transition probability matrix of an ergodic Markov chain, and Ξ is a diagonal matrix with the steady-state probabilities of the chain along the diagonal. Reference [8] provides several general criteria for verifying that ΠT is a contraction, beyond the DP context.

III. Deterministic Iterative Methods for Projected Equations and Linear Mappings

For the remainder of the paper, we assume that T is linear of the form

$$T(x)=Ax+b$$

where A is an $n\times n$ matrix and b is a vector in $\Re^n$. To be able to use the convergence result given in Section II, *we assume that $\hat{R}$ is a polyhedral set, and that the mapping $f(x)=\Xi(x-T(x))$* [cf. (11), (12)] *is strongly monotone over $\hat{S}$* (this is guaranteed under contraction assumptions on T or ΠT, as per Props. 1 and 2). As a result, the VI

$$f(x^*)'(x-x^*)\geq 0, \qquad \forall x\in\hat{S}$$

has a unique solution $x^*\in\hat{S}$.

In the low-dimensional space $\Re^s$, this VI is written as

$$f(\Phi r^*)'\Phi(r-r^*)\geq 0, \qquad \forall r\in\hat{R}$$

and is equivalent to the projected equation $\Phi r=\Pi T(\Phi r)$ (cf. Section I-C). We have $\Phi' f(\Phi r)=\Phi'\Xi(\Phi r-A\Phi r-b)$, or

$$\Phi' f(\Phi r)=Cr-d \tag{18}$$

where

$$C=\Phi'\Xi(I-A)\Phi, \qquad d=\Phi'\Xi b \tag{19}$$

so the VI is equivalent to

$$(Cr^*-d)'(r-r^*)\geq 0, \qquad \forall r\in\hat{R}. \tag{20}$$

Its solution set is $R^*=\{r\in\hat{R}\,|\,\Phi r=x^*\}$, and if Φ has full rank, R^* consists of a single point. The iteration (13) takes the form

$$r_{k+1}=P_{D,\hat{R}}\left[r_k-\gamma D^{-1}(Cr_k-d)\right] \tag{21}$$

and is convergent to some $r^*\in R^*$, under the conditions discussed in Section II.

A. The Unconstrained Case

When r is unconstrained ($\hat{R}=\Re^s$), the algorithm (21) takes the form

$$r_{k+1}=r_k-\gamma D^{-1}(Cr_k-d) \tag{22}$$

and the geometry of the convergence process is illustrated in Fig. 2. The set of solutions R^* is parallel to $\mathrm{N}(\Phi)$, the nullspace of Φ, while since d belongs to $\mathrm{Ra}(C)$, the range space of C, the

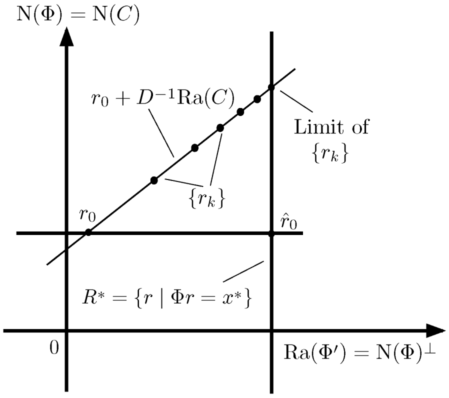
$\mathrm{N}(\Phi) = \mathrm{N}(C)$
$r_0 + D^{-1}\mathrm{Ra}(C)$
Limit of $\{r_k\}$
$\{r_k\}$
r_0
$\hat{r}_0$
$R^* = \{r \mid \Phi r = x^*\}$
0
$\mathrm{Ra}(\Phi') = \mathrm{N}(\Phi)^\perp$

Fig. 2. Illustration of the convergence process of the iteration (22) in the case where Φ does not have full rank. The iteration converges to the intersection of the solution set R^* with the linear manifold $r_0 + D^{-1}\mathrm{Ra}(C)$. If $D = I$, the iteration converges to $\hat{r}_0$, the orthogonal projection of r_0 on R^*.

sequence $\{r_k\}$ generated by iteration (22) lies in the linear manifold $r_0 + D^{-1}\mathrm{Ra}(C)$. This manifold has a unique intersection point with R^*, so $\{r_k\}$ converges to that point.[4] In the special case where $D = I$, $\{r_k\}$ converges to $\hat{r}_0$, the orthogonal projection of r_0 onto R^*, since $r_k - r_0$ belongs to $\mathrm{Ra}(C) \subset \mathrm{Ra}(\Phi')$ [cf. (19)], so it is orthogonal to $\mathrm{N}(\Phi)$ and hence to R^*.

Iteration (22) converges if and only if $I - \gamma D^{-1}C$ is a contraction, so the choice of γ is critical for convergence. However, there is an important special case, where a proper choice of γ is known, namely

$$D = \Phi'\Xi\Phi, \qquad \gamma = 1.$$

Then it can be shown (see [1], [8]) that when iteration (22) is multiplied by Φ, it becomes the *projected Jacobi method*

$$x_{k+1} = \Pi T(x_k)$$

which converges when ΠT is a contraction.

Another special case of iteration (22) is when D is the identity:

$$r_{k+1} = r_k - \gamma(Cr_k - d). \tag{23}$$

An intermediate possibility between the preceding two cases is a matrix D, which is a diagonal approximation to $\Phi'\Xi\Phi$, thereby simplifying the matrix inversion in (22). Then one may expect that a stepsize γ close to 1 will often lead to $I - \gamma D^{-1}C$ being a contraction, thereby facilitating the choice of γ.

The three special cases just discussed admit interesting simulation-based approximate implementations, as we will discuss in Section IV.

B. *Effects of Feature Scaling*

The iteration

$$r_{k+1} = P_{D,\hat{R}}\left[r_k - \gamma D^{-1}(Cr_k - d)\right] \tag{24}$$

[cf. (21)] involves two different types of scaling: one is *direction scaling* embodied in the choice of the matrix D, and the other is *feature scaling* embodied in the choice of the matrix Φ, which defines C and d via (19). We will now show that these two types of scaling are related, and that the algorithmic effect induced by a change in feature scaling can also be induced by a change in direction scaling, and reversely.

To this end, we represent the subspace S with two different matrices Φ and Ψ, related by

$$\Phi = \Psi B$$

where B is an $\bar{s} \times s$ matrix such that the range spaces of Φ and Ψ coincide (and are equal to S).[5] We compare the corresponding high-dimensional sequences

$$x_{k,\Phi} = \Phi r_k, \qquad x_{k,\Psi} = \Psi v_k$$

where r_k and v_k are generated by corresponding iterations of the form (24), written in the quadratic programming form (16):

$$r_{k+1} = \arg\min_{\Phi r \in \hat{S}} \Big\{ f(\Phi r_k)'\Phi(r - r_k) + \frac{1}{2\gamma}(r - r_k)'D_\Phi(r - r_k) \Big\}$$

or

$$x_{k+1,\Phi} = \arg\min_{x \in \hat{S}} \Big\{ f(x_{k,\Phi})'(x - x_{k,\Phi}) + \frac{1}{2\gamma}\min_{\Phi r = x}(r - r_k)'D_\Phi(r - r_k) \Big\} \tag{25}$$

and

$$v_{k+1} = \arg\min_{\Psi v \in \hat{S}} \Big\{ f(\Psi v_k)'\Psi(v - v_k) + \frac{1}{2\gamma}(v - v_k)'D_\Psi(v - v_k) \Big\}$$

or

$$x_{k+1,\Psi} = \arg\min_{x \in \hat{S}} \Big\{ f(x_{k,\Psi})'(x - x_{k,\Psi}) + \frac{1}{2\gamma}\min_{\Phi r = x}(v - v_k)'D_\Psi(v - v_k) \Big\}. \tag{26}$$

A straightforward quadratic programming duality argument shows that

$$\frac{1}{2}\min_{\Phi r = x}(r - r_k)'D_\Phi(r - r_k) = \max_{\mu \in \Re^n}\left\{ -\frac{1}{2}\mu'\Phi D_\Phi^{-1}\Phi'\mu + \mu'(\Phi r_k - x), \right\}$$

[4] To see this, note that $\mathrm{Ra}(C)$ is contained in $\mathrm{Ra}(\Phi')$ [cf. (19)]. Thus, the subspaces $D^{-1}\mathrm{Ra}(C)$ and $\mathrm{N}(\Phi)$ intersect at just the origin [if $r \in D^{-1}\mathrm{Ra}(C) \cap \mathrm{N}(\Phi)$, we have $r = D^{-1}\Phi' v$ for some v and also $r'\Phi' v = 0$, so that $r'Dr = 0$ and $r = 0$]. Since R^* is parallel to $\mathrm{N}(\Phi)$, it intersects $D^{-1}\mathrm{Ra}(C)$ at a unique point.

[5] Given matrices Φ and Ψ with equal range spaces, it is always possible to write $\Phi = \Psi B$ for a suitable matrix B (form a basis for the common range space by using a maximal linearly independent set of columns of Ψ, and express the columns of Φ in terms of that basis). Given matrices Φ and Ψ such that $\Phi = \Psi B$, it can be shown that the range spaces of Φ and Ψ are equal if and only if the range space of B contains the range space of Ψ'. In particular, if the rank of B is $\bar{s}$, the range spaces of Φ and Ψ are equal.

so from (25), we have

$$x_{k+1,\Phi} = \arg\min_{x\in\hat S}\left\{ f(x_{k,\Phi})'(x-x_{k,\Phi}) + \frac{1}{\gamma}\max_{\mu\in\Re^n}\left\{-\frac{1}{2}\mu'\Phi D_\Phi^{-1}\Phi'\mu+\mu'(x_{k,\Phi}-x)\right\}\right\}.$$

Similarly, from (26),

$$x_{k+1,\Psi} = \arg\min_{x\in\hat S}\left\{ f(x_{k,\Psi})'(x-x_{k,\Psi}) + \frac{1}{\gamma}\max_{\mu\in\Re^n}\left\{-\frac{1}{2}\mu'\Psi D_\Psi^{-1}\Psi'\mu+\mu'(x_{k,\Psi}-x)\right\}\right\}.$$

A comparison of the preceding two equations, shows that if the scaling matrices satisfy $\Phi D_\Phi^{-1}\Phi' = \Psi D_\Psi^{-1}\Psi'$, or equivalently using the equation $\Phi = \Psi B$

$$D_\Psi^{-1} = BD_\Phi^{-1}B' \tag{27}$$

the two scaled iterations (25) and (26) produce identical results within the high-dimensional space ($x_{k,\Phi} = x_{k,\Psi}$ for all k, assuming that $x_{0,\Phi} = x_{0,\Psi}$). In conclusion, *alternative choices of feature scaling correspond to alternative choices of direction scaling*.

Another observation is that given a matrix Φ that has full rank, the entire class of iterations (24) can be derived from the simple special case where $D = I$

$$r_{k+1}=\arg\min_{\Phi r\in\hat S}\left\{f(\Phi r_k)'\Phi(r-r_k)+\frac{1}{2\gamma}(r-r_k)'(r-r_k)\right\} \tag{28}$$

by using scaling matrices of the form

$$D^{-1} = BB'$$

corresponding to square invertible feature scaling matrices B [cf. (27)].

IV. Simulation-Based Methods

In this section, we consider simulation-based versions of the deterministic methods of the preceding sections. We focus on the VI

$$(Cr^*-d)'(r-r^*)\geq 0, \qquad \forall r\in\hat R \tag{29}$$

[cf. (20)], and the associated iteration

$$r_{k+1} = P_{D,\hat R}\left[r_k-\gamma D^{-1}(Cr_k-d)\right] \tag{30}$$

[cf. (21)]. We will assume for the remainder of this section that Ξ is a diagonal matrix and that the vector of its (positive) diagonal elements

$$\xi = (\xi_1,\ldots,\xi_n)$$

is a probability distribution over the set of indexes $\{1,\ldots,n\}$.

We consider a simulation process introduced in [8]. We generate a sequence of indexes $\{i_0,i_1,\ldots\}$ (*row sampling*), and a sequence of transitions between indexes $\{(i_0,j_0),(i_1,j_1),\ldots\}$ (*column sampling*). Any probabilistic mechanism may be used for this, subject to the following two requirements:

- *Row Sampling Condition*: The sequence $\{i_0,i_1,\ldots\}$ is generated according to the distribution ξ, which defines the projection norm $\|\cdot\|_\Xi$, in the sense that with probability 1
$$\lim_{k\to\infty}\frac{\sum_{t=0}^k\delta(i_t=i)}{k+1}=\xi_i, \qquad i=1,\ldots,n$$
where $\delta(\cdot)$ denotes the indicator function [$\delta(E)=1$ if the event E has occurred and $\delta(E)=0$ otherwise].
- *Column Sampling Condition*: The sequence $\{(i_0,j_0),(i_1,j_1),\ldots\}$ is generated according to a certain stochastic matrix P with transition probabilities p_{ij} which satisfy
$$p_{ij}>0 \qquad \text{if} \qquad a_{ij}\neq 0$$
in the sense that with probability 1
$$\lim_{k\to\infty}\frac{\sum_{t=0}^k\delta(i_t=i,j_t=j)}{\sum_{t=0}^k\delta(i_t=i)}=p_{ij}$$
$i,j=1,\ldots,n$.

Then C_k and d_k are computed as

$$C_k=\frac{1}{k+1}\sum_{t=0}^k\phi(i_t)\left(\phi(i_t)-\frac{a_{i_tj_t}}{p_{i_tj_t}}\phi(j_t)\right)' \tag{31}$$

and

$$d_k=\frac{1}{k+1}\sum_{t=0}^k\phi(i_t)b_{i_t} \tag{32}$$

where we denote by $\phi(i)'$ the ith row of Φ. It can be shown using simple law of large numbers arguments that $C_k\to C$ and $d_k\to d$ with probability 1 (see [8]). In the case where $A=\alpha P$ with $\alpha\in(0,1)$, we have $C_k=(1/(k+1))\sum_{t=0}^k\phi(i_t)(\phi(i_t)-\alpha\phi(j_t))'$, which is a familiar formula in TD methods applied to α-discounted finite-state DP problems (cf. [1], [3], [4]).

A. Simulation-Based VI Approximation Approach

We now discuss a simulation-based noniterative approach to solve the VI (29), which generalizes the LSTD method of approximate DP. We generate the matrix C_k and vector d_k using (31), (32), and approximate the high-dimensional solution Φr^* by Φr_k^*, where r_k^* satisfies

$$(C_kr_k^*-d_k)'(r-r_k^*)\geq 0, \qquad \forall r\in\hat R. \tag{33}$$

Generally, the existence of a solution of the above VI may need to be verified separately. If Φ has full rank and the VI (29) is strongly monotone, then since $C_k\to C$ and $d_k\to d$ with probability 1, it follows that for sufficiently large k, the VI (33) is also strongly monotone, and therefore has a unique solution. In the unconstrained case ($\hat S = S$ and $\hat R=\Re^s$), the unique solution is $r_k^* = C_k^{-1} d_k$. In the context of approximate DP, the preceding equation is the well-known LSTD algorithm due to [20] (also described in textbooks such as [1]).

It is important to note, however, that the VI (29) is equivalent to the projected equation $\Phi r = \Pi T(\Phi r)$ regardless of its monotonicity properties, so *the VI* (33) *approximates the projected equation, regardless of whether it is monotone*. In particular, when $\hat{S} = S$ and C is invertible, C_k is also invertible for sufficiently large k, and $C_k^{-1} d_k$ converges to the unique solution $C^{-1}d$ of the projected equation. Thus, while iterative methods may require monotonicity and contraction assumptions for their validity, the noniterative/matrix inversion approach that uses simulation-based approximation of the projected equation is less restricted, although it still requires invertibility of C (in Section IV-C we will develop an iterative method for the case $\hat{S} = S$ that also does not rely on monotonicity and contraction assumptions, and does not require invertibility of C).

We have the following proposition, where we assume that the corresponding VIs of the form (33) are monotone for all k.

Proposition 3: The high-dimensional sequence obtained from the simulation process of (31), (32) is scale-free in the following sense: if $\{C_{k,\Phi}, d_{k,\Phi}\}$ is the sequence generated by these equations and $\{C_{k,\Psi}, d_{k,\Psi}\}$ is the corresponding sequence generated when Φ is replaced by Ψ, where $\Phi = \Psi B$ and B is an $s \times s$ invertible matrix, then the set of solutions of the corresponding VIs,

$$R_k^* = \Big\{ r_k^* | \Phi r_k^* \in \hat{S}, (C_{k,\Phi} r_k^* - d_{k,\Phi})' (r - r_k^*) \geq 0, \qquad \forall r \text{ with } \Phi r \in \hat{S} \Big\}$$

and

$$V_k^* = \Big\{ v_k^* | \Psi v_k^* \in \hat{S}, (C_{k,\Psi} v_k^* - d_{k,\Psi})' (v - v_k^*) \geq 0, \qquad \forall v \text{ with } \Psi v \in \hat{S} \Big\}$$

are in one-to-one correspondence via the transformation $V_k^* = BR_k^*$, so the corresponding sets of high-dimensional solutions ΦR_k^* and ΨV_k^* are equal.

Proof: Let $\psi(i)'$ denote the rows of Ψ, so that

$$\phi(i)' = \psi(i)' B, \qquad i = 1, \ldots, n.$$

Using (31), (32), we have

$$\begin{aligned} C_{k,\Phi} &= \frac{1}{k+1} \sum_{t=0}^{k} \phi(i_t) \left(\phi(i_t) - \frac{a_{i_t j_t}}{p_{i_t j_t}} \phi(j_t) \right)' \\ &= \frac{1}{k+1} \sum_{t=0}^{k} B' \psi(i_t) \left(\psi(i_t)' B - \frac{a_{i_t j_t}}{p_{i_t j_t}} \psi(j_t)' B \right) \\ &= B' C_{k,\Psi} B \end{aligned}$$

and similarly

$$d_{k,\Phi} = B' d_{k,\Psi}.$$

We have that $r_k^* \in R_k^*$ if and only if

$$(C_{k,\Phi} r_k^* - d_{k,\Phi})' (r - r_k^*) \geq 0, \qquad \forall r \text{ with } \Phi r \in \hat{S}$$

or equivalently

$$(B' C_{k,\Psi} B r_k^* - B' d_{k,\Psi})' (r - r_k^*) \geq 0, \qquad \forall r \text{ with } \Phi r \in \hat{S}$$

or equivalently, by introducing $v = Br$ and $v_k^* = Br_k^*$,

$$(C_{k,\Psi} v_k^* - d_{k,\Psi})' (v - v_k^*) \geq 0, \qquad \forall v \text{ with } \Phi v \in \hat{S}.$$

It follows that $V_k^* = BR_k^*$. ■

Note that the preceding proposition depends on using the specific simulation process of (31), (32), so that the equations $C_{k,\Phi} = B' C_{k,\Psi} B$, and $d_{k,\Phi} = B' d_{k,\Psi}$ hold. For a different simulation process that satisfies the consistency property $C_k \to C$, $d_k \to d$, the scale-free property can be guaranteed to hold only in the limit as $k \to \infty$.

B. Simulation-Based Iterative Methods

Let us now consider a simulation-based version of the deterministic iterative method (21). It is given by

$$r_{k+1} = P_{D_k, \hat{R}} \left[r_k - \gamma D_k^{-1} (C_k r_k - d_k) \right] \tag{34}$$

where C_k and d_k are the simulation-based estimates of (31), (32), D_k is chosen so that $D_k \to D$, and D is a positive definite symmetric scaling matrix. Using (31), (32) it can be written as

$$r_{k+1} = P_{D_k, \hat{R}} \left[r_k - \frac{\gamma}{k+1} \sum_{t=0}^{k} \phi(i_t) q_{k,t} \right]$$

where

$$q_{k,t} = \phi(i_t)' r_k - \frac{a_{i_t j_t}}{p_{i_t j_t}} \phi(j_t)' r_k - b_{i_t}, \qquad t \leq k$$

is a generalized form of TD [cf. (5)].

One possibility is a simulation-based approximation D_k to $D = \Phi' \Xi \Phi$:

$$D_k = \frac{1}{k+1} \sum_{t=0}^{k} \phi(i_t) \phi(i_t)' \tag{35}$$

or

$$D_k = \frac{1}{k+1} \left(\beta I + \sum_{t=0}^{k} \phi(i_t) \phi(i_t)' \right) \tag{36}$$

where βI is a positive multiple of the identity (to ensure that D_k is positive definite). In the unconstrained case ($\hat{R} = \Re^s$), this is the approximate projected Jacobi method given in [8], which for an approximate DP/policy evaluation problem, reduces to the LSPE method.

Another possibility is to let D_k be a diagonal approximation to $\Phi' \Xi \Phi$, obtained by discarding the off-diagonal terms of the matrix (35) or (36). This facilitates the stepsize choice, since a stepsize γ close to 1 usually works well.

The special case of iteration (34), where $\hat{R} = \Re^s$ and D_k is the identity

$$r_{k+1} = r_k - \gamma (C_k r_k - d_k) \tag{37}$$

can be written as [cf. (31), (32)]

$$r_{k+1} = r_k - \frac{\gamma}{k+1} \sum_{t=0}^{k} \phi(i_t) q_{k,t}. \tag{38}$$

This algorithm is simple and is reminiscent of the TD(0) method of approximate DP [cf. (6)], which when extended for solution of general linear fixed point problems, takes the form

$$r_{k+1} = r_k - \gamma_k \phi(i_k) q_{k,k} \tag{39}$$

where γ_k is a stepsize that diminishes to 0 at an appropriately fast rate [such as $\gamma_k = \gamma/(k+1)$]; see [8]. The difference is that the preceding TD(0)-like method (39) uses only the last TD term, whereas the iteration (38) uses a time average of all the preceding TD terms. Just like the simple deterministic iteration (23), both the multiple sample iteration (37) and its single sample TD(0)-like version (39) generate iterates that lie in the manifold

$$r_0 + \mathrm{Ra}(\Phi')$$

and converge to the projection of r_0 onto the manifold $R^* = \{r | \Phi r = x^*\}$, regardless of the choice of Φ (cf. Fig. 2). As a special case, this behavior is also exhibited by TD(0) for approximate DP: its convergence does not depend on Φ having full rank, as is universally assumed in the literature.

Let us also mention the FPKF algorithm [11], which may be viewed as a scaled version of the preceding TD(0)-like method. When extended to our more general setting, it has the form

$$r_{k+1} = r_k - \gamma_k D_k^{-1} \phi(i_k) q_{k,k}$$

where D_k is a positive definite symmetric matrix, which may be generated by (35) or (36). Similar to the preceding TD(0)-like method (39), it is reminiscent of the simulation-based iteration (34), but uses only the last simulation sample.

C. *Regularization Methods for the Nearly Singular Case*

Let us consider the VI (33) for the unconstrained case where $\hat{S} = S$ and $\hat{R} = \Re^s$

$$C_k r = d_k. \tag{40}$$

If C is nonsingular but is "nearly singular" (has a very large ratio of largest to smallest singular value), C_k will be invertible for sufficiently large k, but the solution $C_k^{-1} d_k$ will be highly sensitive to the simulation noise errors $C_k - C$ and $d_k - d$. This is a well-known phenomenon from the theory of nearly singular linear equations, whose solution is highly sensitive to roundoff error in the problem data.

To get a rough sense of the effect of the simulation error, consider the one-dimensional case ($s = 1$) and a nearly singular C. For $d_k \equiv 1$, the equation approximation approach can be viewed as a process of approximate inversion of a small nonzero number C, which is estimated with simulation error ϵ. The absolute and relative errors are

$$E = \frac{1}{C+\epsilon} - \frac{1}{C}, \qquad E_r = \frac{E}{1/C}.$$

By a first order Taylor series expansion around $\epsilon = 0$, we obtain for small ϵ

$$E \approx \left. \frac{\partial\left(1/(C+\epsilon)\right)}{\partial \epsilon} \right|_{\epsilon=0} \epsilon = -\frac{\epsilon}{C^2}, \qquad E_r \approx -\frac{\epsilon}{C}.$$

Thus for the estimate $1/(C+\epsilon)$ to be reliable, we must have $|\epsilon| \ll |C|$. If N independent samples are used to estimate c, the variance of ϵ is proportional to $1/N$, so for a small relative error, N must be much larger than $1/C^2$. Thus, as C approaches 0, the amount of sampling required for reliable simulation-based inversion increases very fast.

To reduce this type of sensitivity, we may use a regularization approach, which is well-known in the theory of the proximal point algorithm for monotone variational inequalities (see Martinet [29], Rockafellar [30], or the text by Facchinei and Pang [25], Section 12.3). In particular, we approximate the equation $C_k r = d_k$ by

$$(C_k + \beta I) r = d_k + \beta \bar{r} \tag{41}$$

where β is a positive scalar and $\bar{r}$ is some guess of the solution $r^* = C^{-1} d$.

We may also start with (41) with $\bar{r} = r_k$ and iterate, thereby obtaining the iteration

$$r_{k+1} = (C_k + \beta I)^{-1} (d_k + \beta r_k)$$

which can also be written as

$$r_{k+1} = r_k - (C_k + \beta I)^{-1} (C_k r_k - d_k). \tag{42}$$

The convergence of this iteration can be proved, assuming that C is positive definite, based on the fact $C_k \to C$ and convergence results for the proximal point algorithm

$$r_{k+1} = r_k - (C + \beta I)^{-1} (C r_k - d)$$

for solving the equation $Cr = d$.[6]

We may also use an alternative regularization approach, based on a conversion to a least squares problem (also used in a related simulation-based equation approximation context by Wang, Polydorides, and Bertsekas [31]). We introduce a

[6] In the more general case of the VI (33), where $\hat{R} \neq \Re^s$, (41) should be replaced by the problem of finding $\hat{r} \in \hat{R}$ that solves the VI

$$(C_k \hat{r} - d_k + \beta(\hat{r} - \bar{r}))' (r - \hat{r}) \geq 0, \qquad \forall r \in \hat{R}.$$

This VI is strongly monotone if $C_k + \beta I$ is a positive definite matrix. If C is positive definite, asymptotically $C_k + \beta I$ becomes positive definite for all $\beta > 0$, since $C_k \to C$. The algorithm (42) should be replaced by the algorithm that solves for $r_{k+1} \in \hat{R}$ the VI

$$(C_k r_{k+1} - d_k + \beta(r_{k+1} - r_k))' (r - r_{k+1}) \geq 0, \qquad \forall r \in \hat{R};$$

cf. [25], Section 12.3.

positive definite symmetric matrix Σ_k and replace the equation $C_k r = d_k$ with minimization of

$$(C_k r - d_k)' \Sigma_k^{-1} (C_k r - d_k)$$

over $r \in \Re^s$. We then iterate according to

$$r_{k+1} = \arg \min_{r \in \Re^s} \left\{ (C_k r - d_k)' \Sigma_k^{-1} (C_k r - d_k) + \beta \| r - r_k \|^2 \right\} \quad (43)$$

or equivalently

$$r_{k+1} = r_k - \left(C_k' \Sigma_k^{-1} C_k + \beta I \right)^{-1} C_k' \Sigma_k^{-1} (C_k r_k - d_k) \quad (44)$$

where β is a positive scalar (a regularization parameter). If $C_k \to C$, $d_k \to d$, and $\{\Sigma_k^{-1}\}$ is bounded, this iteration can be shown to converge to $r^* = C^{-1} d$, assuming that C is nonsingular. The reason is that the matrix

$$\left(C' \Sigma^{-1} C + \beta I \right)^{-1} C' \Sigma^{-1} C \quad (45)$$

has eigenvalues in the interval (0,1) for any $\beta > 0$. To see this, let $\lambda_1, \ldots, \lambda_s$ be the eigenvalues of $C' \Sigma^{-1} C$ and let $U \Lambda U'$ be its singular value decomposition, where $\Lambda = \text{diag}\{\lambda_1, \ldots, \lambda_s\}$ and U is a unitary matrix ($UU' = I$). We also have

$$C' \Sigma^{-1} C + \beta I = U (\Lambda + \beta I) U'$$

so

$$\begin{aligned} \left(C' \Sigma^{-1} C + \beta I \right)^{-1} C' \Sigma^{-1} C &= \left(U (\Lambda + \beta I) U' \right)^{-1} U \Lambda U' \\ &= U (\Lambda + \beta I)^{-1} \Lambda U'. \end{aligned}$$

It follows that the eigenvalues of the above matrix are $\lambda_i / (\lambda_i + \beta)$, $i = 1, \ldots, s$, and lie in the interval (0,1), so the eigenvalues of $I - \left(C' \Sigma^{-1} C + \beta I \right)^{-1} C' \Sigma^{-1} C$ lie within the unit circle and the convergence of iteration (44) follows for the case where Σ is constant. The proof for the case where Σ is variable is similar. Note that the preceding convergence argument does not require positive definiteness of C, only that C is nonsingular so that $C' \Sigma^{-1} C$ is positive definite.

Actually, the deterministic version of iteration (44)

$$r_{k+1} = r_k - \left(C' \Sigma^{-1} C + \beta I \right)^{-1} C' \Sigma^{-1} (C r_k - d) \quad (46)$$

converges to a solution of the equation $Cr = d$ even if C is singular, as long as $Cr = d$ has a solution. The reason is that the iteration is a special case of the proximal point algorithm for minimizing $(Cr - d)' \Sigma^{-1} (Cr - d)$ [cf. (43)]. From known results about this algorithm (see [29], [30]) it follows that the iteration (46) converges to a minimizing point of $(Cr - d)' \Sigma^{-1} (Cr - d)$. Whether the simulation-based approximation (44) has similarly strong convergence properties is a plausible conjecture that merits investigation.

D. Rate of Convergence Issues

We will now discuss a practically important property regarding asymptotic convergence rate. It can be shown that *all the iterative simulation-based iterations of the form* (34), (42), *and* (44) *perform identically in the long run, as long as they converge* (a phenomenon first described for the LSPE context in the paper [14]). The reason is that the corresponding deterministic methods (21) and (46) have a linear convergence rate, which is fast relative to the slow convergence rate of the simulation-generated D_k, C_k, and d_k. As a result the iterations (34), (42), and (44) operate on two time scales (see, e.g., Borkar [32], Ch. 6): the slow time scale at which D_k, C_k, and d_k change, and the fast time scale at which r_k adapts to changes in D_k, C_k, and d_k. It follows that there is convergence in the fast time scale before there is appreciable change in the slow time scale. Roughly speaking, r_k "sees D_k, C_k, and d_k as effectively constant", so that for large k, r_k is essentially equal to the corresponding limit of iterations (34), (42), and (44) with D_k, C_k, and d_k held fixed. This limit is a vector r_k^* that satisfies

$$(C_k r_k^* - d_k)' (r - r_k^*) \geq 0, \qquad \forall r \in \hat{R}.$$

Assuming that Φ has full rank, it can be shown that the high-dimensional sequence Φr_k generated by iterations (34), (42), and (44) "tracks" the sequence Φr_k^* in the sense that for any norm $\| \cdot \|$,

$$\| \Phi r_k - \Phi r_k^* \| \ll \| \Phi r_k - \Phi r^* \|, \qquad \text{for large } k \quad (47)$$

independent of the choice of the scaling matrix D that is approximated by D_k. The proof uses a two-time scale argument, which is long but very similar to the one of [15], [33] for the approximate DP context and LSPE. It will not be given in this paper.

Since for a given subspace S and any Φ that generates S, the high-dimensional sequence Φr_k^* does not depend on Φ (by Prop. 3), the simulation-based iterations (34), (42), and (44) (for any D_k and γ that lead to convergence) *produce asymptotically the same high-dimensional sequence* Φr_k, *regardless of the choices of* Φ, D_k, *and* γ! By this we mean that for different choices of Φ, D_k, and γ, the sequences Φr_k^* and Φr_k (for all Φ, D_k, and γ) converge onto each other faster than they converge to their common limit Φr^* (the unique solution of the projected equation). Some illustrative computational results can be found in [7].

Of course the preceding description refers to the long-term convergence behavior of the methods. In various contexts involving limited simulation, such as DP applications involving policy iteration, the short-term convergence behavior of the methods is also important (one may use few samples per policy, as in optimistic policy iteration methods), and this behavior depends on γ and Φ. Moreover, in practice it may be desirable to trade off extra overhead in the computation of the matrix multiplying $(C_k r_k - d_k)$ [e.g., the matrix D_k^{-1} with D_k as given by (35), (36), or the matrix $(C_k + \beta I)^{-1}$ as in (42), or the matrix $\left(C_k' \Sigma_k^{-1} C_k + \beta I \right)^{-1} C_k' \Sigma_k^{-1}$ as in (44)] with the convenience of knowing a suitable stepsize value that guarantees convergence (e.g., $\gamma = 1$). By comparison, the short-term convergence of the simple iteration (37) ($D_k = I$) may be slow, and a suitable value of γ for its convergence may be hard to find.

When Φ does not have full rank, a similar analysis of the convergence rate issues may be attempted. Even in the unconstrained case where $\hat{R} = \Re^s$, this analysis, as an initial step, must deal with the difficulty of defining the analog of the high-dimensional sequence $\Phi C_k^{-1} d_k$, for example by using the pseudoinverse of C_k in place of its inverse. The details are considerably more complex and are beyond the scope of the present paper.

E. Multistep Simulation-Based Implementations

Let us now consider the algorithms of the preceding sections, with T replaced by a multistep version that has the same fixed points. One possibility is to use T^ℓ, the ℓth power of T, with $\ell > 1$, or to use $T^{(\lambda)}$ given by

$$T^{(\lambda)} = (1-\lambda)\sum_{\ell=0}^{\infty} \lambda^\ell T^{\ell+1}$$

where $\lambda \in (0,1)$ is such that the preceding infinite series is convergent, i.e., λA has eigenvalues strictly within the unit circle. We will focus on $T^{(\lambda)}$, and consider applying variants of the preceding simulation algorithms to find a fixed point of $T^{(\lambda)}$ in place of T. This idea is inherent in the $\mathrm{TD}(\lambda)$, $\mathrm{LSTD}(\lambda)$, and $\mathrm{LSPE}(\lambda)$ methods, and its motivation is extensively discussed in the approximate DP literature (see also [8] for the nonDP case).

To extend the methods developed so far to $\lambda > 0$, we note that the mapping $T^{(\lambda)}$ can be written as

$$T^{(\lambda)}x = A^{(\lambda)}x + b^{(\lambda)}$$

where

$$A^{(\lambda)} = (1-\lambda)\sum_{\ell=0}^{\infty} \lambda^\ell A^{\ell+1}, \qquad b^{(\lambda)} = \sum_{\ell=0}^{\infty} \lambda^\ell A^\ell b.$$

In the unconstrained case ($\hat{S} = S$ and $\hat{R} = \Re^s$), given $C_k^{(\lambda)}$ and $d_k^{(\lambda)}$, by analogy to the case $\lambda = 0$, the projected equation is

$$\Phi r = \Pi T^{(\lambda)} x = C^{(\lambda)} r - d^{(\lambda)}$$

where

$$C^{(\lambda)} = \Phi' \Xi \left(I - A^{(\lambda)}\right)\Phi, \qquad d^{(\lambda)} = \Phi' \Xi b^{(\lambda)}.$$

Similar to the earlier simulation approach, we may construct simulation-based approximations $C_k^{(\lambda)}$ and $d_k^{(\lambda)}$ to $C^{(\lambda)}$ and $d^{(\lambda)}$, respectively. A method for doing so is described in [8], and requires a restriction in the row and column sampling schemes [the row index sequence $\{i_0, i_1, \ldots\}$ is generated using a Markov chain with transition matrix P, the same as the one used for generating the transition sequence $\{(i_0, j_0), (i_1, j_1), \ldots\}$]. The solution of the projected equation may be approximated by

$$\left(C_k^{(\lambda)}\right)^{-1} d_k^{(\lambda)};$$

this is a generalization of the $\mathrm{LSTD}(\lambda)$ method of approximate DP. Similarly, the iterative method

$$r_{k+1} = r_k - \gamma D_k^{-1}\left(C_k^{(\lambda)} r_k - d_k^{(\lambda)}\right)$$

is a multistep variant of the iterative method (34), and contains as a special case the $\mathrm{LSPE}(\lambda)$ method of approximate DP. The convergence and convergence rate analysis given earlier for the case $\lambda = 0$ generalizes in straightforward manner to the case $\lambda > 0$. Analogs for the constrained case ($\hat{S} \neq S$) are similarly obtained.

V. Conclusions

In this paper we have considered the solution of projected equations that are derived from large-scale fixed point problems by using low-dimensional subspace approximation. We have proposed a unifying framework, based on a new connection with VIs, for a broadly applicable methodology that uses simulation and low-order calculations. Prominent within our framework are iterative algorithms that generalize TD methods for approximate DP. New algorithms of this type offer benefits such as implementation convenience (a matrix Φ that is rank-deficient), reduced overhead (no matrix inversion at each iteration), and the ability to use projection on a polyhedral subset of the approximation subspace.

We have investigated both deterministic iterative methods and simulation-based versions that use low-dimensional calculations. There is a sharp distinction between the two types of methods in terms of the choices of the direction matrix D, the stepsize γ, and the matrix Φ that represents the approximation subspace S. The convergence rate of the deterministic methods is profoundly affected by D, γ, and Φ. By contrast, the asymptotic convergence rate of the simulation-based versions is largely unaffected by D, γ, and Φ, but instead depends on the choice of the row and column sampling mechanisms in ways that are not fully understood at present. Various mathematical convergence issues, extensions to nonlinear special cases of the mapping T, and related optimization applications are interesting subjects for further investigation.

Acknowledgment

The author would like to gratefully acknowledge V. Borkar, J. Tsitsiklis, and H. Yu for helpful comments and Me. Wang for many discussions and assistance with computational experimentation.

References

[1] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, 3rd ed. Belmont, MA: Athena Scientific, 2007, vol. II.

[2] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. New York: Wiley, 1994.

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

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

[5] W. B. Powell, *Approximate Dynamic Programming: Solving the Curses of Dimensionality*. New York: Wiley, 2007.

[6] R. S. Sutton, "Learning to predict by the methods of temporal differences," *Mach. Learning*, vol. 3, pp. 9–44, 1988.

[7] D. P. Bertsekas, "Projected equations, variational inequalities, and temporal difference methods," MIT, Lab. for Information and Decision Systems Report LIDS-P-2808, 2009.

[8] D. P. Bertsekas and H. Yu, "Projected equation methods for approximate solution of large linear systems," *J. Comput. Appl. Math.*, vol. 227, pp. 27–50, 2009.

[9] H. Yu, "Least squares temporal difference methods: An analysis under general conditions," Univ. Helsinki, Dept. Computer Science Tech. Rep. C-2010-39, 2010.

[10] V. R. Konda, "Actor-Critic Algorithms," Ph.D. dissertation, Dept. EECS, MIT, Cambridge, MA, 2002.

[11] D. S. Choi and B. V. Roy, "A generalized Kalman filter for fixed point approximation and efficient temporal-difference learning," *Discrete Event Dynamic Syst.: Theory Appl.*, vol. 16, pp. 207–239, 2006.

[12] D. P. Bertsekas and S. Ioffe, Temporal Differences-Based Policy Iteration and Applications in Neuro-Dynamic Programming MIT, Cambridge, MA, Lab. Info. Decision Syst. Rep. LIDS-P-2349, 1996.

[13] A. Nedić and D. P. Bertsekas, "Least squares policy evaluation algorithms with linear function approximation," *Discrete Event Dynamic Syst.: Theory Appl.*, vol. 13, pp. 79–110, 2003.

[14] D. P. Bertsekas, V. S. Borkar, and A. Nedić, "Improved temporal difference methods with linear function approximation," in *Learning and Approximate Dynamic Programming*, J. Si, A. Barto, W. Powell, and D. Wunsch, Eds. New York: IEEE Press, 2004.

[15] H. Yu and D. P. Bertsekas, "Convergence results for some temporal difference methods based on least squares," MIT, Lab. Information and Decision Systems Report LIDS-P-2697, 2006.

[16] D. P. de Farias and B. V. Roy, "On the existence of fixed points for approximate value iteration and temporal-difference learning," *J. Optimization Theory Appl.*, vol. 105, pp. 589–608, 2000.

[17] J. N. Tsitsiklis and B. V. Roy, "An analysis of temporal-difference learning with function approximation," *IEEE Trans. Autom. Contr.*, vol. 42, no. 5, pp. 674–690, May 1997.

[18] J. N. Tsitsiklis and B. V. Roy, "Optimal stopping of Markov processes: Hilbert space theory, approximation algorithms, and an application to pricing financial derivatives," *IEEE Trans. Autom. Contr.*, vol. 44, no. 10, pp. 1840–1851, Oct. 1999.

[19] H. Yu and D. P. Bertsekas, "A least squares Q-Learning algorithm for optimal stopping problems," MIT, Lab. Information and Decision Systems Report LIDS-P-2731, 2007.

[20] S. J. Bradtke and A. G. Barto, "Linear least-squares algorithms for temporal difference learning," *Mach. Learning*, vol. 22, pp. 33–57, 1996.

[21] J. A. Boyan, "Technical update: Least-squares temporal difference learning," *Mach. Learning*, vol. 49, pp. 1–15, 2002.

[22] M. A. Krasnoselskii *et al., Approximate Solution of Operator Equations*. Groningen: Wolters-Noordhoff Pub., 1972, translated by D. Louvish.

[23] C. A. J. Fletcher, *Computational Galerkin Methods*. New York: Springer-Verlag, 1984.

[24] Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd ed. Philadelphia, PA: SIAM, 2003.

[25] F. Facchinei and J. S. Pang, *Finite-Dimensional Variational Inequalities and Complementarity Problems*. New York: Springer-Verlag, 2003.

[26] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Englewood Cliffs, NJ: Prentice-Hall, 1989, republished by Athena Scientific, Belmont, MA, 1997.

[27] D. P. Bertsekas and E. Gafni, "Projection methods for variational inequalities with applications to the traffic assignment problem," *Math. Progr. Studies*, vol. 17, pp. 139–159, 1982.

[28] J. N. Tsitsiklis and B. V. Roy, "Average cost temporal-difference learning," *Automatica*, vol. 35, pp. 1799–1808, 1999.

[29] B. Martinet, "Regularisation d' inequations variationnelles par approximations successives," *Revue Francaise d'Informatique et de Recherche Operationnelle*, vol. 2, pp. 154–159, 1970.

[30] R. T. Rockafellar, "Monotone operators and the proximal point algorithm," *SIAM J. Contr. Optimization*, vol. 14, pp. 877–898, 1976.

[31] M. Wang, N. Polydorides, and D. P. Bertsekas, "Approximate simulation-based solution of large-scale least squares problems," MIT, Lab. Information and Decision Systems Report LIDS-P-2819, 2009.

[32] V. S. Borkar, *Stochastic Approximation: A Dynamical Systems Viewpoint*. Cambridge, U.K.: Cambridge Univ. Press, 2008.

[33] H. Yu and D. P. Bertsekas, "Convergence results for some temporal difference methods based on least squares," *IEEE Trans. Autom. Contr.*, vol. 54, no. 7, pp. 1515–1531, Jul. 2009.

Dimitri P. Bertsekas studied engineering at the National Technical University of Athens, Greece, received the M.S. degree in electrical engineering at the George Washington University, Washington, DC in 1969, and the Ph.D. degree in system science in 1971 at the Massachusetts Institute of Technology, Cambridge.

He has held faculty positions with the Engineering-Economic Systems Dept., Stanford University (1971–1974) and the Electrical Engineering Department of the University of Illinois, Urbana (1974–1979). Since 1979 he has been teaching at the Electrical Engineering and Computer Science Department of the Massachusetts Institute of Technology (MIT), Cambridge, where he is currently McAfee Professor of Engineering. He consults regularly with private industry and has held editorial positions in several journals. His research at MIT spans several fields, including optimization, control, large-scale computation, and data communication networks, and is closely tied to his teaching and book authoring activities. He has written numerous research papers, and fourteen books, several of which are used as textbooks in MIT classes. His recent books are *Dynamic Programming and Optimal Control: 3rd Edition* (Athena Scientific, 2007), *Introduction to Probability: 2nd Edition* (Athena Scientific, 2008), and *Convex Optimization Theory* (Athena Scientific, 2009).

Professor Bertsekas was awarded the INFORMS 1997 Prize for Research Excellence in the Interface Between Operations Research and Computer Science for his book "Neuro-Dynamic Programming" (co-authored with John Tsitsiklis), the 2000 Greek National Award for Operations Research, the 2001 ACC John R. Ragazzini Education Award, and the 2009 INFORMS Expository Writing Award. In 2001, he was elected to the United States National Academy of Engineering.