March 2009; Revised: October 2009 **Report LIDS-P-2808**

Projected Equations, Variational Inequalities, and Temporal Difference Methods

Dimitri P. Bertsekas

Laboratory for Information and Decision Systems (LIDS)

Massachusetts Institute of Technology

MA 02139, USA

Email: dimitrib@mit.edu

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 a class of iterative 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. We discuss deterministic and simulation-based versions of our methods and we show a sharp qualitative distinction between them: the performance of the former is greatly affected by direction and feature scaling, yet the latter asymptotically perform identically, regardless of scaling.

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

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

so solving the projected equation (1) is equivalent to finding r that satisfies

$$\Phi r = \Pi T(\Phi r), \qquad r \in \hat{R}. \tag{4}$$

Note that our choice of a fixed point format is not strictly necessary for our development. 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 allowed to be 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. 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 [Ber07], Puterman [Put94]. 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 [BeT96], Sutton and Barto [SuB98], and Powell [Pow07]; Bertsekas [Ber07] 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 [BeT96], [SuB98], [Ber07].

[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.

These are iterative simulation-based methods that produce a sequence $\{r_k\}$ converging to a solution of the projected Bellman's equation $\Phi r = \Pi(A\Phi r + b)$. They generate a sequence of indices $\{i_0, i_1, \ldots\}$ using the Markov chain associated with P, and they use the temporal differences (TD) defined by

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

where $\phi(i)'$ denotes the ith row of the matrix Φ. The original method known as TD(0), due to Sutton [Sut88], 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 $\Phi'\Xi(\Phi r - A\Phi r - b) = 0$ (which is a necessary and sufficient condition for r to solve the projected equation). Indeed, using Eqs. (5)-(6), it can be seen that the direction of change $\phi(i_k)q_{k,k}$ 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 [Kon02]), alternative iterative methods have been proposed. One of them is the Fixed Point Kalman Filter (FPKF, proposed by Choi and Van Roy [ChV06]) 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 type of scaled stochastic approximation method. The choice

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

is suggested in [ChV06] and some favorable computational results are reported, albeit without theoretical proof of convergence rate superiority over TD(0).

Another alternative to TD(0) is the Least Squares Policy Evaluation algorithm (LSPE, proposed by Bertsekas and Ioffe [BeI96]; see also Nedić and Bertsekas, [NeB03], Bertsekas, Borkar, and Nedić [BBN04], Yu and Bertsekas [YuB06]) and given by

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

where D_k is given by Eq. (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

[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 TD(λ) [Sut88]. In this paper, we mostly focus on $\lambda = 0$, but our algorithms and qualitative conclusions apply to general $\lambda \in [0,1)$ (see [Ber07] and [Ber09] for further discussion, and Section IV.F).

[cf. Eqs. (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 [BeI96], [Kon02], and [YuB06]).

The validity of all these 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 [BeT96], [DFV00] (a notable exception is optimal stopping problems, as shown by Tsitsiklis and Van Roy [TsV97], [TsV99b]; see also Yu and Bertsekas [YuB07]).

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., [Kra72], [Fle84]). 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 [Saa03]). 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 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 $x \in S$. This 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 1A.

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 direct or iterative methods. Thus the methodology can be applied only to problems of dimension small enough, so that the linear algebra calculations to obtain the exact form of the projected equation are feasible. This motivates the extension of simulation-based approximate DP methods to more general non-DP contexts where n can be extremely large.

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 shortly explain. This equivalence has not been noticed earlier and is the

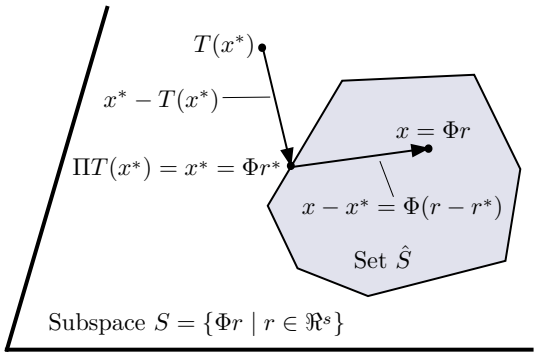

Fig. 1. Equivalence of a projected equation $x^* = \Pi T(x^*)$ with the variational inequality $f(\Phi r^*)'\Phi(r - r^*) \geq 0$, $\forall\ r \in \hat{R}$, where $f(x) = \Xi\big(x - T(x)\big)$ and $\hat{R} = \{r \mid \Phi r \in \hat{S}\}$. By the properties of projection, we have $x^* = \Pi T(x^*)$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

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.,

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

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. This can be equivalently written 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

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

and $\hat{R} = \{r \mid \Phi r \in \hat{S}\}$ [cf. Eq. (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.

There are several interesting problems from optimization and game theory that can be modeled by VIs (see e.g., [BeT89], [PaF03]), and the connection with projected equations can be used as a basis for an approximate solution approach; see Section V.

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

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

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

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 1A), 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 Eq. (5). This is similar to the LSPE method (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 (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 [Ber09]).
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 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 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 [BeY09].

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 apply the iteration (13) to projected equations and we focus on the case where T is linear. 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. In Section IV we develop simulation-based algorithms, which require low (s-dimensional) calculations only. In the process we recover the existing TD methods for approximate DP, including LSPE [through the more general form (14)], and the Least Squares Temporal Differences method (LSTD; proposed by Bradtke and Barto [BrB96], and followed up by Boyan [Boy02], and Nedić and Bertsekas [NeB03]). We also derive the connections to TD(0) and FPKF. Generally, in simulation-based implementations, the slower speed of simulation dominates, and based on this, we prove an interesting fact: *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). Finally in Section V, we discuss some simulation-based optimization applications of our VI framework.

As a byproduct of our analysis we prove a new result: Φ need not have full rank for convergence of iterative TD-type methods (unless this is required for invertibility of D_k). This full rank assumption has been universally made in the convergence analyses of TD(0) and related methods thus far. 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):

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

Equivalently, using the definition of projection with respect to $\|\cdot\|_D$, the iteration can be written in terms of 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 method 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 [BeT89], Ch. 3).

The properties of this method are closely linked with monotonicity properties of F (see e.g., Pang and Facchinei [PaF03] 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

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

(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 $\Re^n$ 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 [PaF03]. When $\hat{R} = \Re^s$, the VI is equivalent to the linear system $Cr = d$.

The standard convergence result for the projection method (13) (see e.g., [BeT89], Section 3.5.3, or [PaF03], 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., [BeT89], p. 270).

Let now F have the special form [cf. Eq. (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. Eq. (13)]

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

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, the paper [BeG82] has 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).

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 Eq. (12) is strongly monotone over $\hat{S}$.*

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

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

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 [TsV97] (Lemma 9); see also [BeY09], 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 Eq. (12) is strongly monotone over S.*

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

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

since vectors of the form $x - \Pi x$ are orthogonal (with respect to the norm $\|\cdot\|_\Xi$) to vectors in 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 $x_1, x_2 \in S$,

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

where the first equation follows from Eq. (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 [BeT96], [TsV97], [TsV99a], [Ber07], [YuB06]). An example is discounted or average cost DP, where $T(x) = \alpha P x + b$, 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 [BeY09] 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

We now discuss the iteration (13) as applied to the case where 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 $f(x) = \Xi\big(x - T(x)\big)$ of the underlying VI [cf. Eqs. (11)-(12)] is strongly monotone over $\hat{S}$ (this is guaranteed under contraction assumptions on T or ΠT, as per Props. 1 and 2).

We have $\Phi' f(\Phi r) = \Phi'\Xi(\Phi r - A\Phi r - b)$, so that

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

with

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

Since f is strongly monotone over $\hat{S}$, the VI $f(x^*)'(x - x^*) \geq 0$ for all $x \in \hat{S}$, and its equivalent projected equation $x = \Pi T(x)$ have a unique solution $x^* \in \hat{S}$. In the low-dimensional space $\Re^s$, this VI is written as

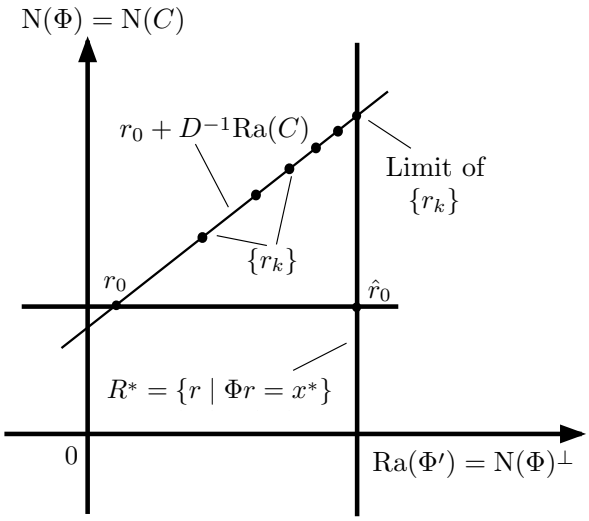

Fig. 2. Illustration of the convergence process of the iteration (22) in the case where Φ does not have full rank. The iteration converges to the intersection of the solution set R^* with the linear manifold $r_0 + D^{-1}\mathrm{Ra}(C)$. If $D = I$, the iteration converges to $\hat{r}_0$, the orthogonal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

$f(\Phi r^*)'\Phi(r - r^*) \geq 0$ for all $r \in \hat{R}$, or [cf. Eqs. (18)-(19)]

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

and is equivalent to the projected equation $\Phi r = \Pi T(\Phi r)$. The set of its solutions is $R^* = \{r \in \hat{R} \mid \Phi r = x^*\}$, and if Φ has full rank, R^* consists of a single point. The iteration (13) takes the form

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

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

In the case where $\hat{S} = S$ and r is unconstrained, 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 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. Eq. (19)], so it is orthogonal to $\mathrm{N}(\Phi)$ and hence to R^*.

Iteration (22) converges if and only $I - \gamma D^{-1}C$ 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

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

[4]To see this, note that $\mathrm{Ra}(C)$ is contained in $\mathrm{Ra}(\Phi')$ [cf. Eq. (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.

Then it can be shown (see [Ber07], [BeY09]) 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 Eq. (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 show in Section IV.

A. *Effects of Feature Scaling*

The iteration

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

[cf. Eq. (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 Eq. (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):

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

or

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

[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.

and

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

or

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

A straightforward quadratic programming duality argument shows that

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

so from Eq. (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 Eq. (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 $\Phi = \Psi B$,

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

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. Eq. (27)].

IV. Simulation-Based Methods

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

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

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

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

[cf. Eq. (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 indices $\{1,\ldots,n\}$.

We consider a simulation process introduced in [BeY09]. We generate a sequence of indices $\{i_0,i_1,\ldots\}$ (*row sampling*), and a sequence of transitions between indices $\big\{(i_0,j_0),(i_1,j_1),\ldots\big\}$ (*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 $\big\{(i_0,j_0),(i_1,j_1),\ldots\big\}$ 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 [BeY09]).

A. Equation Approximation Approach

A simulation-based noniterative approach to solve the VI (29) is to generate the matrix C_k and vector d_k using Eqs. (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. On the other hand, if Φ has full rank, the VI (29) is strongly monotone, and 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 case where $\hat{S}=S$, strong monotonicity is equivalent to positive definiteness of C_k, in which case 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 [BrB96] (also described in [Ber07]). 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 Eqs. (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,*

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

and

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

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 Eqs. (31)-(32), we have

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

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

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

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 Eqs. (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. Approximate Solution for Nearly Singular Cases

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

$$C_k r = d_k. \tag{34}$$

If C is nonsingular, the same is true for C_k, for sufficiently large k, but if C is nearly singular, 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 reduce this type of sensitivity, we may use a regularization approach, which is well-known in the algorithmic theory of monotone variational inequalities. In particular, we replace Eq. (34) with the equivalent equation

$$\Sigma^{-1} C_k r = \Sigma^{-1} d_k, \tag{35}$$

where Σ is some positive definite symmetric matrix. We then approximate this equation by

$$(\Sigma^{-1} C_k + \beta I)\hat{r} = \Sigma^{-1} d_k + \beta \bar{r}, \tag{36}$$

where β is a positive scalar and $\bar{r}$ is some guess of the solution $r^* = C^{-1} d$. In the more general case of the VI (33), where $\hat{R} \neq \Re^s$, the preceding equation should be replaced by the VI

$$\big(\Sigma^{-1}(C_k \hat{r} - d_k) + \beta(\hat{r} - \bar{r})\big)'(r - \hat{r}) \geq 0, \qquad \forall\ r \in \hat{R}, \tag{37}$$

which can be shown to be strongly monotone if C_k is a positive definite matrix.

In the case $\hat{R} = \Re^s$, we may also start with Eq. (36) with $\bar{r} = \hat{r}_k$ and iterate using a variable matrix Σ_k:

$$\hat{r}_{k+1} = (\Sigma_k^{-1} C_k + \beta I)^{-1}(\Sigma_k^{-1} d_k + \beta \hat{r}_k).$$

This algorithm can also be written as

$$\hat{r}_{k+1} = \hat{r}_k - (\Sigma_k^{-1} C_k + \beta I)^{-1} \Sigma_k^{-1}(C_k \hat{r}_k - d_k), \tag{38}$$

and bears similarity to the iterative method (40), to be presented in the next subsection. In the case where C_k is replaced by the positive definite matrix C and d_k is replaced by d, the algorithm is a special case of the proximal point algorithm applied to monotone VIs (see Martinet [Mar70] and Rockafellar [Roc76]). A similar iteration based on Eq. (37) can be used in the more general case where $\hat{R} \neq \Re^s$.

Another type of regularization approach for the case $\hat{R} = \Re^s$, is to replace the system $C_k r = d_k$ with the equivalent system

$$C_k' \Sigma_k^{-1} C_k r = C_k' \Sigma_k^{-1} d_k,$$

where Σ_k is some positive definite symmetric matrix. The corresponding iterative algorithm, which is analogous to Eq. (38), is given by

$$\hat{r}_{k+1} = \hat{r}_k - (C_k' \Sigma_k^{-1} C_k + \beta I)^{-1} C_k' \Sigma_k^{-1}(C_k \hat{r}_k - d_k). \tag{39}$$

If $C_k \to C$, $d_k \to d$, and $\{\Sigma_k^{-1}\}$ is bounded, this algorithm can be shown to converge to $r^* = C^{-1} d$, assuming that C is nonsingular. The reason is that the matrix

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

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 $\Lambda = \mathrm{diag}\{\lambda_1, \ldots, \lambda_s\}$ and U is a unitary matrix ($UU' = I$). We also have $C'\Sigma^{-1}C + \beta I = U(\Lambda + \beta I)U'$, so

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

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 convergence of iteration (39) follows for the case where Σ is constant. The proof for the case where Σ is variable is similar.

C. Simulation-Based Iterative Methods

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

$$r_{k+1} = P_{D_k,\hat{R}}\left[r_k - \gamma D_k^{-1}(C_k r_k - d_k)\right], \tag{40}$$

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

$$r_{k+1} = P_{D_k,\hat{R}}\left[r_k - \frac{\gamma}{k+1}\sum_{t=0}^{k}\phi(i_t)q_{k,t}\right], \tag{41}$$

where

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

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

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

$$D_k = \frac{1}{k+1}\sum_{t=0}^{k}\phi(i_t)\phi(i_t)', \tag{42}$$

or

$$D_k = \frac{1}{k+1}\left(\beta I + \sum_{t=0}^{k}\phi(i_t)\phi(i_t)'\right), \tag{43}$$

where βI is a positive multiple of the identity (to ensure that D_k is positive definite). When $\hat{S} = S$, this is the approximate projected Jacobi method given in [BeY09], which in the case of 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 (42) or (43). This facilitates the stepsize choice, since a stepsize γ close to 1 usually works well.

The simple special case of iteration (40), where $\hat{S} = S$ and D_k is the identity,

$$r_{k+1} = r_k - \gamma(C_k r_k - d_k), \tag{44}$$

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

$$r_{k+1} = r_k - \frac{\gamma}{k+1}\sum_{t=0}^{k}\phi(i_t)q_{k,t}. \tag{45}$$

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

$$r_{k+1} = r_k - \gamma_k \phi(i_k) q_{k,k}, \tag{46}$$

where γ_k is a stepsize that diminishes to 0 at an appropriately fast rate [such as $\gamma_k = \gamma/(k+1)$]; see [BeY09]. The difference is that the preceding TD(0)-like method (46) uses only the last TD term, whereas the iteration (45) uses a time average of all the preceding TD terms. Just like the simple deterministic iteration (23), both the multiple sample iteration (44) and its single sample TD(0)-like version (46) generate iterates that lie in the manifold

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

and converge to the projection of r_0 onto the manifold $R^* = \{r \mid \Phi r = x^*\}$, 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 [ChV06], 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 Eqs. (42) or (43). Similar to the preceding TD(0)-like method (46), it is reminiscent of the simulation-based iteration (40), but uses only the last simulation sample.

D. Rate of Convergence Issues

Let us now discuss a practically important property regarding asymptotic convergence rate. It can be shown that *all the simulation-based iterations of the form (40) perform identically in the long run, as long as they converge*. The reason is that the corresponding deterministic method (21) has 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 iteration (40) operates on two time scales (see, e.g., Borkar [Bor08], 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. As a result, essentially, 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 iteration (40) 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 iteration (40) "tracks" the sequence Φr_k^* in the sense that for any norm $\|\cdot\|$,

$$\|\Phi r_k - \Phi r_k^*\| << \|\Phi r_k - \Phi r^*\|, \qquad \text{for large } k, \tag{47}$$

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 one used in [YuB06] for the approximate DP context and LSPE (see also [BBN04]). It will not be given in this paper. Some illustrative computational results can be found in [Ber09].

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), for any D_k and γ that lead to convergence *the simulation-based iteration (40) produces 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).

When Φ does not have full rank a similar analysis of the convergence rate issues is possible, but the details are considerably more complex and are beyond the scope of the present paper.

E. Computational Experimentation

The algorithms of this paper have been validated by computational experiments involving the three test problems of [YuB06]. Our main conclusions regarding convergence, and the contrasting role of the scaling matrix D and the feature matrix Φ in deterministic versus simulation-based algorithms were verified. In particular, the high-dimensional sequences produced by various simulation-based iterations exhibited substantially different behavior in the early iterations where the deterministic character of the algorithm dominates, and the choice of D_k and Φ makes a substantial difference. After the early iterations, however, the sequences produced with different choices of D_k and Φ converged onto each other much faster than they converged to their eventual limit [cf. Eq. (47)]. This is similar to the behavior observed in the tests of [YuB06] that compare LSPE and LSTD.

We show some typical computational results obtained with problem 3 of [YuB06] using simulation-based methods. This is a DP average cost policy evaluation problem involving a "slow-mixing" Markov chain with 100 states (the matrix A here is the transition probability matrix of the Markov chain, and the vector ξ is the steady-state distribution of the chain). We used two different 100×3 randomly generated matrices Φ and Ψ, having the same range space S, we run a single simulation trajectory involving 10000 samples, and we calculated $C_{k,\Phi}$, $C_{k,\Psi}$, $d_{k,\Phi}$, $d_{k,\Psi}$ using Eqs. (31)-(32). For this problem, T does not have a unique fixed point, but ΠT does (according to the results of [YuB06]). We also verified that ΠT is a contraction with respect to the norm $\|\cdot\|_\xi$, which guarantees convergence of the algorithms tested. We compared four different simulation-based algorithms. These are:

- The equation approximation method, which calculates

$$r_{k,\Phi} = C_{k,\Phi}^{-1} d_{k,\Phi}, \qquad r_{k,\Psi} = C_{k,\Phi}^{-1} d_{k,\Psi}.$$

- The approximate Jacobi/LSPE method, which calculates $\hat{r}_{k,\Phi}$ and $\hat{r}_{k,\Psi}$ using the iteration (40), and direction scaling matrices $D = \Phi' \Xi \Phi$ and $D = \Psi' \Xi \Psi$, respectively, approximated by D_k as computed by the simulation formula (42). The stepsize was $\gamma = 1$.
- The diagonal approximation to the Jacobi/LSPE method, which calculates $\tilde{r}_{k,\Phi}$ and $\tilde{r}_{k,\Psi}$ using direction scaling matrices obtained from $\Phi' \Xi \Phi$ and $\Psi' \Xi \Psi$ by setting to 0 the off-diagonal components [again computed by the simulation formula (42)]. The stepsize was $\gamma = 1$.

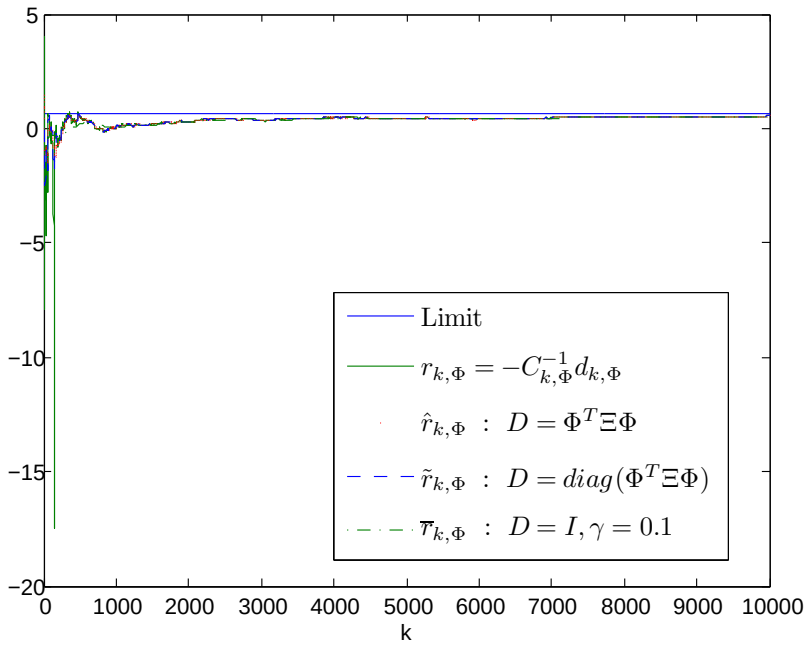

Fig. 3. First component of low-dimensional sequences corresponding to Φ.

- The simple iteration (40), where $D_k = I$, and a stepsize $\gamma = 0.1$ when both Φ and Ψ were used. The iterates are denoted $\bar{r}_{k,\Phi}$ and $\bar{r}_{k,\Psi}$, respectively.

In all these algorithms, the initial matrices $C_{0,\Phi}$, $C_{0,\Psi}$ were formed with a batch of 50 samples, to ensure that they are invertible. The graphs in the following five figures give some typical results. In Figs. 3 and 4 only the first components of the corresponding vector sequences are shown.

Figures 3 and 4 show the low-dimensional sequences generated by the four methods, for the two feature scaling matrices Φ and Ψ, respectively. These are asymptotically identical (very close to each other for $k > 100$ for the first three methods, and for $k > 1000$ for the fourth method), as predicted by the theory. The sequences and their limits depend on whether Φ or Ψ is used (compare Figs. 3 and 4).

Figures 5, 6, and 7 compare the high-dimensional sequence $\{\Phi r_{k,\Phi}\}$ with each of $\{\Phi \hat{r}_{k,\Phi}\}$, $\{\Phi \tilde{r}_{k,\Phi}\}$, $\{\Phi \bar{r}_{k,\Phi}\}$, $\{\Psi \hat{r}_{k,\Psi}\}$, $\{\Psi \tilde{r}_{k,\Psi}\}$, $\{\Psi \bar{r}_{k,\Psi}\}$. Again these six sequences coincide asymptotically with $\{\Phi r_{k,\Phi}\}$ (which itself coincides with $\{\Psi r_{k,\Psi}\}$, as shown in Prop. 3). In particular, all sequences $\{\Phi \hat{r}_{k,\Phi}\}$, $\{\Phi \tilde{r}_{k,\Phi}\}$, $\{\Phi \bar{r}_{k,\Phi}\}$, $\{\Psi \hat{r}_{k,\Psi}\}$, $\{\Psi \tilde{r}_{k,\Psi}\}$, and $\{\Psi \bar{r}_{k,\Psi}\}$ converge to $\{\Phi r_{k,\Phi}\}$ faster than they converge to their limit, as shown in the figures.

The results indicate that the simple iteration (40), where $D_k = I$ is a little slower than the others, but not dramatically so. This is due to the fact that the deterministic version of the iteration tends to be slower than the others.

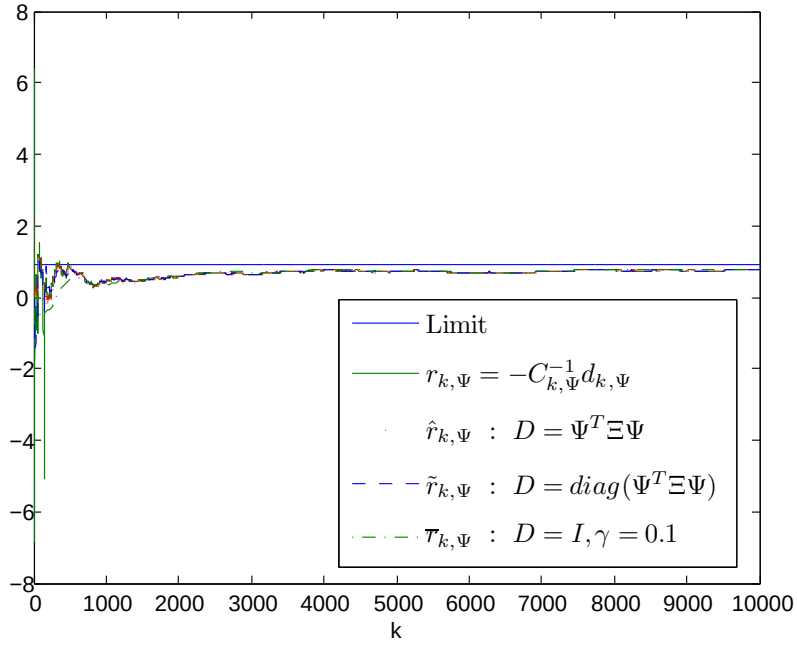

Fig. 4. First component of low-dimensional sequences corresponding to Ψ.

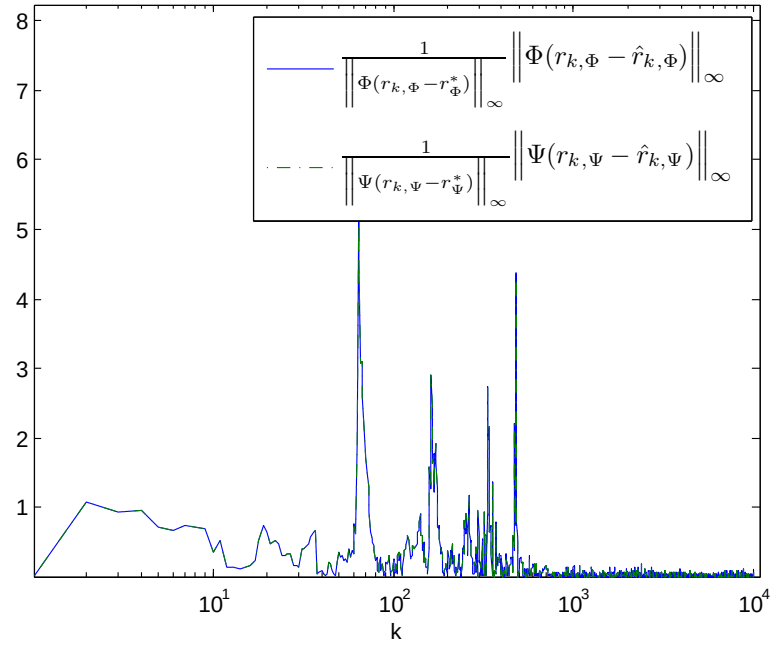

Fig. 5. Convergence behavior of the simulation-based projected Jacobi iteration. The iterates using Φ and Ψ are identical, since the method is scale-free.

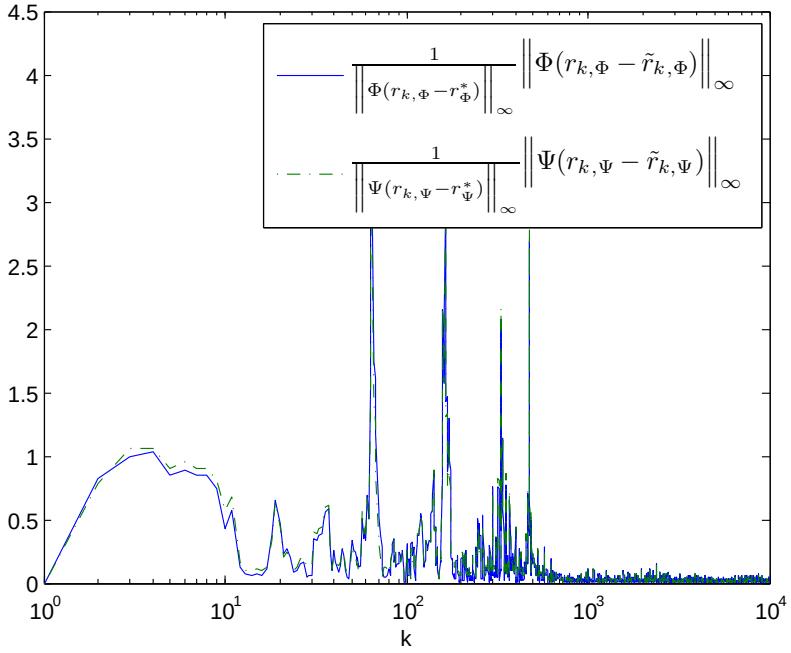

Fig. 6. Convergence behavior of the diagonal approximation of the simulation-based projected Jacobi iteration.

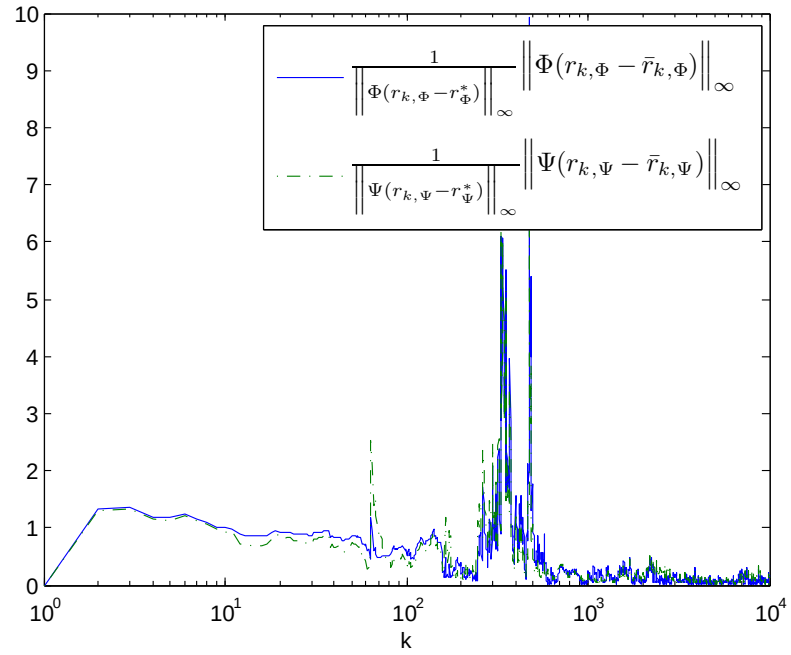

Fig. 7. Convergence behavior of the simulation-based simple iteration.

F. Multistep Simulation-Based Implementations

One may consider replacing T with 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 TD(λ), LSTD(λ), and LSPE(λ) methods, and its motivation is extensively discussed in the approximate DP literature (see also [BeY08] for the nonDP case).

The methods developed so far correspond to $\lambda = 0$, but can be extended to $\lambda > 0$. In particular, it is straightforward to verify that the mapping $T^{(\lambda)}$ can be written as

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

where

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

By analogy to the case $\lambda = 0$, the projected equation is

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

where

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

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 [BeY08], 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 $\{(i_0,j_0),(i_1,j_1),\ldots\}$]. Given $C_k^{(\lambda)}$ and $d_k^{(\lambda)}$, the solution of the projected equation may be approximated by

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

this is a generalization of the LSTD(λ) 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 (40), and contains as a special case the LSPE(λ) method of approximate DP. The convergence and convergence rate analysis given earlier for the case $\lambda = 0$ generalizes in straightforward manner to the case $\lambda > 0$.

V. Additional Optimization Applications

In earlier sections we focused on exploiting the connection of VI and projection equations for solving approximately linear fixed point and DP-related optimization problems. In this section we provide some examples of alternative (nonDP) optimization contexts where this connection may be useful.

Example 1 (Optimization over a Convex Set)**.** Consider the minimization of a differentiable convex function $H : \Re^n \mapsto \Re$ over a convex set $\hat{S}$, and the following low-dimensional approximation:

$$\min_{\Phi r \in \hat{S}} H(\Phi r).$$

This problem is equivalent to solving the necessary and sufficient condition for optimality:

$$\nabla H(\Phi r^*)' \Phi (r - r^*) \geq 0, \quad \forall\ r \in \hat{R} \equiv \{r \mid \Phi r \in \hat{S}\},$$

which is a VI of the form (11). We may convert it to the projected equation $x = \Pi T(x)$, where Π is the projection on $\hat{R}$ with respect to $\|\cdot\|_\Xi$ and

$$T(x) = x - \Xi^{-1} \nabla H(x)$$

[cf. Eq. (12)]. Note that if H is a positive definite quadratic function, then T can be shown to be a contraction mapping provided the norm of Ξ^{-1} is sufficiently small.

When the problem is unconstrained, i.e., $\hat{S} = \Re^n$, the projected equation is equivalent to the system of equations

$$\Phi' \nabla H(\Phi r) = 0.$$

One possibility, noted in [BeY09], is to solve this system iteratively, using Newton's method. In this method r_{k+1} is determined from r_k by solving the linear system

$$\big(\Phi' \nabla^2 H(\Phi r_k) \Phi\big)(r_{k+1} - r_k) + \Phi' \nabla H(\Phi r_k) = 0,$$

which may be done with simulation-based methods that use s-dimensional operations only. In particular, for an $n \times n$ matrix M, the matrix

$$\Phi' M \Phi = \sum_{i=1}^n \sum_{j=1}^n m_{ij} \phi(i) \phi(j)'$$

can be estimated as

$$\Phi' M \Phi \approx \frac{1}{k+1} \sum_{t=0}^k \frac{m_{i_t j_t}}{\zeta_{i_t j_t}} \phi(i_t) \phi(j_t)'$$

where k is a large number and:

- ζ_{ij}, $i, j = 1, \ldots, n$, are probabilities satisfying

$$\sum_{i=1}^n \sum_{j=1}^n \zeta_{ij} = 1, \qquad \zeta_{ij} > 0 \ \text{ if } \ m_{ij} \neq 0.$$

- $\big\{(i_0, j_0), (i_1, j_1), \ldots\big\}$ is an independent random sequence of pairs taking the value (i, j) with probability ζ_{ij}.

Similarly, for a vector $b \in \Re^n$, the vector

$$\Phi' b = \sum_{i=1}^n \sum_{j=1}^n \phi(i) b_j$$

can be estimated as

$$\Phi' b \approx \frac{1}{k+1} \sum_{t=0}^{k} \frac{1}{\zeta_{i_t j_t}} \phi(i_t) b_{j_t}.$$

When the problem is constrained, a potentially serious difficulty is that the set $\hat{R} = \{r \mid \Phi r \in \hat{S}\}$ may be hard to handle because it may involve a large number (order n) of inequalities. For example when $\hat{S}$ is the nonnegative orthant $\{x \mid x \geq 0\}$, the set $\hat{R}$ involves the n inequalities $\phi(i)' r \geq 0$, where $\phi(i)'$ are the rows of Φ. In this case, one may consider approximations of $\hat{R}$, involving for example constraint sampling (dropping some of the constraints), constraint generation (dynamically adding and/or dropping constraints), constraint aggregation (combining constraints), exploitation of special problem structure, and special types of basis functions that implicitly take into account the constraints; see [CaC05], [DFV04], [GKP03], [GrH91], [MoK99], [PaT00], [ScP01], [TrZ93], [TrZ97], for related methods, analysis, and discussion of this issue in the context of approximate linear programming methods in DP and beyond.

A possible alternative is to eliminate the constraint $x \in \hat{S}$ by using a penalty or interior point method, leading to a sequence of unconstrained optimization problems. These problems may be addressed by using Newton's method, possibly in combination with a constraint sampling or constraint generation scheme. This may be an interesting subject for further investigation. □

Example 2 (Optimization Subject to Linear Constraints)**.** Consider the problem of minimizing a differentiable convex function $H(x)$ subject to $x \in \hat{S}$ and the linear constraints $Ax \leq b$. Under a standard condition ($\hat{S}$ is convex and contains a feasible solution in its relative interior), the problem is equivalent to solving the necessary and sufficient condition for optimality:

$$\big(\nabla H(x^*) + A'\mu^*\big)'(x - x^*) \geq 0, \qquad (Ax^* - b)'(\mu - \mu^*) \geq 0,$$

for all $x \in \hat{S}$ and $\mu \geq 0$, where $\mu^* \geq 0$ is a Lagrange multiplier. This is a VI in (x, μ). We may introduce low-dimensional representations Φr for x and Wv for μ, thereby obtaining a low-dimensional VI, which is in turn equivalent to a projected equation, as discussed in Example 1. This VI may be addressed using appropriate algorithms: if H is linear or quadratic, one may use linear or quadratic programming methods, or iterative methods such as the extragradient method ([PaF03], Section 12.1.2) or penalty and interior point methods. Again this formulation suffers from the difficulty of a possibly intractable number of constraints for (r, v), so for large-dimensional problems, some scheme for dealing with these constraints may be needed, such as the ones mentioned in the preceding example. □

From the preceding two examples and the equivalence between projected equations and VIs of the form (11)-(12), we may obtain an interesting insight for approximate DP: methods that are alternative to TD, and are based on linear cost function approximation, such as approximate linear programming, the Bellman equation error method, or aggregation methods, can be classified as projected equation methods, just like TD. Thus the projected equation methodology can be viewed as a general framework, encompassing all the principal simulation-based approaches currently available in approximate DP.

Example 3 (Noncooperative Games)**.** Consider a game with m players, each choosing a strategy x_i belonging to a closed convex set $X_i \subset \Re^{n_i}$. Player i has a cost function $H_i(x_1, \ldots, x_m)$, where $H_i : \Re^{n_i} \mapsto \Re$ is convex and

differentiable. The problem is to find an equilibrium strategy $x^* = (x_1^*, \ldots, x_m^*) \in X_1 \times \cdots \times X_m$, i.e., one that satisfies

$$H_i(x_1^*, \ldots, x_m^*) \leq H_i(x_1^*, \ldots, x_{i-1}^*, x_i, x_{i+1}^*, \ldots, x_m^*),$$

for all $x_i \in X_i$, $i = 1, \ldots, m$. A low-dimensional approximation is the game problem where each x_i is replaced by $\Phi_i r_i$ with $r_i \in R_i \equiv \{r \mid \Phi_i r_i \in X_i\}$. This problem is equivalent to solving the player-by-player optimality condition

$$\nabla_i H_i(\Phi_1 r_1^*, \ldots, \Phi_m r_m^*)' \Phi_i (r_i - r_i^*) \geq 0, \quad \forall\ r_i \in R_i, \quad i = 1, \ldots, m,$$

where $\nabla_i H_i$ is the gradient of H_i with respect to x_i. This is a VI of the form (11). Similar to Example 1, we may convert this VI to the projected equation

$$x_i = \Pi_i T_i(x_1, \ldots, x_m), \qquad i = 1, \ldots, m,$$

where Π_i is the projection on R_i with respect to the standard Euclidean norm and

$$T_i(x_1, \ldots, x_m) = x_i - \alpha \nabla H_i(x_1, \ldots, x_m)$$

with α being a positive scalar. □

VI. 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 need not have full rank), reduced overhead (no matrix inversion at each iteration), and the ability to use projection on a restricted 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 and the atrix Φ that represents the approximation subspace S. The convergence rate of the deterministic methods is profoundly affected by D and Φ. By contrast, the convergence rate of the simulation-based versions is largely unaffected by D 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.

Acknowledgments

Helpful comments by Vivek Borkar, John Tsitsiklis, and Janey Yu are gratefully acknowledged. Thanks are due to Mengdi Wang for discussions and assistance with computational experimentation. This work was supported by NSF Grant ECCS-0801549.

REFERENCES

[BBN04] Bertsekas, D. P., Borkar, V. S., and Nedić, A., 2004. "Improved Temporal Difference Methods with Linear Function Approximation," in Learning and Approximate Dynamic Programming, by J. Si, A. Barto, W. Powell, and D. Wunsch (Eds.), IEEE Press, N. Y.

[BeG82] Bertsekas, D. P., and Gafni, E., 1982. "Projection Methods for Variational Inequalities with Applications to the Traffic Assignment Problem," Math. Progr. Studies, Vol. 17, pp. 139-159.

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

[BeT89] Bertsekas, D. P., and Tsitsiklis, J. N., 1989. Parallel and Distributed Computation: Numerical Methods, Prentice-Hall, Englewood Cliffs, N. J; republished by Athena Scientific, Belmont, MA, 1997.

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

[BeY09] Bertsekas, D. P., and Yu, H., 2009. "Projected Equation Methods for Approximate Solution of Large Linear Systems," Journal of Computational and Applied Mathematics, Vol. 227, pp. 27-50.

[Ber07] Bertsekas, D. P., 2007. Dynamic Programming and Optimal Control, 3rd Edition, Vol. II, Athena Scientific, Belmont, MA.

[Ber09] D. P. Bertsekas, 2009. "Projected Equations, Variational Inequalities, and Temporal Difference Methods," Lab. for Information and Decision Systems Report LIDS-P-2808, MIT.

[Bor08] Borkar, V. S., 2008. Stochastic Approximation: A Dynamical Systems Viewpoint, Cambridge Univ. Press.

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

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

[CaC05] Calafiore, G., and Campi, M. C., 2005. "Uncertain Convex Programs: Randomized Solutions and Confidence Levels," Math. Programming, Vol. 102, pp. 25-46.

[ChV06] Choi, D. S., and Van Roy, B., 2006. "A Generalized Kalman Filter for Fixed Point Approximation and Efficient Temporal-Difference Learning, Discrete Event Dynamic Systems: Theory and Applications, Vol. 16, pp. 207-239.

[DFV00] de Farias, D. P., and Van Roy, B., 2000. "On the Existence of Fixed Points for Approximate Value Iteration and Temporal-Difference Learning," J. of Optimization Theory and Applications, Vol. 105.

[DFV04] de Farias, D. P., and Van Roy, B., 2004. "On Constraint Sampling in the Linear Programming Approach to Approximate Dynamic Programming," Mathematics of Operations Research, Vol. 29, pp. 462-478.

[Fle84] Fletcher, C. A. J., 1984. Computational Galerkin Methods, Springer-Verlag, N. Y.

[GKP03] Guestrin, C., Koller, D., and Parr, R., 2003. "Efficient Solution Algorithms for Factored MDP," J. Artificial Intelligence Res., Vol. 19, pp. 399-468.

[GrH91] Grotschel, M., and Holland, O., 1991. "Solution of Large-Scale Symmetric Travelling Salesman Problems, Math. Programming, Vol. 51, pp. 141-202.

[Kon02] Konda, V. R., 2002. Actor-Critic Algorithms, Ph.D. Thesis, Dept. of EECS, M.I.T., Cambridge, MA.

[Kra72] Krasnoselskii, M. A., et. al, 1972. Approximate Solution of Operator Equations, Translated by D. Louvish, Wolters-Noordhoff Pub., Groningen.

[Mar70] Martinet, B., "Regularisation d'inequations variationnelles par approximations successives", Rev. Francaise Inf. Rech. Oper., pp. 154-159, 1970.

[MoK99] Morrison, J. R., and Kumar, P. R., 1999. "New Linear Program Performance Bounds for Queueing Networks," J. Opt. Theory and Applications, Vol. 100, pp. 575-597.

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

[PaF03] Pang, J. S., and Facchinei, F., 2003. Finite-Dimensional Variational Inequalities and Complementarity Problems, Springer-Verlag, N. Y.

[PaT00] Paschalidis, I. C., and Tsitsiklis, J. N., 2000. "Congestion-Dependent Pricing of Network Services," IEEE/ACM Transactions on Networking, Vol. 8, pp. 171-184.

[Pow07] Powell, W. B., 2007. Approximate Dynamic Programming: Solving the Curses of Dimensionality, Wiley, N. Y.

[Put94] Puterman, M. L., 1994. Markov Decision Processes: Discrete Stochastic Dynamic Programming, J. Wiley, N. Y.

[Roc76] Rockafellar, R. T., "Monotone Operators and the Proximal Point Algorithm", SIAM J. on Control and Optimization, Vol. 14, 1976, pp. 877-898.

[Saa03] Saad, Y., 2003. Iterative Methods for Sparse Linear Systems, 2nd edition, SIAM, Phila., PA.

[ScP01] Schuurmans, D., and Patrascu, R., 2001. "Direct Value-Approximation for Factored MDPs," Advances in Neural Information Processing Systems, Vol. 14, MIT Press, Cambridge, MA, pp. 1579-1586.

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

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

[TrZ93] Trick, M. S., and Zin, S., 1993. "A Linear Programming Approach to Solving Dynamic Programs, Unpublished Manuscript.

[TrZ97] Trick, M. S., and Zin, S., 1993. "Spline Approximations to Value Functions: A Linear Programming Approach," Macroeconomics Dynamics, Vol. 1, pp. 255-277.

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

[TsV99a] Tsitsiklis, J. N., and Van Roy, B., 1999. "Average Cost Temporal-Difference Learning," Automatica, Vol. 35, pp. 1799-1808.

[TsV99b] Tsitsiklis, J. N., and Van Roy, B., 1999. "Optimal Stopping of Markov Processes: Hilbert Space Theory, Approximation Algorithms, and an Application to Pricing Financial Derivatives", IEEE Transactions on Automatic Control, Vol. 44, pp. 1840-1851.

[YuB06] Yu, H., and Bertsekas, D. P., 2006. "Convergence Results for Some Temporal Difference Methods Based on Least Squares," Lab. for Information and Decision Systems Report 2697, MIT; IEEE Trans. on Aut. Control, Vol. 54, 2009, pp. 1515-153.

[YuB07] Yu, H., and Bertsekas, D. P., 2007. "A Least Squares Q-Learning Algorithm for Optimal Stopping Problems," Lab. for Information and Decision Systems Report LIDS-P-2731, MIT.