Journal of Computational and Applied Mathematics 227 (2009) 27–50

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

Projected equation methods for approximate solution of large linear systems

Dimitri P. Bertsekas [a,*], Huizhen Yu [b]

[a] Department of Electr. Engineering and Comp. Science, M.I.T., Cambridge, MA, 02139, United States

[b] Helsinki Institute for Information Technology, University of Helsinki, Finland

ARTICLE INFO

Article history:
Received 26 February 2008
Received in revised form 29 May 2008

Keywords:
Linear equations
Projected equations
Dynamic programming
Temporal differences
Simulation
Value iteration
Jacobi method

ABSTRACT

We consider linear systems of equations and solution approximations derived by projection on a low-dimensional subspace. We propose stochastic iterative algorithms, based on simulation, which converge to the approximate solution and are suitable for very large-dimensional problems. The algorithms are extensions of recent approximate dynamic programming methods, known as temporal difference methods, which solve a projected form of Bellman's equation by using simulation-based approximations to this equation, or by using a projected value iteration method.

1. Introduction

In this paper we focus on systems of linear equations of the form

$$x = Ax + b, \tag{1.1}$$

where A is an $n \times n$ matrix and b is a column vector in the n-dimensional space $\Re^n$. We propose methods to compute an approximate solution within a subspace spanned by a relatively small number of basis functions.

Our motivation comes from recent advances in the field of dynamic programming (DP), where large systems of equations of the form (1.1) appear in the context of evaluation of the cost of a stationary policy in a Markovian decision problem. In this DP context, we are given an n-state Markov chain with transition probability matrix P, which evolves for an infinite number of discrete time periods, and a cost vector $g \in \Re^n$, whose components g_i represent the costs of being at the corresponding states $i = 1, \ldots, n$, for a single time period. The problem is to evaluate the total cost vector

$$x^* = \sum_{t=0}^{\infty} \alpha^t P^t g,$$

where $\alpha \in (0, 1)$ is a discount factor, and the components x_i represent the total expected α-discounted cost over an infinite number of time periods, starting from the corresponding states $i = 1, \ldots, n$. It is well known that x is the unique solution of the equation

$$x = \alpha Px + g,$$

* Corresponding author.
E-mail address: dimitrib@mit.edu (D.P. Bertsekas).

0377-0427/$ – see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2008.07.037

and furthermore, x can also be computed iteratively by the Jacobi method $x_{t+1} = \alpha P x_t + g$ (also known as value iteration in the context of DP), since the mapping $x \mapsto \alpha P x + g$ is a contraction with respect to the sup norm; see textbooks on DP, such as for example [6], or [20].

We focus on the case where n is very large, and it may be worth (even imperative) considering a low-dimensional approximation of a solution within a subspace

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

where the columns of the $n \times s$ matrix Φ can be viewed as basis functions. This type of approximation approach has been the subject of much recent research in approximate DP, where several methods have been proposed and substantial computational experience has been accumulated. The most popular of these methods use projection with respect to the weighted Euclidean norm given by

$$\|x\|_\xi = \sqrt{\sum_{i=1}^n \xi_i x_i^2},$$

where $\xi \in \Re^n$ is a probability distribution with positive components. We denote by Π the projection operation onto S with respect to this norm (while Π depends on ξ, we do not show the dependence, since the associated vector ξ will always be clear from the context). The aforementioned methods for approximating the solution of the DP equation $x = \alpha P x + g$ aim to solve the equation

$$\Phi r = \Pi(\alpha P \Phi r + b)$$

with ξ being the invariant distribution of the transition probability matrix P (which is assumed irreducible; i.e., has a single recurrent class and no transient states). The more general methods of this paper aim to approximate a fixed point of the mapping

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

by solving the equation

$$\Phi r = \Pi T(\Phi r) = \Pi(A\Phi r + b), \tag{1.2}$$

where the projection norm $\|\cdot\|_\xi$ is determined in part by the structure of A in a way to induce some desired property. We view Π as a matrix and *we implicitly assume throughout that $I - \Pi A$ is invertible*. Thus, for a given ξ, there is a unique vector y^* such that $y^* = \Pi T(y^*)$, and we have $y^* = \Phi r^*$ for some $r^* \in \Re^s$ (if Φ has linearly independent columns, r^* is also unique).

To evaluate the distance between Φr^* and a fixed point x^* of T, we write

$$x^* - \Phi r^* = x^* - \Pi x^* + \Pi x^* - \Phi r^* = x^* - \Pi x^* + \Pi T x^* - \Pi T \Phi r^* = x^* - \Pi x^* + \Pi A(x^* - \Phi r^*), \tag{1.3}$$

from which

$$x^* - \Phi r^* = (I - \Pi A)^{-1}(x^* - \Pi x^*).$$

Thus, we have for any norm $\|\cdot\|$ and fixed point x^* of T

$$\|x^* - \Phi r^*\| \le \|(I - \Pi A)^{-1}\| \, \|x^* - \Pi x^*\|, \tag{1.4}$$

and the approximation error $\|x^* - \Phi r^*\|$ is proportional to the distance of the solution x^* from the approximation subspace. If ΠT is a contraction mapping of modulus $\alpha \in (0, 1)$ with respect to $\|\cdot\|$, from Eq. (1.3), we have

$$\|x^* - \Phi r^*\| \le \|x^* - \Pi x^*\| + \|\Pi T(x^*) - \Pi T(\Phi r^*)\| \le \|x^* - \Pi x^*\| + \alpha \|x^* - \Phi r^*\|,$$

so that

$$\|x^* - \Phi r^*\| \le \frac{1}{1-\alpha} \|x^* - \Pi x^*\|. \tag{1.5}$$

A better bound is obtained when ΠT is a contraction mapping of modulus $\alpha \in [0, 1)$ with respect to a Euclidean norm (e.g., $\|\cdot\|_\xi$). Then, using the Pythagorean Theorem, we have

$$\begin{aligned} \|x^* - \Phi r^*\|^2 &= \|x^* - \Pi x^*\|^2 + \|\Pi x^* - \Phi r^*\|^2 \\ &= \|x^* - \Pi x^*\|^2 + \|\Pi T(x^*) - \Pi T(\Phi r^*)\|^2 \\ &\le \|x^* - \Pi x^*\|^2 + \alpha^2 \|x^* - \Phi r^*\|^2 \end{aligned}$$

from which we obtain

$$\|x^* - \Phi r^*\|^2 \le \frac{1}{1-\alpha^2} \|x^* - \Pi x^*\|^2. \tag{1.6}$$

The error bounds (1.4)–(1.6) depend only on the subspace S and are valid even if Φ does not have full rank, as long as $I - \Pi A$ is invertible and Φr^* is interpreted as the unique solution of the projected equation $y = \Pi T(y)$. For the remainder of the paper, however, *we assume that the columns of Φ are linearly independent*, since this is needed for the subsequently proposed algorithms for solving the projected equation.

In the case where ΠT is a contraction mapping with respect to some norm, there are some additional algorithmic approaches for approximation. In particular, we may consider a Jacobi/fixed point iteration, restricted within S, which involves projection of the iterates onto S:

$$\Phi r_{t+1} = \Pi T(\Phi r_t), \quad t = 0, 1, \ldots. \tag{1.7}$$

In the context of DP this is known as *projected value iteration* (see [6]). It converges to r^*, but is unwieldy when n is very large, because the vector $T(\Phi r_t)$ has dimension n.

The preceding observations suggest that it is desirable for ΠT to be a contraction with respect to some norm. However, this is a complicated issue because *ΠT need not be a contraction with respect to a given norm, even if T is a contraction with respect to that norm*. It may thus be important to choose ξ, and the associated projection Π, in special ways that guarantee that ΠT is a contraction. This question will be discussed in Section 3.

In this paper, we introduce simulation-based algorithms for solving the equation $\Phi r = \Pi T(\Phi r)$. *The key favorable property of these algorithms is that they involve low-dimensional matrices and vectors, so they do not require n-dimensional calculations*. We consider two types of methods:

(a) *Equation approximation methods*, whereby r^* is approximated by $\hat{r}$, the solution of a linear system of the form

$$\Phi r = \hat{\Pi}\hat{T}(\Phi r), \tag{1.8}$$

where $\hat{\Pi}$ and $\hat{T}$ are simulation-based approximations to Π and T, respectively. As the number of simulation samples increases, $\hat{r}$ converges to r^*.

(b) *Approximate Jacobi methods*, which [without explicit calculation of $T(\Phi r_t)$] can be written in the form

$$\Phi r_{t+1} = \Pi T(\Phi r_t) + \epsilon_t, \quad t = 0, 1, \ldots, \tag{1.9}$$

where ϵ_t is a simulation-induced error that diminishes to 0 as the number of simulation samples increases. Similarly to the methods in (a), they do not require n-dimensional calculations, but apply only when ΠT is a contraction with respect to some norm. Then, since ϵ_t converges to 0, asymptotically iteration (1.9) becomes the Jacobi iteration (1.7), and r_t converges to r^*. We will also interpret later iteration (1.9) as a single iteration of an algorithm for solving the system (1.8).

Within the DP context, the approximation methods in (a) above have been proposed in [9,8] (see also the analysis in [18]), and are known as *least squares temporal differences* (LSTD) methods. The approximate Jacobi methods in (b) above have been proposed in [4] (see also the analysis in [18,2,29,6]), and are known as *least squares policy evaluation* (LSPE) methods. An earlier, but computationally less effective method, is TD(λ), which was first proposed by Sutton [23] and was instrumental in launching a substantial body of research on approximate DP in the 1990s (see [7,22,24,25] for discussion, extensions, and analysis of this method). Within the specialized approximate DP context, LSTD, LSPE, and TD(λ) offer some distinct advantages, which make them suitable for the approximate solution of problems involving Markov chains of very large dimension (in a case study where LSPE was used to evaluate the expected score of a game playing strategy, a Markov chain with more than 2^{200} states was involved; see [4] and [7], Section 8.3). These advantages are:

(1) The vector x need not be stored at any time. Furthermore, inner products involving the rows of A need not be computed; this can be critically important if some of the rows are not sparse.
(2) There is a projection norm such that the matrix ΠA is a contraction, so the bound (1.5) applies.
(3) The vector ξ of the projection norm need not be known explicitly. Instead, the values of the components of ξ are naturally incorporated within the simulation as relative frequencies of occurrence of the corresponding states (ξ is the invariant distribution vector of the associated Markov chain).

These advantages, particularly the first, make the simulation-based approach an attractive (possibly the only) option in problems so large that traditional approaches are prohibitively expensive in terms of time and storage. An additional advantage of our methods is that they are far better suited for parallel computation than traditional methods, because the associated simulation is easily parallelizable.

The present paper extends the approximate DP methods just discussed to the case where A does not have the character of a stochastic matrix; just invertibility of $I - \Pi A$ is assumed. An important difficulty in the non-DP context considered here is that there may be no natural choice of ξ (and associated Markov chain to be used in the simulation process) such that ΠT is a contraction. Nonetheless, we show that all of the advantages (1)–(3) of LSTD, LSPE, and TD(λ) within the DP context are preserved under certain conditions, the most prominent of which is

$$|a_{ij}| \le q_{ij}, \quad \forall\, i, j = 1, \ldots, n,$$

where a_{ij} are the components of A and q_{ij} are the transition probabilities of a Markov chain, which is used for simulation. In this case, again ξ is an invariant distribution of the chain and need not be known a priori. This is shown in Section 3, where some examples, including the important special case of a weakly diagonally dominant system, are also discussed.

When the condition $|a_{ij}| \leq q_{ij}$, for all i, j, does not hold, the selection of the Markov chain used for simulation and the associated vector ξ used in the projection operation may be somewhat ad hoc. Furthermore, if ΠA is not a contraction, the approximate Jacobi methods are not valid and the bound (1.5) does not apply. Instead, the bound of Eq. (1.4) applies and the equation approximation methods of Section 2 are valid. Note that when $I - \Pi A$ is nearly singular, the bounds are poor, and the associated equation (1.3) suggests potential difficulties. Still, the methods we propose maintain some important characteristics, namely that the vector x need not be stored at any time, and inner products involving the rows of A need not be computed.

We note that LSTD and LSPE are in fact entire classes of methods, parameterized with a scalar $\lambda \in [0, 1)$. They are called LSTD(λ) and LSPE(λ), respectively, and they use the parameter λ similarly to the method of TD(λ). A value $\lambda > 0$ corresponds to approximating, in place of $x = T(x)$, the equation $x = T^{(\lambda)}(x)$, where $T^{(\lambda)}$ is the mapping

$$T^{(\lambda)} = (1 - \lambda) \sum_{k=0}^{\infty} \lambda^k T^{k+1}. \tag{1.10}$$

Note that the fixed points of T are also fixed points of $T^{(\lambda)}$, and that $T^{(\lambda)}$ coincides with T for $\lambda = 0$. However, it can be seen that when T is a contraction with respect to some norm with modulus $\alpha \in [0, 1)$, $T^{(\lambda)}$ is a contraction with the more favorable modulus

$$\alpha_\lambda = (1 - \lambda) \sum_{k=0}^{\infty} \lambda^k \alpha^{k+1} = \frac{\alpha(1 - \lambda)}{1 - \alpha\lambda}.$$

Thus, when approximating the equation $x = T^{(\lambda)}(x)$, rather than $x = T(x)$, the error bounds (1.5) and (1.6) become more favorable; in fact $\alpha_\lambda \to 0$ as $\lambda \to 1$, so asymptotically, from Eq. (1.5), we obtain optimal approximation: $\|x^* - \Phi r^*\| = \|x^* - \Pi x^*\|$. Furthermore, $T^{(\lambda)}$ and $\Pi T^{(\lambda)}$ are arbitrarily close to 0, if λ is sufficiently close to 1, so they can become contractions with respect to any norm. An important characteristic of our methods is that under certain conditions they can be straightforwardly applied to the equation $x = T^{(\lambda)}(x)$, while this is much harder with traditional methods (see Section 5). However, while the error bounds improve as λ is increased towards 1, the simulation required to solve the equation $\Phi r = \Pi T^{(\lambda)}(\Phi r)$ becomes more time-consuming because the associated simulation samples become more "noisy." This "accuracy–noise" tradeoff is widely recognized in the approximate DP literature (see e.g., the textbook [6] and the references quoted there).

In this paper, we focus on describing the methods, making the connection with their DP antecedents, proving some basic results relating to contraction properties of ΠA, and providing some examples of interesting special cases. Our methodology and analysis, while new almost in their entirety, extend for the most part known ideas from approximate DP. Some of our methods and analysis, however, are new even within the DP context, as will be specifically pointed out later. A more detailed delineation of important relevant classes of problems, the associated formulations, projection norm selections, computational experimentation, and other related issues, are beyond our scope. We note the extended report [5], which provides some additional material and has a lot of overlap with the present paper. Let us also mention that the solution of linear systems of equations by using Monte Carlo methods was suggested by von Neumann and Ulam, as recounted in [13], and has been further discussed in [28,11,12] (see also the survey in [14]). Barto and Duff [3] have pointed out connections and similarities with approximate DP and TD(1). In contrast with the present paper, these works do not consider approximation/projection onto a low-dimensional subspace, and require that T be a contraction with respect to some norm.

The paper is organized as follows. In Section 2, we formulate the simulation framework that underlies our methods, and we discuss the equation approximation approach of (a) above. In Section 3, we discuss the selection of Markov chains, together with some related examples and special cases, for which various contraction properties can be shown. Among others, we derive here some new algorithms for Markovian decision problems, which address the well-known issue of exploration. In Section 4, we develop the approximate Jacobi methods in (b) above, and we discuss their relation with the equation approximation methods in (a) above. In Section 5, we develop multistep analogs of the methods of Sections 2 and 4, in the spirit of the LSTD(λ), LSPE(λ), and TD(λ) methods of approximate DP. The methods of Sections 2–5 assume that the rows of Φ are either explicitly known or can be easily generated when needed. In Section 6, we discuss special methods that use basis functions of the form $A^m g$, $m \geq 0$, where g is some vector the components of which can be exactly computed. These methods bear similarity to Krylov subspace methods (see e.g. [21]), but suffer from the potential difficulty that the rows of $A^m g$ may be hard to compute. We discuss variants of our methods of Sections 2–5 where the rows of $A^m g$ are approximated by simulation of a single sample. These variants are new even in the context of approximate DP ($A = \alpha P$), where generating appropriate basis vectors for cost function approximation is a currently prominent research issue. Finally, in Section 7, we discuss some extensions and related methods, including a general approach for linear least squares problems and applications to some nonlinear fixed point problems. In particular, we generalize approximate DP methods proposed for optimal stopping problems (see [26,10,30]).

Regarding notation, throughout the paper, vectors are considered to be column vectors, and a prime denotes transposition. We generally use subscripts to indicate the scalar components of various vectors and matrices. Vector and matrix inequalities are to be interpreted componentwise. For example, for two matrices A, B, the inequality $A \leq B$ means that $A_{ij} \leq B_{ij}$ for all i and j. For a vector x, we denote by $|x|$ the vector whose components are the absolute values of the

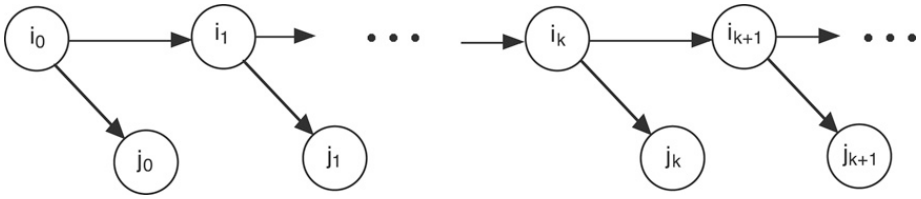

Fig. 2.1. The basic simulation methodology consists of generating a sequence of indices $\{i_0, i_1, \ldots\}$ according to the distribution ξ, and a sequence of transitions $\{(i_0, j_0), (i_1, j_1), \ldots\}$ according to transition probabilities p_{ij}. It is possible that $j_k = i_{k+1}$, but this is not necessary.

components of x, i.e., $|x|_i = |x_i|$ for all i. We use similar notation for matrices, so for example, we denote by $|A|$ the matrix whose components are $|A|_{ij} = |a_{ij}|$ for all $i, j = 1, \ldots, n$.

2. Equation approximation methods

In this section, we discuss the construction of simulation-based approximations to the projected equation $\Phi r = \Pi(A\Phi r + b)$. This methodology descends from the LSTD methods of approximate DP, referred to in Section 1. Let us assume that the positive distribution vector ξ is given. By the definition of projection with respect to $\|\cdot\|_\xi$, the unique solution r^* of this equation satisfies

$$r^* = \arg\min_{r\in\Re^s} \sum_{i=1}^n \xi_i \left(\phi(i)'r - \sum_{j=1}^n a_{ij}\phi(j)'r^* - b_i\right)^2,$$

where $\phi(i)'$ denotes the ith row of the matrix Φ. By setting the gradient of the minimized expression above to 0, we have

$$\sum_{i=1}^n \xi_i\phi(i)\left(\phi(i)'r^* - \sum_{j=1}^n a_{ij}\phi(j)'r^* - b_i\right) = 0.$$

We thus obtain the following equivalent form of the projected equation $\Phi r = \Pi(A\Phi r + b)$:

$$\sum_{i=1}^n \xi_i\phi(i)\left(\phi(i) - \sum_{j=1}^n a_{ij}\phi(j)\right)' r^* = \sum_{i=1}^n \xi_i\phi(i)b_i. \tag{2.1}$$

The key idea of our methodology can be simply explained by focusing on the two expected values with respect to ξ, which appear in the left and right sides of the above equation: *we approximate these two expected values by simulation-obtained sample averages*. When the matrix A is not sparse, we also approximate the summation over $a_{ij}\phi(j)$ by simulation-obtained sample averages. In the most basic form of our methods, we generate a sequence of indices $\{i_0, i_1, \ldots\}$, and a sequence of transitions between indices $\{(i_0, j_0), (i_1, j_1), \ldots\}$. We use any probabilistic mechanism for this, subject to the following two requirements (cf. Fig. 2.1):

(1) 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,

$$\lim_{t\to\infty} \frac{\sum_{k=0}^t \delta(i_k = i)}{t+1} = \xi_i, \quad i = 1, \ldots, n, \tag{2.2}$$

where $\delta(\cdot)$ denotes the indicator function [$\delta(E) = 1$ if the event E has occurred and $\delta(E) = 0$ otherwise].

(2) 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

$$p_{ij} > 0 \quad \text{if } a_{ij} \neq 0, \tag{2.3}$$

in the sense that with probability 1,

$$\lim_{t\to\infty} \frac{\sum_{k=0}^t \delta(i_k = i, j_k = j)}{\sum_{k=0}^t \delta(i_k = i)} = p_{ij}, \quad i, j = 1, \ldots, n. \tag{2.4}$$

At time t, we form the linear equation

$$\sum_{k=0}^{t} \phi(i_k)\left(\phi(i_k) - \frac{a_{i_k j_k}}{p_{i_k j_k}}\phi(j_k)\right)' r = \sum_{k=0}^{t} \phi(i_k) b_{i_k}. \tag{2.5}$$

We claim that this is a valid approximation to Eq. (2.1), the equivalent form of the projected equation.

Indeed, by counting the number of times an index occurs and collecting terms, we can write Eq. (2.5) as

$$\sum_{i=1}^{n} \hat{\xi}_{i,t}\phi(i)\left(\phi(i) - \sum_{j=1}^{n} \hat{p}_{ij,t}\frac{a_{ij}}{p_{ij}}\phi(j)\right)' r = \sum_{i=1}^{n} \hat{\xi}_{i,t}\phi(i) b_i, \tag{2.6}$$

where

$$\hat{\xi}_{i,t} = \frac{\sum_{k=0}^{t} \delta(i_k = i)}{t+1}, \qquad \hat{p}_{ij,t} = \frac{\sum_{k=0}^{t} \delta(i_k = i, j_k = j)}{\sum_{k=0}^{t} \delta(i_k = i)}.$$

In view of the assumption

$$\hat{\xi}_{i,t} \to \xi_i, \qquad \hat{p}_{ij,t} \to p_{ij}, \quad i, j = 1, \ldots, n,$$

[cf. Eqs. (2.2) and (2.4)], by comparing Eqs. (2.1) and (2.6), we see that they asymptotically coincide. Since the solution r^* of the system (2.1) exists and is unique, the same is true for the system (2.6) for all t sufficiently large. Thus, with probability 1,

$$\hat{r}_t \to r^*,$$

where $\hat{r}_t$ is the solution of the system (2.5).

A comparison of Eqs. (2.1) and (2.6) indicates some considerations for selecting the stochastic matrix P. It can be seen that "important" (e.g., large) components a_{ij} should be simulated more often (p_{ij}: large).[1] In particular, if (i, j) is such that $a_{ij} = 0$, there is an incentive to choose $p_{ij} = 0$, since corresponding transitions (i, j) are "wasted" in that they do not contribute to improvement of the approximation of Eq. (2.1) by Eq. (2.6). This suggests that the structure of P should match in some sense the structure of the matrix A, to improve the efficiency of the simulation. On the other hand, the choice of P does not affect the limit of $\Phi\hat{r}_t$, which is the solution Φr^* of the projected equation. By contrast, the choice of ξ affects the projection Π and hence also Φr^*.

Note that there is a lot of flexibility for generating the sequence $\{i_0, i_1, \ldots\}$ and the transition sequence $\{(i_0, j_0), (i_1, j_1), \ldots\}$ to satisfy Eqs. (2.2) and (2.4). For example, to satisfy Eq. (2.2), the indices i_t do not need to be sampled independently according to ξ. Instead, it may be convenient to introduce an irreducible Markov chain having states $1, \ldots, n$ and ξ as its invariant distribution, and to start at some state i_0 and generate the sequence $\{i_0, i_1, \ldots\}$ as a single infinitely long trajectory of the chain. For the transition sequence, we may optionally let $j_k = i_{k+1}$ for all k, in which case P would be identical to the transition matrix of the selected Markov chain.

Let us discuss two possibilities for constructing a Markov chain with invariant distribution ξ. The first is useful when a desirable distribution ξ is known up to a normalization constant. Then, we can construct such a Markov chain starting with a proposal transition matrix (which matches the structure of A, for instance), by using the so-called detailed balance

[1] For a simplified analysis, note that the variance of each coefficient $\hat{p}_{ij,t}\frac{a_{ij}}{p_{ij}}$ appearing in Eq. (2.6) can be calculated to be

$$V_{ij,t} = \gamma_t p_{ij}(1 - p_{ij})\frac{a_{ij}^2}{p_{ij}^2} = \frac{\gamma_t a_{ij}^2}{p_{ij}} - \gamma_t a_{ij}^2,$$

where γ_t is the expected value of $1/\sum_{k=0}^{t} \delta(i_k = i)$, assuming the initial i_0 is distributed according to ξ. [To see this, note that $\hat{p}_{ij,t}$ is the average of Bernoulli random variables whose mean and variance are p_{ij} and $p_{ij}(1 - p_{ij})$, respectively, and whose number is the random variable $\sum_{k=0}^{t} \delta(i_k = i)$.] For a given i, let us consider the problem of finding $p_{ij}, j = 1, \ldots, n$, that minimize $\sum_{j=1}^{n} V_{ij,t}$ subject to the constraints $p_{ij} = 0$ if and only if $a_{ij} = 0$, and $\sum_{j=1}^{n} p_{ij} = 1$. By introducing a Lagrange multiplier ν for the constraint $\sum_{j=1}^{n} p_{ij} = 1$, and forming and minimizing the corresponding Lagrangian, we see that the optimal solution satisfies

$$\frac{a_{ij}^2}{p_{ij}^2} = \frac{\nu}{\gamma_t},$$

implying that p_{ij} should be chosen proportional to $|a_{ij}|$ (indeed this is standard practice in approximate DP, and is consistent with the principles of importance sampling [16]). This analysis, however, does not take into account the fact that the choice of p_{ij} may also affect ξ_i (as when ξ is the invariant distribution of the Markov chain associated with P), and through them the variance of both sides of Eq. (2.6). In order to optimize more meaningfully the choice of p_{ij}, this relation must be taken into account, as well as the dependence of the variance of the solution of Eq. (2.6) on other terms, such as the vectors $\phi(i)$ and b.

condition. (The Markov chains thus constructed are reversible.) This construction procedure is well known in Markov chain Monte Carlo (MCMC) methods and we refer to [16] for the details.

Another possibility, which is useful when there is no particularly desirable ξ, is to specify first the transition matrix of the Markov chain and let ξ be its invariant distribution. Then the requirement (2.2) will be satisfied if the Markov chain is irreducible, in which case ξ will be the unique invariant distribution of the chain and will have positive components. An important observation is that explicit knowledge of ξ is not required; it is just necessary to know the Markov chain and to be able to simulate its transitions. The approximate DP applications fall into this context (see the references given in Section 1). In Section 3, we will discuss favorable methods for constructing the Markov chain from A, which result in ΠT being a contraction so that Jacobi methods are applicable.

We finally note that multiple simulated sequences can be used to form the Eq. (2.5). For example, in the Markov-chain-based sampling schemes, we can generate multiple infinitely long trajectories of the chain, starting at several different states, and for each trajectory use $j_k = i_{k+1}$ for all k. This will work even if the chain has multiple recurrent classes, as long as there are no transient states and at least one trajectory is started from within each recurrent class. Again ξ will be an invariant distribution of the chain, and need not be known explicitly. Note that using multiple trajectories may be interesting even if there is a single recurrent class, for at least two reasons:

(a) The generation of trajectories may be parallelized among multiple processors, resulting in significant speedup.
(b) The empirical frequencies of occurrence of the states may approach the invariant probabilities more quickly; this is particularly so for large and "stiff" Markov chains.

3. Markov chain construction

In this section we will derive conditions for ΠT to be a contraction, so that the error bounds (1.5) and (1.6) apply, and the approximate Jacobi methods of the next section may also be used. Our results generalize corresponding results known for DP.

We consider the case where the index sequence $\{i_0, i_1, \ldots\}$ is generated as an infinitely long trajectory of a Markov chain whose invariant distribution is ξ. We denote by Q the corresponding transition probability matrix and by q_{ij} the components of Q. [In general, Q may not be the same as P, which is used to generate the transition sequence $\{(i_0, j_0), (i_1, j_1), \ldots\}$ to satisfy Eqs. (2.2) and (2.4).] It seems hard to guarantee that ΠT is a contraction mapping, unless $|A| \leq Q$. The following propositions assume this condition.

Proposition 1. *Assume that Q is irreducible and that $|A| \leq Q$. Then T and ΠT are contraction mappings under any one of the following three conditions:*

(1) *For some scalar $\alpha \in (0, 1)$, we have $|A| \leq \alpha Q$.*
(2) *There exists an index $\bar{i}$ such that $|a_{\bar{i}j}| < q_{\bar{i}j}$ for all $j = 1, \ldots, n$.*
(3) *There exists an index $\bar{i}$ such that $\sum_{j=1}^{n} |a_{\bar{i}j}| < 1$.*

Proof. Let ξ be the invariant distribution of Q. Assume condition (1). Since Π is nonexpansive with respect to $\|\cdot\|_\xi$, it will suffice to show that A is a contraction with respect to $\|\cdot\|_\xi$. We have

$$|Az| \leq |A|\,|z| \leq \alpha Q|z|, \quad \forall\, z \in \Re^n. \tag{3.1}$$

Using this relation, we obtain

$$\|Az\|_\xi \leq \alpha \|Q|z|\|_\xi \leq \alpha \|z\|_\xi, \quad \forall\, z \in \Re^n, \tag{3.2}$$

where the last inequality follows since $\|Qx\|_\xi \leq \|x\|_\xi$ for all $x \in \Re^n$ (see e.g., [24] or [7], Lemma 6.4). Thus, A is a contraction with respect to $\|\cdot\|_\xi$ with modulus α.

Assume condition (2). Then, in place of Eq. (3.1), we have

$$|Az| \leq |A|\,|z| \leq Q|z|, \quad \forall\, z \in \Re^n,$$

with strict inequality for the row corresponding to $\bar{i}$ when $z \neq 0$, and in place of Eq. (3.2), we obtain

$$\|Az\|_\xi < \|Q|z|\|_\xi \leq \|z\|_\xi, \quad \forall\, z \neq 0.$$

It follows that A is a contraction with respect to $\|\cdot\|_\xi$, with modulus $\max_{\|z\|_\xi \leq 1} \|Az\|_\xi$.

Assume condition (3). It will suffice to show that the eigenvalues of ΠA lie strictly within the unit circle.[2] Let $\bar{Q}$ be the matrix which is identical to Q except for the $\bar{i}$th row which is identical to the $\bar{i}$th row of $|A|$. From the irreducibility of Q, it

[2] In the following argument, the projection Πz of a complex vector z is obtained by separately projecting the real and the imaginary components of z on S. The projection norm for a complex vector $x + \mathrm{i}y$ is defined by

$$\|x + \mathrm{i}y\|_\xi = \sqrt{\|x\|_\xi^2 + \|y\|_\xi^2}.$$

follows that for any $i_1 \neq \bar{i}$ it is possible to find a sequence of nonzero components $\bar{Q}_{i_1 i_2}, \ldots, \bar{Q}_{i_{k-1} i_k}, \bar{Q}_{i_k \bar{i}}$ that "lead" from i_1 to $\bar{i}$. Using a well-known result, we have $\bar{Q}^t \to 0$. Since $|A| \leq \bar{Q}$, we also have $|A|^t \to 0$, and hence also $A^t \to 0$ (since $|A^t| \leq |A|^t$). Thus, all eigenvalues of A are strictly within the unit circle. We next observe that from the proof argument under conditions (1) and (2), we have

$$\|\Pi A z\|_\xi \leq \|z\|_\xi, \quad \forall\, z \in \Re^n,$$

so the eigenvalues of ΠA cannot lie outside the unit circle.

We now use an argument that has been used to prove Lemma 1 of [29]. Assume to arrive at a contradiction that ν is an eigenvalue of ΠA with $|\nu| = 1$, and let ζ be a corresponding eigenvector. We claim that $A\zeta$ must have both real and imaginary components in the subspace S. If this were not so, we would have $A\zeta \neq \Pi A\zeta$, so that

$$\|A\zeta\|_\xi > \|\Pi A\zeta\|_\xi = \|\nu\zeta\|_\xi = |\nu|\,\|\zeta\|_\xi = \|\zeta\|_\xi,$$

which contradicts the fact $\|Az\|_\xi \leq \|z\|_\xi$ for all z, shown earlier. Thus, the real and imaginary components of $A\zeta$ are in S, which implies that $A\zeta = \Pi A\zeta = \nu\zeta$, so that ν is an eigenvalue of A. This is a contradiction because $|\nu| = 1$, while the eigenvalues of A are strictly within the unit circle. □

Note that the preceding proof has shown that under conditions (1) and (2) of Proposition 1, T and ΠT are contraction mappings with respect to the specific norm $\|\cdot\|_\xi$, and that under condition (1), the modulus of contraction is α. Furthermore, Q need not be irreducible under these conditions – it is sufficient that Q has no transient states (so that it has an invariant distribution ξ with positive components). Under condition (3), T and ΠT need not be contractions with respect to $\|\cdot\|_\xi$. For a counterexample, take $a_{i,i+1} = 1$ for $i = 1, \ldots, n-1$, and $a_{n,1} = 1/2$, with every other entry of A equal to 0. Take also $q_{i,i+1} = 1$ for $i = 1, \ldots, n-1$, and $q_{n,1} = 1$, with every other entry of Q equal to 0, so $\xi_i = 1/n$ for all i. Then for $z = (0, 1, \ldots, 1)'$ we have $Az = (1, \ldots, 1, 0)'$ and $\|Az\|_\xi = \|z\|_\xi$, so A is not a contraction with respect to $\|\cdot\|_\xi$. Taking S to be the entire space $\Re^n$, we see that the same is true for ΠA.

When the row sums of $|A|$ are not greater than one, one can construct Q with $|A| \leq Q$ by adding another matrix to $|A|$:

$$Q = |A| + \mathrm{diag}(e - |A|e)R, \tag{3.3}$$

where R is a transition probability matrix, e is the unit vector that has all components equal to 1, and $\mathrm{diag}(e - |A|e)$ is the diagonal matrix with $1 - \sum_{m=1}^n |a_{im}|, i = 1, \ldots, n$, on the diagonal. Then the row sum deficit of the ith row of A is distributed to the columns j according to fractions r_{ij}, the components of R.

The next proposition uses different assumptions from Proposition 1, and applies to cases where there is no special index $\bar{i}$ such that $\sum_{j=1}^n |a_{ij}| < 1$. In fact A may itself be a transition probability matrix, so that $I - A$ need not be invertible, and the original system may have multiple solutions; see the subsequent Example 2 for the average cost DP case. The proposition suggests the use of a damped version of the T mapping in various methods, and is closely connected to a result on approximate DP methods for average cost problems ([29], Prop. 3).

Proposition 2. *Assume that there are no transient states corresponding to Q, that ξ is an invariant distribution of Q, and that $|A| \leq Q$. Assume further that $I - \Pi A$ is invertible. Then the mapping ΠT_γ, where*

$$T_\gamma = (1-\gamma)I + \gamma T,$$

is a contraction with respect to $\|\cdot\|_\xi$ for all $\gamma \in (0, 1)$.

Proof. The argument of the proof of Proposition 1 shows that the condition $|A| \leq Q$ implies that A is nonexpansive with respect to the norm $\|\cdot\|_\xi$. Furthermore, since $I - \Pi A$ is invertible, we have $z \neq \Pi Az$ for all $z \neq 0$. Hence for all $\gamma \in (0, 1)$,

$$\|(1-\gamma)z + \gamma\Pi Az\|_\xi < (1-\gamma)\|z\|_\xi + \gamma\|\Pi Az\|_\xi \leq (1-\gamma)\|z\|_\xi + \gamma\|z\|_\xi = \|z\|_\xi, \quad \forall\, z \in \Re^n, \tag{3.4}$$

where the strict inequality follows from the strict convexity of the norm, and the weak inequality follows from the nonexpansiveness of ΠA. If we define

$$\rho_\gamma = \sup\left\{\|(1-\gamma)z + \gamma\Pi Az\|_\xi \mid \|z\| \leq 1\right\},$$

and note that the supremum above is attained by Weierstrass' Theorem, we see that Eq. (3.4) yields $\rho_\gamma < 1$ and

$$\|(1-\gamma)z + \gamma\Pi Az\|_\xi \leq \rho_\gamma\|z\|_\xi, \quad \forall\, z \in \Re^n.$$

From the definition of T_γ, we have for all $x, y \in \Re^n$,

$$\Pi T_\gamma x - \Pi T_\gamma y = \Pi T_\gamma(x-y) = (1-\gamma)\Pi(x-y) + \gamma\Pi A(x-y) = (1-\gamma)\Pi(x-y) + \gamma\Pi(\Pi A(x-y)),$$

so defining $z = x - y$, and using the preceding two relations and the nonexpansiveness of Π, we obtain

$$\|\Pi T_\gamma x - \Pi T_\gamma y\|_\xi = \|(1-\gamma)\Pi z + \gamma\Pi(\Pi Az)\|_\xi \leq \|(1-\gamma)z + \gamma\Pi Az\|_\xi \leq \rho_\gamma\|z\|_\xi = \rho_\gamma\|x-y\|_\xi,$$

for all $x, y \in \Re^n$. □

Note that the mappings ΠT_γ and ΠT have the same fixed points, so under the assumptions of Proposition 2, there is a unique fixed point Φr^* of ΠT. However, if T has a nonempty linear manifold of fixed points, there arises the question of how close Φr^* is to this manifold. It may be possible to address this issue in specialized contexts; in particular, it has been addressed in [25] in the context of average cost DP problems (cf. the subsequent Example 2).

We now discuss examples of choices of ξ and Q in some interesting special cases.

Example 1 (*Discounted Markovian Decision Problems*). As mentioned in Section 1, Bellman's equation for the cost vector of a stationary policy in an n-state discounted Markovian decision problem has the form $x = T(x)$, where

$$T(x) = \alpha P x + g,$$

g is the vector of single-stage costs associated with the n states, P is the transition probability matrix of the associated Markov chain, and $\alpha \in (0, 1)$ is the discount factor. If P is an irreducible Markov chain, and ξ is chosen to be its unique invariant distribution, the equation approximation method based on Eq. (2.5) yields a popular policy evaluation method known as LSTD(0) (see the references given in Section 1). Furthermore, since condition (1) of Proposition 1 is satisfied, it follows that ΠT is a contraction with respect to $\|\cdot\|_\xi$, the error bound (1.6) holds, and the Jacobi/fixed point method (1.7) applies. These results are well known in the approximate DP literature (see the references given in Section 1).

The methodology of the present paper also allows the use of simulation using a Markov chain Q other than P, with an attendant change in ξ (in the DP context there is motivation for doing so in cases where P is not irreducible or some states have very small steady-state probabilities; this is the issue of "exploration" discussed for example in [7,22]). In particular, a sequence of states $\{i_0, i_1, \ldots\}$ may be generated according to an irreducible transition probability matrix of the form

$$Q = (I - B)P + BR,$$

where B is a diagonal matrix with diagonal components $\beta_i \in [0, 1]$ and R is another transition probability matrix (possibly one corresponding to a different policy). Thus, at state i, the next state is generated according to p_{ij} with probability $1 - \beta_i$, and according to r_{ij} with probability β_i, which may be viewed as an *exploration probability* at state i [we are exploring other states, which might not be visited as frequently (or at all) under P]. Thus we solve $x = \Pi(\alpha P x + g)$ with the weights ξ in Π being the invariant distribution of Q. If $\beta_i < 1 - \alpha$ for all i, then $\alpha P \leq \overline{\alpha} Q$ for some $\overline{\alpha} < 1$, so by Proposition 1, ΠT will still be a contraction with respect to $\|\cdot\|_\xi$ (actually with a refinement of the proof of Proposition 1, it can be shown that ΠT is a contraction if $\beta_i < 1 - \alpha^2$). For other values of β_i, ΠT may not be a contraction, but the equation approximation approach of Section 2 still applies. [The multistep LSTD(λ) methodology of the subsequent Section 5 also applies. In fact, the contraction modulus of the multistep mapping $T^{(\lambda)}$ of Eq. (1.10) approaches 0 as $\lambda \to 1$ (see the subsequent Proposition 3), so $\Pi T^{(\lambda)}$ is a contraction for any values $\beta_i < 1$, provided λ is sufficiently close to 1.]

The corresponding simulation-based algorithms can take a number of forms. After generating a sequence of states $\{i_0, i_1, \ldots\}$ according to Q, we construct and then solve the equation

$$\sum_{k=0}^{t} \phi(i_k) \left(\phi(i_k) - \alpha \frac{p_{i_k i_{k+1}}}{q_{i_k i_{k+1}}} \phi(i_{k+1}) \right)' r = \sum_{k=0}^{t} \phi(i_k) g_{i_k} \tag{3.5}$$

[cf. Eq. (2.5) with the identifications $j_k = i_{k+1}$, $a_{i_k j_k} = \alpha p_{i_k i_{k+1}}$ and $p_{i_k j_k} = q_{i_k i_{k+1}}$] to obtain an approximation $\hat{r}_t$ of r^*. When g_i itself depends on P, e.g., when, as often in DP, g_i is the expected cost at state i, $g_i = \sum_{j=1}^{n} p_{ij} g(i, j)$, where $g(i, j)$ is the cost of a transition (i, j), we use the following extension of Eq. (3.5),

$$\sum_{k=0}^{t} \phi(i_k) \left(\phi(i_k) - \alpha \frac{p_{i_k i_{k+1}}}{q_{i_k i_{k+1}}} \phi(i_{k+1}) \right)' r = \sum_{k=0}^{t} \frac{p_{i_k i_{k+1}}}{q_{i_k i_{k+1}}} \phi(i_k) g(i_k, i_{k+1}).$$

The ratios $\frac{p_{i_k i_{k+1}}}{q_{i_k i_{k+1}}}$ can be calculated, when P and Q are known explicitly (as in queueing applications, for instance), or, even when the latter are not known explicitly, as in Q-factor learning where the state space of the Markov chains associated with P and Q actually corresponds to the joint state-action space of the problem (see the approximate DP literature). As an alternative to the above sampling scheme, in a simulation context, it is also straightforward to generate another sequence of transitions $\{(i_0, j_0), (i_1, j_1), \ldots\}$ according to P, in addition to and independently of the sequence $\{i_0, i_1, \ldots\}$, which is generated according to Q. Then, in place of Eq. (3.5), we may form and solve the equation

$$\sum_{k=0}^{t} \phi(i_k)(\phi(i_k) - \alpha \phi(j_k))' r = \sum_{k=0}^{t} \phi(i_k) g(i_k, j_k), \tag{3.6}$$

as described in Eqs. (2.2)–(2.5).

The idea of using two different Markov chains within approximate DP methods to introduce exploration is well known (see e.g., [22]). However, the methods based on Eqs. (3.5) and (3.6), as well as their LSPE(λ) analogs, are new to our knowledge, and have the guaranteed convergence property $\hat{r}_t \to r^*$.

Example 2 (*Undiscounted Markovian Decision Problems*). Consider the equation $x = Ax + b$, for the case where A is a substochastic matrix ($a_{ij} \geq 0$ for all i, j and $\sum_{j=1}^{n} a_{ij} \leq 1$ for all i). Here $1 - \sum_{j=1}^{n} a_{ij}$ may be viewed as a transition probability from state i to some absorbing state denoted 0. This is related to Bellman's equation for the cost vector of a stationary policy of a DP problem of the stochastic shortest path type (see e.g., [6]). If the policy is proper in the sense that from any state $i \neq 0$ there exists a path of positive probability transitions from i to the absorbing state 0, the matrix

$$Q = |A| + \operatorname{diag}(e - |A|e)R$$

[cf. Eq. (3.3)] is irreducible, provided R has positive components. As a result, the conditions of Proposition 1 under condition (2) are satisfied, and T and ΠT are contractions with respect to $\|\cdot\|_\xi$. It is also possible to use a matrix R whose components are not all positive, as long as Q is irreducible, in which case Proposition 1 under condition (3) applies.

Consider also the equation $x = Ax + b$ for the case where A is an irreducible transition probability matrix, with invariant distribution ξ. This is related to Bellman's equation for the differential cost vector of a stationary policy of an average cost DP problem involving a Markov chain with transition probability matrix A. Then, if the unit vector e is not contained in the subspace S spanned by the basis functions, it can be shown that the matrix $I - \Pi A$ is invertible (see [25], which also gives a related error bound). As a result, Proposition 2 applies and shows that the mapping $(1 - \gamma)I + \gamma A$, is a contraction with respect to $\|\cdot\|_\xi$ for all $\gamma \in (0, 1)$. The corresponding equation approximation approach and approximate Jacobi method are discussed in [29].

Example 3 (*Weakly Diagonally Dominant Systems*). Consider the solution of the system

$$Cx = d,$$

where $d \in \Re^n$ and C is an $n \times n$ matrix that is weakly diagonally dominant, i.e., its components satisfy

$$c_{ii} \neq 0, \qquad \sum_{j \neq i} |c_{ij}| \leq |c_{ii}|, \quad i = 1, \ldots, n. \tag{3.7}$$

By dividing the ith row by c_{ii}, we obtain the equivalent system $x = Ax + b$, where the components of A and b are

$$a_{ij} = \begin{cases} 0 & \text{if } i = j, \\ -\dfrac{c_{ij}}{c_{ii}} & \text{if } i \neq j, \end{cases} \qquad b_i = \frac{d_i}{c_{ii}}, \quad i = 1, \ldots, n.$$

Then, from Eq. (3.7), we have

$$\sum_{j=1}^{n} |a_{ij}| = \sum_{j \neq i} \frac{|c_{ij}|}{|c_{ii}|} \leq 1, \quad i = 1, \ldots, n,$$

so Propositions 1 and 2 may be used under the appropriate conditions. In particular, if the matrix Q given by Eq. (3.3) has no transient states and there exists an index $\bar{i}$ such that $\sum_{j=1}^{n} |a_{\bar{i}j}| < 1$, Proposition 1 applies and shows that ΠT is a contraction.

Alternatively, instead of Eq. (3.7), assume the somewhat more restrictive condition

$$|1 - c_{ii}| + \sum_{j \neq i} |c_{ij}| \leq 1, \quad i = 1, \ldots, n, \tag{3.8}$$

and consider the equivalent system $x = Ax + b$, where

$$A = I - C, \qquad b = d.$$

Then, from Eq. (3.8), we have

$$\sum_{j=1}^{n} |a_{ij}| = |1 - c_{ii}| + \sum_{j \neq i} |c_{ij}| \leq 1, \quad i = 1, \ldots, n,$$

so again Propositions 1 and 2 apply under appropriate conditions.

Example 4 (*Discretized Poisson's Equation*). Diagonally dominant linear systems arise in many contexts, including discretized partial differential equations, finite element methods, and economics applications. As an example, consider a discretized version of Poisson's equation over a two-dimensional square grid of N^2 points with fixed boundary conditions, which has the form

$$x_{i,j} = \frac{1}{4}(x_{i+1,j} + x_{i-1,j} + x_{i,j+1} + x_{i,j-1}) + g_{i,j}, \quad i, j = 1, \ldots, N,$$

where $g_{i,j}$ are given scalars, and by convention $x_{N+1,j} = x_{0,j} = x_{i,N+1} = x_{i,0} = 0$. A subset of the points (i, j) in the square grid are "boundary points", where $x_{i,j}$ is fixed and given. The problem is to compute the values $x_{i,j}$ at the remaining points, which

are referred to as "interior points". Thus, we have one equation for each interior grid point. Clearly, this is a special case of Examples 2 and 3, with the row components of A corresponding to (i, j) being 1/4 for each neighboring interior point of (i, j), and 0 otherwise. If from any interior point it is possible to arrive at some boundary point through a path of adjacent interior points, then clearly based on the graph-geometric structure of the problem, one can construct an irreducible Q satisfying $|A| \leq Q$.

Let us finally address the question whether it is possible to find Q such that $|A| \leq Q$ and the corresponding Markov chain has no transient states or is irreducible. To this end, assume that $\sum_{j=1}^{n} |a_{ij}| \leq 1$ for all i. If A is itself irreducible, then any Q such that $|A| \leq Q$ is also irreducible. Otherwise, consider the set

$$\bar{I} = \left\{ i \,\middle|\, \sum_{j=1}^{n} |a_{ij}| < 1 \right\},$$

and assume that it is nonempty (otherwise the only possibility is $Q = |A|$). Let $\tilde{I}$ be the set of i such that there exists a sequence of nonzero components $a_{ij_1}, a_{j_1 j_2}, \ldots, a_{j_m \bar{i}}$ such that $\bar{i} \in \bar{I}$, and let $\hat{I} = \{i \mid i \notin \bar{I} \cup \tilde{I}\}$ (we allow here the possibility that $\tilde{I}$ or $\hat{I}$ may be empty). Note that the square submatrix of $|A|$ corresponding to $\hat{I}$ is a transition probability matrix, and that we have $a_{ij} = 0$ for all $i \in \hat{I}$ and $j \notin \hat{I}$. Then it can be shown that there exists Q with $|A| \leq Q$ and no transient states if and only if the Markov chain corresponding to $\hat{I}$ has no transient states. Furthermore, there exists an irreducible Q with $|A| \leq Q$ if and only if $\hat{I}$ is empty. We refer to the extended report [5] for further discussion and methods to construct Q.

4. Approximate Jacobi methods

We will now focus on the iteration

$$\Phi r_{t+1} = \Pi T(\Phi r_t), \quad t = 0, 1, \ldots, \tag{4.1}$$

[cf. Eq. (1.7)], which we refer to as the *projected Jacobi* method (PJ for short). We assume throughout this section that *ΠT is a contraction with respect to some norm*, and note that Propositions 1 and 2 provide tools for verifying that this is so. A more general iteration involves a stepsize $\gamma \in (0, 1]$,

$$\Phi r_{t+1} = (1 - \gamma)\Phi r_t + \gamma \Pi T(\Phi r_t), \quad t = 0, 1, \ldots,$$

and has similar algorithmic properties, but for simplicity, we restrict attention to the case $\gamma = 1$. Our simulation-based approximation to the PJ iteration descends from the LSPE methods of approximate DP, referred to in Section 1.

By expressing the projection as a least squares minimization, we can write the PJ iteration (4.1) as

$$r_{t+1} = \arg\min_{r \in \Re^s} \|\Phi r - T(\Phi r_t)\|_\xi^2,$$

or equivalently

$$r_{t+1} = \arg\min_{r \in \Re^s} \sum_{i=1}^{n} \xi_i \left(\phi(i)' r - \sum_{j=1}^{n} a_{ij} \phi(j)' r_t - b_i \right)^2. \tag{4.2}$$

By setting the gradient of the cost function above to 0 and using a straightforward calculation, we have

$$r_{t+1} = \left(\sum_{i=1}^{n} \xi_i \phi(i) \phi(i)' \right)^{-1} \sum_{i=1}^{n} \xi_i \phi(i) \left(\sum_{j=1}^{n} a_{ij} \phi(j)' r_t + b_i \right). \tag{4.3}$$

Similarly to the equation approximation methods of Section 2, we observe that this iteration involves two expected values with respect to the distribution ξ, and we approximate these expected values by sample averages. Thus, we approximate iteration (4.3) with

$$r_{t+1} = \left(\sum_{k=0}^{t} \phi(i_k) \phi(i_k)' \right)^{-1} \sum_{k=0}^{t} \phi(i_k) \left(\frac{a_{i_k j_k}}{p_{i_k j_k}} \phi(j_k)' r_t + b_{i_k} \right), \tag{4.4}$$

which we refer to as the *approximate projected Jacobi* method (APJ for short). Here again $\{i_0, i_1, \ldots\}$ is an index sequence and $\{(i_0, j_0), (i_1, j_1), \ldots\}$ is a transition sequence satisfying Eqs. (2.2)–(2.4).

Similarly to Section 2, we write Eq. (4.4) as

$$r_{t+1} = \left(\sum_{i=1}^{n} \hat{\xi}_{i,t}\, \phi(i) \phi(i)' \right)^{-1} \sum_{i=1}^{n} \hat{\xi}_{i,t} \phi(i) \left(\sum_{j=1}^{n} \hat{p}_{ij,t} \frac{a_{ij}}{p_{ij}} \phi(j)' r_t + b_i \right), \tag{4.5}$$

where $\hat{\xi}_{i,t}$ and $\hat{p}_{ij,t}$ are defined by

$$\hat{\xi}_{i,t} = \frac{\sum_{k=0}^{t} \delta(i_k = i)}{t+1}, \qquad \hat{p}_{ij,t} = \frac{\sum_{k=0}^{t} \delta(i_k = i, j_k = j)}{\sum_{k=0}^{t} \delta(i_k = i)}.$$

We then note that by Eqs. (2.2) and (2.4), $\hat{\xi}_{i,t}$ and $\hat{p}_{ij,t}$ converge (with probability 1) to ξ_i and p_{ij}, respectively, so by comparing Eqs. (4.3) and (4.5), we see that they asymptotically coincide. Since Eq. (4.3) is a contracting fixed point iteration that converges to r^*, it follows with a simple argument that the same is true for iteration (4.5) (with probability 1).

To streamline and efficiently implement the APJ iteration (4.4), we introduce the matrices

$$B_t = \sum_{k=0}^{t} \phi(i_k)\phi(i_k)', \qquad C_t = \sum_{k=0}^{t} \phi(i_k) \left(\frac{a_{i_k j_k}}{p_{i_k j_k}} \phi(j_k) - \phi(i_k) \right)',$$

and the vector

$$d_t = \sum_{k=0}^{t} \phi(i_k) b_{i_k}.$$

We then write Eq. (4.4) compactly as

$$r_{t+1} = r_t + B_t^{-1}(C_t r_t + d_t), \tag{4.6}$$

and also note that B_t, C_t, and d_t can be efficiently updated using the formulas

$$B_t = B_{t-1} + \phi(i_t)\phi(i_t)', \qquad C_t = C_{t-1} + \phi(i_t) \left(\frac{a_{i_t j_t}}{p_{i_t j_t}} \phi(j_t) - \phi(i_t) \right)', \tag{4.7}$$

$$d_t = d_{t-1} + \phi(i_t) b_{i_t}. \tag{4.8}$$

Let us also observe that Eq. (2.5), the first equation approximation method of Section 2, can be written compactly as

$$C_t r + d_t = 0. \tag{4.9}$$

We can use this formula to establish a connection between the equation approximation and APJ approaches. In particular, suppose that we truncate the state and transition sequences after t transitions, but continue the APJ iteration with B_t, C_t, and d_t held fixed, i.e., consider the iteration

$$r_{m+1} = r_m + B_t^{-1}(C_t r_m + d_t), \quad m = t, t+1, \ldots. \tag{4.10}$$

Then, since APJ approximates PJ and ΠT is assumed to be a contraction, it follows that with probability 1, for sufficiently large t, the matrix $I + B_t^{-1} C_t$ will be a contraction and iteration (4.10) will converge, by necessity to the solution $\hat{r}_t$ of Eq. (4.9). The conclusion is that, for large t, *the APJ iteration (4.6) can be viewed as a single/first iteration of the algorithm (4.10) that solves the approximate projected equation (4.9)*.

Another issue of interest is the rate of convergence of the difference $r_t - \hat{r}_t$ of the results of the two approaches. Within the DP context and under some natural assumptions, $r_t - \hat{r}_t$ converges to 0 faster, in a certain probabilistic sense, than the error differences $r_t - r^*$ and $\hat{r}_t - r^*$ (see [2,29]). Within the more general context of the present paper, a similar analysis is possible, but is outside our scope.

5. Multistep versions

We now consider multistep versions that replace T with another mapping that has the same fixed points, such as T^l with $l > 1$, or $T^{(\lambda)}$ given by

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

where $\lambda \in (0, 1)$ is such that the preceding infinite series is convergent, i.e., λA must have eigenvalues strictly within the unit circle. This is seldom considered in traditional fixed point methods, because either the gain in rate of convergence is offset by increased overhead per iteration, or the implementation becomes cumbersome, or both. However, in the context of our simulation-based methods, this replacement is possible, and in fact has a long history in approximate DP, as mentioned in Section 1.

As motivation, note that if T is a contraction, the modulus of contraction may be enhanced through the use of T^l or $T^{(\lambda)}$. In particular, if $\alpha \in [0, 1)$ is the modulus of contraction of T, the modulus of contraction of T^l is α^l, while the modulus of contraction of $T^{(\lambda)}$ is

$$\alpha^{(\lambda)} = (1-\lambda)\sum_{k=0}^{\infty}\lambda^l\alpha^{l+1} = \frac{\alpha(1-\lambda)}{1-\alpha\lambda} < \alpha.$$

Thus the error bounds (1.5) or (1.6) are enhanced. Moreover, there are circumstances where $T^{(\lambda)}$ is a contraction, while T is not, as we will demonstrate shortly (see the following Proposition 3).

To gain some understanding into the properties of $T^{(\lambda)}$, let us write it as

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

where from the equations $T^{l+1}(x) = A^{l+1}x + \sum_{m=0}^{l} A^m b$ and $T^{(\lambda)} = (1-\lambda)\sum_{l=0}^{\infty}\lambda^l T^{l+1}$, we have

$$A^{(\lambda)} = (1-\lambda)\sum_{l=0}^{\infty}\lambda^l A^{l+1}, \qquad b^{(\lambda)} = (1-\lambda)\sum_{l=0}^{\infty}\lambda^l\sum_{m=0}^{l}A^m b = (1-\lambda)\sum_{l=0}^{\infty}A^l b\sum_{m=l}^{\infty}\lambda^m = \sum_{l=0}^{\infty}\lambda^l A^l b. \tag{5.1}$$

The following proposition provides some properties of $A^{(\lambda)}$, which in turn determine contraction and other properties of $T^{(\lambda)}$. We denote by $a_{ij}^{(\lambda)}$ the components of $A^{(\lambda)}$, and by $\sigma(A)$ and $\sigma(A^{(\lambda)})$ the spectral radius of A and $A^{(\lambda)}$, respectively. Note that $\sigma(M) \le \|M\|$, where for any $n\times n$ matrix M and norm $\|\cdot\|$ of $\Re^n$, we denote by $\|M\|$ the corresponding matrix norm of M: $\|M\| = \max_{\|z\|\le 1}\|Mz\|$. Note also that for a transition probability matrix Q, having as an invariant distribution a vector ξ with positive components, we have $\sigma(Q) = \|Q\|_\xi = 1$ (see e.g., [7], Lemma 6.4).

Proposition 3. *Let $I - A$ be invertible and $\sigma(A) \le 1$.*

(a) *We have $\sigma(A^{(\lambda)}) < 1$ for all $\lambda \in (0,1)$. Furthermore, $\lim_{\lambda\to 1}\sigma(A^{(\lambda)}) = 0$.*

(b) *Assume further that $|A| \le Q$, where Q is a transition probability matrix having as invariant distribution a vector ξ with positive components. Then,*

$$|A^{(\lambda)}| \le Q^{(\lambda)}, \qquad \|A^{(\lambda)}\|_\xi \le \|Q^{(\lambda)}\|_\xi = 1, \quad \forall\,\lambda\in[0,1),$$

where $Q^{(\lambda)} = (1-\lambda)\sum_{l=0}^{\infty}\lambda^l Q^{l+1}$. Furthermore, for all $\lambda\in(0,1)$ the eigenvalues of $\Pi A^{(\lambda)}$ lie strictly within the unit circle, where Π denotes projection on S with respect to $\|\cdot\|_\xi$.

Proof. (a) From Eq. (5.1), we see that the eigenvalues of $A^{(\lambda)}$ have the form

$$(1-\lambda)\sum_{l=0}^{\infty}\lambda^l\beta^{l+1} = \frac{\beta(1-\lambda)}{1-\beta\lambda}, \tag{5.2}$$

where β is an eigenvalue of A. Since $|\beta| \le 1$ and $\beta \ne 1$, there exist integers i and j such that $\beta^i \ne \beta^j$, so a convex combination of β^i and β^j lies strictly within the unit circle, and the same is true for the eigenvalues $(1-\lambda)\sum_{l=0}^{\infty}\lambda^l\beta^{l+1}$ of $A^{(\lambda)}$. It follows that $\sigma(A^{(\lambda)}) < 1$. It is also evident from Eq. (5.2) that $\lim_{\lambda\to 1}\sigma(A^{(\lambda)}) = 0$.

(b) To see that $|A^{(\lambda)}| \le Q^{(\lambda)}$, note that for all $l > 1$, the components of $|A|^l$ are not greater than the corresponding components of Q^l, since they can be written as products of corresponding components of $|A|$ and Q, and by assumption, we have $|a_{ij}| \le q_{ij}$ for all $i, j = 1,\ldots,n$. We have $\|Q^{(\lambda)}\|_\xi = 1$ because $Q^{(\lambda)}$ is a transition probability matrix and ξ is an invariant distribution of $Q^{(\lambda)}$. The inequality $\|A^{(\lambda)}\|_\xi \le \|Q^{(\lambda)}\|_\xi$ follows by a simple modification of the proof of Proposition 1.

Since $\|A^{(\lambda)}\|_\xi \le \|Q^{(\lambda)}\|_\xi = 1$ and Π is nonexpansive with respect to $\|\cdot\|_\xi$, it follows that $\|\Pi A^{(\lambda)}\|_\xi \le 1$, so all eigenvalues of $\Pi A^{(\lambda)}$ lie within the unit circle. Furthermore, by Lemma 1 of [29], all eigenvalues ν of $\Pi A^{(\lambda)}$ with $|\nu| = 1$ must also be eigenvalues of $A^{(\lambda)}$. Since by part (a) we have $\sigma(A^{(\lambda)}) < 1$ for $\lambda > 0$, there are no such eigenvalues. □

Note from Proposition 3(a) that $T^{(\lambda)}$ is a contraction for $\lambda > 0$ even if T is not, provided that $I - A$ is invertible and $\sigma(A) \le 1$. This is not true for T^l, $l > 1$, since the eigenvalues of A^l are β^l where β is an eigenvalue of A, so that $\sigma(A^l) = 1$ if $\sigma(A) = 1$. Furthermore, under the assumptions of Proposition 3(b), $\Pi T^{(\lambda)}$ is a contraction for $\lambda > 0$. This suggests an advantage for using $\lambda > 0$, and the fact $\lim_{\lambda\to 1}\sigma(A^{(\lambda)}) = 0$ also suggests an advantage for using λ close to 1. However, as we will discuss later, there is also a disadvantage in our simulation-based methods for using λ close to 1, because of increased simulation noise.

The key idea of the subsequent simulation-based algorithms is that the ith component $(A^m g)(i)$ of a vector of the form $A^m g$, where $g\in\Re^n$, can be computed by averaging over properly weighted simulation-based sample values, just as $(Ag)(i)$ for $m = 1$. In particular, we generate the index sequence $\{i_0, i_1, \ldots\}$ and the transition sequence $\{(i_0,i_1),(i_1,i_2),\ldots\}$ by using the same irreducible transition matrix P, and we form the average of $w_{k,m}g_{i_{k+m}}$ over all indices k such that $i_k = i$, where

$$w_{k,m} = \begin{cases} \dfrac{a_{i_k i_{k+1}}}{p_{i_k i_{k+1}}}\dfrac{a_{i_{k+1}i_{k+2}}}{p_{i_{k+1}i_{k+2}}}\cdots\dfrac{a_{i_{k+m-1}i_{k+m}}}{p_{i_{k+m-1}i_{k+m}}} & \text{if } m \ge 1,\\ 1 & \text{if } m = 0. \end{cases} \tag{5.3}$$

In short

$$(A^m g)(i) \approx \frac{\sum_{k=0}^{t} \delta(i_k = i) w_{k,m} g_{i_{k+m}}}{\sum_{k=0}^{t} \delta(i_k = i)}. \tag{5.4}$$

The justification is that, by the ergodicity of the associated Markov chain, we have

$$\lim_{t\to\infty} \frac{\sum_{k=0}^{t} \delta(i_k = i, i_{k+1} = j_1, \ldots, i_{k+m} = j_m)}{\sum_{k=0}^{t} \delta(i_k = i)} = p_{ij_1} p_{j_1 j_2} \cdots p_{j_{m-1} j_m}, \tag{5.5}$$

and the limit of the right-hand side of Eq. (5.4) can be written as

$$\begin{aligned}
\lim_{t\to\infty} \frac{\sum_{k=0}^{t} \delta(i_k = i) w_{k,m} g_{i_{k+m}}}{\sum_{k=0}^{t} \delta(i_k = i)} &= \lim_{t\to\infty} \frac{\sum_{k=0}^{t} \sum_{j_1=1}^{n} \cdots \sum_{j_m=1}^{n} \delta(i_k = i, i_{k+1} = j_1, \ldots, i_{k+m} = j_m) w_{k,m} g_{i_{k+m}}}{\sum_{k=0}^{t} \delta(i_k = i)} \\
&= \sum_{j_1=1}^{n} \cdots \sum_{j_m=1}^{n} \lim_{t\to\infty} \frac{\sum_{k=0}^{t} \delta(i_k = i, i_{k+1} = j_1, \ldots, i_{k+m} = j_m)}{\sum_{k=0}^{t} \delta(i_k = i)} w_{k,m} g_{i_{k+m}} \\
&= \sum_{j_1=1}^{n} \cdots \sum_{j_m=1}^{n} a_{ij_1} a_{j_1 j_2} \cdots a_{j_{m-1} j_m} g_{j_m} \\
&= (A^m g)(i),
\end{aligned}$$

where the third equality follows using Eqs. (5.3) and (5.5). By using the approximation formula (5.4), it is possible to construct complex simulation-based approximations to formulas that involve powers of A. The subsequent multistep methods in this section and the basis construction methods of Section 6 rely on this idea.

5.1. l-step methods

Let us now develop simulation methods based on the equation $\Phi r = \Pi T^l(\Phi r)$, corresponding to T^l with $l > 1$. An advantage of these methods is that they do not require any assumption on the spectral radius of A. By contrast, the subsequent λ-methods require that $\lambda\sigma(A) < 1$ in order for $T^{(\lambda)}$ to be defined. Consider the projected Jacobi iteration

$$\Phi r_{t+1} = \Pi T^l(\Phi r_t) = \Pi \left(A^l \Phi r_t + \sum_{m=0}^{l-1} A^m b \right).$$

Equivalently,

$$\begin{aligned}
r_{t+1} &= \arg\min_{r\in\Re^s} \sum_{i=1}^{n} \xi_i \left(\phi(i)' r - (T^l(\Phi r_t))(i) \right)^2 \\
&= \arg\min_{r\in\Re^s} \sum_{i=1}^{n} \xi_i \left(\phi(i)' r - (A^l \Phi)(i) r_t - \left(\sum_{m=0}^{l-1} A^m b \right)(i) \right)^2,
\end{aligned}$$

where $(A^l\Phi)(i)$ is the ith row of the matrix $A^l\Phi$, and $(T^l(\Phi r_t))(i)$ and $\left(\sum_{m=0}^{l-1} A^m b\right)(i)$ are the ith components of the vectors $T^l(\Phi r_t)$ and $\sum_{m=0}^{l-1} A^m b$, respectively. By solving for the minimum over r, we finally obtain

$$r_{t+1} = \left(\sum_{i=1}^{n} \xi_i \phi(i) \phi(i)' \right)^{-1} \sum_{i=1}^{n} \xi_i \phi(i) \left((A^l \Phi)(i) r_t + \left(\sum_{m=0}^{l-1} A^m b \right)(i) \right).$$

We propose the following approximation to this iteration:

$$r_{t+1} = \left(\sum_{k=0}^{t} \phi(i_k) \phi(i_k)' \right)^{-1} \sum_{k=0}^{t} \phi(i_k) \left(w_{k,l} \phi(i_{k+l})' r_t + \sum_{m=0}^{l-1} w_{k,m} b_{i_{k+m}} \right) \tag{5.6}$$

where $w_{k,m}$ is given by Eq. (5.3). Its validity is based on Eq. (5.4), and on the fact that each index i is sampled with probability ξ_i, and each transition (i, j) is generated with probability p_{ij}, as part of the infinitely long sequence of states $\{i_0, i_1, \ldots\}$ of a Markov chain whose invariant distribution is ξ and transition probabilities are p_{ij}.

As in Section 4, we can express the l-step APJ iteration (5.6) in compact form. In particular, we can write Eq. (5.6) as

$$r_{t+1} = r_t + B_t^{-1}(C_t r_t + h_t), \tag{5.7}$$

where

$$C_t = \sum_{k=0}^{t} \phi(i_k)(w_{k,l}\phi(i_{k+l}) - \phi(i_k))', \qquad B_t = \sum_{k=0}^{t} \phi(i_k)\phi(i_k)', \qquad h_t = \sum_{k=0}^{t} \phi(i_k) \sum_{m=0}^{l-1} w_{k,m} b_{i_{k+m}}.$$

Note that to calculate C_t and h_t, it is necessary to generate the future l states $i_{t+1}, \ldots, i_{t+l}$. Note also that C_t, B_t, and h_t can be efficiently updated via

$$C_t = C_{t-1} + \phi(t_t)(w_{t,l}\phi(i_{t+l}) - \phi(i_t))', \qquad B_t = B_{t-1} + \phi(i_t)\phi(i_t)', \qquad h_t = h_{t-1} + \phi(i_t)z_t,$$

where z_t is given by

$$z_t = \sum_{m=0}^{l-1} w_{t,m} b_{i_{t+m}},$$

and can be updated by

$$z_t = \frac{z_{t-1} - b_{i_{t-1}}}{w_{t-1,1}} + w_{t,l-1} b_{i_{t+l-1}}.$$

An important observation is that compared to the case $l = 1$ [cf. Eqs. (4.4) and (2.1)], the term $w_{k,l}$ multiplying $\phi(i_{k+l})'$ and the terms $w_{k,m}$ multiplying $b_{i_{k+m}}$ in Eq. (5.6) tend to increase the variance of the samples used in the approximations as l increases. This is a generic tradeoff in the multistep methods of this section: by using equations involving greater dependence on more distant steps (larger values of l or λ) we improve the modulus of contraction, but we degrade the quality of the simulation through greater variance of the associated samples.

The preceding analysis also yields an equation approximation method corresponding to the APJ iteration (5.7). It has the form $C_t r + h_t = 0$, and we have $\hat{r}_t \to r^*$ with probability 1, where $\hat{r}_t$ is a solution to this equation.

5.2. λ-methods

We will now develop simulation methods based on the equation $\Phi r = \Pi T^{(\lambda)}(\Phi r)$. We will express these methods using convenient recursive formulas that use temporal differences; these are residual-like terms of the form $A^m(b + Ax - x)$, $m \geq 0$, which are used widely in approximate DP algorithms (see the references given in Section 1). However, we note that there are several alternative recursive formulas, of nearly equal effectiveness, which do not involve temporal differences. We first express $T^{(\lambda)}$ in terms of temporal difference-like terms[3]:

$$T^{(\lambda)}(x) = x + \sum_{m=0}^{\infty} \lambda^m (A^m b + A^{m+1}x - A^m x).$$

Using the above expression, we write the projected Jacobi iteration as

$$\Phi r_{t+1} = \Pi T^{(\lambda)}(\Phi r_t) = \Pi \left(\Phi r_t + \sum_{m=0}^{\infty} \lambda^m (A^m b + A^{m+1}\Phi r_t - A^m \Phi r_t) \right),$$

[3] This can be seen from the following calculation [cf. Eq. (5.1)]:

$$\begin{aligned} T^{(\lambda)}(x) &= \sum_{l=0}^{\infty} (1-\lambda)\lambda^l (A^{l+1}x + A^l b + A^{l-1}b + \cdots + b) \\ &= x + (1-\lambda) \sum_{l=0}^{\infty} \lambda^l \sum_{m=0}^{l} (A^m b + A^{m+1}x - A^m x) \\ &= x + (1-\lambda) \sum_{m=0}^{\infty} \left(\sum_{l=m}^{\infty} \lambda^l \right) (A^m b + A^{m+1}x - A^m x) \\ &= x + \sum_{m=0}^{\infty} \lambda^m (A^m b + A^{m+1}x - A^m x). \end{aligned}$$

or equivalently

$$r_{t+1} = \arg\min_{r\in\Re^s} \sum_{i=1}^{n} \xi_i \left(\phi(i)'r - \phi(i)'r_t - \sum_{m=0}^{\infty} \lambda^m ((A^m b)(i) + (A^{m+1}\Phi)(i)r_t - (A^m\Phi)(i)r_t) \right)^2,$$

where $(A^k\Phi)(i)$ denotes the ith row of the matrix $A^k\Phi$, and $(A^l b)(i)$ denotes the ith component of the vector $A^l b$, respectively. By solving for the minimum over r, we can write this iteration as

$$r_{t+1} = r_t + \left(\sum_{i=1}^{n} \xi_i \phi(i)\phi(i)' \right)^{-1} \sum_{i=1}^{n} \xi_i \phi(i) \left(\sum_{m=0}^{\infty} \lambda^m ((A^m b)(i) + (A^{m+1}\Phi)(i)r_t - (A^m\Phi)(i)r_t) \right). \tag{5.8}$$

We approximate this iteration by

$$r_{t+1} = r_t + \left(\sum_{k=0}^{t} \phi(i_k)\phi(i_k)' \right)^{-1} \sum_{k=0}^{t} \phi(i_k) \sum_{m=k}^{t} \lambda^{m-k} w_{k,m-k} d_t(i_m), \tag{5.9}$$

where $d_t(i_m)$ are the temporal differences

$$d_t(i_m) = b_{i_m} + w_{m,1}\phi(i_{m+1})'r_t - \phi(i_m)'r_t, \quad t \geq 0, m \geq 0. \tag{5.10}$$

Similarly to earlier cases, the basis for this is to replace the two expected values in the right-hand side of Eq. (5.8) with averages of samples corresponding to the states i_k, $k = 0, 1, \ldots$. In particular, we view

$\phi(i_k)\phi(i_k)'$ as a sample whose steady-state expected value is $\sum_{i=1}^{n} \xi_i \phi(i)\phi(i)'$,

$\phi(i_k) \sum_{m=k}^{t} \lambda^{m-k} w_{k,m-k} d_t(i_m)$ as a sample whose steady-state expected value is approximately

$$\sum_{i=1}^{n} \xi_i \phi(i) \sum_{m=0}^{\infty} \lambda^m ((A^m b)(i) + (A^{m+1}\Phi)(i)r_t - (A^m\Phi)(i)r_t).$$

Note that the summation of the second sample above is truncated at time t, but is a good approximation when k is much smaller than t and also when λ is small (see the convergence proof of the subsequent Proposition 4).

By using the temporal difference formula (5.10), we can write iteration (5.9) in compact form as

$$r_{t+1} = r_t + B_t^{-1}(C_t r_t + h_t), \tag{5.11}$$

where

$$B_t = \sum_{k=0}^{t} \phi(i_k)\phi(i_k)', \tag{5.12}$$

$$C_t = \sum_{k=0}^{t} \phi(i_k) \sum_{m=k}^{t} \lambda^{m-k} w_{k,m-k} (w_{m,1}\phi(i_{m+1}) - \phi(i_m))', \tag{5.13}$$

$$h_t = \sum_{k=0}^{t} \phi(i_k) \sum_{m=k}^{t} \lambda^{m-k} w_{k,m-k} b_{i_m}. \tag{5.14}$$

We now introduce the auxiliary vector

$$z_k = \sum_{m=0}^{k} \lambda^{k-m} w_{m,k-m} \phi(i_m), \tag{5.15}$$

and we will show that C_t and h_t can be written as

$$C_t = \sum_{k=0}^{t} z_k (w_{k,1}\phi(i_{k+1}) - \phi(i_k))', \tag{5.16}$$

$$h_t = \sum_{k=0}^{t} z_k b_{i_k}. \tag{5.17}$$

Thus, the quantities B_t, C_t, h_t, and z_t can be efficiently updated with the recursive formulas:

$$B_t = B_{t-1} + \phi(i_t)\phi(i_t)', \qquad C_t = C_{t-1} + z_t(w_{t,1}\phi(i_{t+1}) - \phi(i_t))',$$
$$h_t = h_{t-1} + z_t b_{i_t}, \qquad z_t = \lambda w_{t-1,1} z_{t-1} + \phi(i_t).$$

Indeed, we write Eq. (5.13) as

$$\begin{aligned} C_t &= \sum_{k=0}^{t}\sum_{m=k}^{t} \lambda^{m-k} w_{k,m-k}\phi(i_k)(w_{m,1}\phi(i_{m+1}) - \phi(i_m))' \\ &= \sum_{m=0}^{t}\sum_{k=0}^{m} \lambda^{m-k} w_{k,m-k}\phi(i_k)(w_{m,1}\phi(i_{m+1}) - \phi(i_m))' \\ &= \sum_{k=0}^{t}\sum_{m=0}^{k} \lambda^{k-m} w_{m,k-m}\phi(i_m)(w_{k,1}\phi(i_{k+1}) - \phi(i_k))' \\ &= \sum_{k=0}^{t} z_k(w_{k,1}\phi(i_{k+1}) - \phi(i_k))', \end{aligned}$$

thus proving Eq. (5.16). Similarly,

$$\begin{aligned} h_t &= \sum_{k=0}^{t}\sum_{m=k}^{t} \lambda^{m-k} w_{k,m-k}\phi(i_k) b_{i_m} \\ &= \sum_{m=0}^{t}\sum_{k=0}^{m} \lambda^{m-k} w_{k,m-k}\phi(i_k) b_{i_m} \\ &= \sum_{k=0}^{t}\sum_{m=0}^{k} \lambda^{k-m} w_{m,k-m}\phi(i_m) b_{i_k} \\ &= \sum_{k=0}^{t} z_k b_{i_k}, \end{aligned}$$

thus proving Eq. (5.17).

The approach of Section 2 yields an equation approximation method, which is based on solving the equation

$$C_t r + h_t = 0.$$

This method generalizes the LSTD(λ) algorithm of approximate DP, and is analogous to the APJ iteration (5.11).

We will now address the convergence of this method and the APJ iteration (5.11) to the solution of $\Phi r = \Pi T^{(\lambda)}(\Phi r)$. Their convergence hinges on the convergence of the matrix C_t and the vector h_t, and in the case of APJ, also the contraction property of $\Pi T^{(\lambda)}$, similarly to the special case of LSTD(λ) and LSPE(λ) in approximate DP (see [18,2,29]). In particular, letting Ξ be the diagonal matrix having the probabilities ξ_i along the diagonal, and using the full rank assumption on Φ, the equation $\Phi r = \Pi T^{(\lambda)}(\Phi r)$ can be written as

$$\Phi r = \Phi(\Phi'\Xi\Phi)^{-1}\Phi'\Xi(A^{(\lambda)}\Phi r + b^{(\lambda)}),$$

or,

$$(\Phi'\Xi\Phi)r = \Phi'\Xi(A^{(\lambda)}\Phi r + b^{(\lambda)}),$$

which are equivalent to $r = r + B^{-1}(Cr + h)$ or $Cr + h = 0$, respectively, where

$$B = \Phi'\Xi\Phi, \qquad C = \Phi'\Xi(A^{(\lambda)} - I)\Phi, \qquad h = \Phi'\Xi b^{(\lambda)}. \tag{5.18}$$

The equation $r = r + B^{-1}(Cr + h)$ corresponds to the Jacobi iteration $\Pi T^{(\lambda)}$ viewed on the space of r. It is clear that $\frac{1}{t+1}B_t \to B$ with probability 1, where B_t is given by Eq. (5.12). The convergence of $\frac{1}{t+1}C_t \to C$ and $\frac{1}{t+1}h_t \to h$ can be analyzed by viewing (z_t, i_t, i_{t+1}) jointly as a Markov process and exploiting the ergodicity property of the latter. Alternatively, one may also avoid dealing with the infinite state-space Markov chain by using a truncation argument which reduces the case of interest to a finite state-space Markov chain. We give such a convergence proof. We shall make the assumption $\lambda \max_{i,j} |a_{ij}|/p_{ij} < 1$, which is stronger than the condition $\lambda\sigma(A) < 1$, required for the multistep mapping $T^{(\lambda)}$ to be well-defined [the assumption implies that $\lambda|A| \le \beta P$ for some $\beta \in (0, 1)$, so using Proposition 1, $\lambda\sigma(A) \le \lambda\|A\|_\xi \le \beta\|P\|_\xi = \beta < 1$].

Proposition 4. *Assume that P is irreducible, and that λ satisfies $\lambda \max_{i,j} |a_{ij}|/p_{ij} < 1$ and $\lambda \in [0, 1)$. Let C_t, h_t, C, and h be given by Eqs. (5.13), (5.14), and (5.18), respectively, and let r^* be the solution of the equation $\Phi r = \Pi T^{(\lambda)}(\Phi r)$. Then, for every given initial C_0 and h_0, we have $\frac{1}{t+1}C_t \to C$ and $\frac{1}{t+1}h_t \to h$ with probability 1. Furthermore, $\hat{r}_t \to r^*$ with probability 1, where $\hat{r}_t$ is the solution of the equation $C_t r + h_t = 0$. If in addition $\sigma(A^{(\lambda)}) < 1$, then $r_t \to r^*$ with probability 1, where r_t is generated by the APJ iteration (5.11) starting with any r_0.*

Proof. We will show that $\frac{1}{t+1}C_t \to C$ with probability 1, and the argument for $\frac{1}{t+1}h_t \to h$ is similar. Consider the vector z_k of Eq. (5.15), and a "truncated" version involving the last $l+1$ terms of the summation:

$$z_{k,l} = \begin{cases} \sum_{j=0}^{l} \lambda^j w_{k-j,j}\phi(i_{k-j}) & \text{if } k \geq l, \\ z_k & \text{if } k < l. \end{cases} \tag{5.19}$$

Corresponding to $z_{k,l}$, we define $C_{t,l}$ by

$$C_{t,l} = C_{t-1,l} + z_{t,l}(w_{t,1}\phi(i_{t+1}) - \phi(i_t))',$$

with $C_{0,l} = C_0$. We have for some ϵ_l such that $\epsilon_l \to 0$ as $l \to \infty$,[4]

$$\|z_k - z_{k,l}\|_\infty \leq \epsilon_l. \tag{5.20}$$

For a matrix B, denote $\|B\|_\infty = \max_{i,j} |B_{ij}|$. Using Eq. (5.16), we have

$$\left\| \frac{1}{1+t}C_t - \frac{1}{1+t}C_{t,l} \right\|_\infty = \frac{1}{t+1} \left\| \sum_{k=l}^{t} (z_k - z_{k,l}) \left(w_{k,1}\phi(i_{k+1}) - \phi(i_k) \right)' \right\|_\infty,$$

so, using Eq. (5.20), we have for some constant $\bar{\epsilon}_l$ with $\bar{\epsilon}_l \to 0$,

$$\left\| \frac{1}{1+t}C_t - \frac{1}{1+t}C_{t,l} \right\|_\infty \leq \bar{\epsilon}_l.$$

This implies that

$$\liminf_{t\to\infty} \frac{1}{1+t}C_{t,l} - \bar{\epsilon}_l ee' \leq \liminf_{t\to\infty} \frac{1}{1+t}C_t \leq \limsup_{t\to\infty} \frac{1}{1+t}C_t \leq \limsup_{t\to\infty} \frac{1}{1+t}C_{t,l} + \bar{\epsilon}_l ee',$$

where e is the unit vector that has all components equal to 1. Thus $\lim_{t\to\infty} \frac{1}{1+t}C_t$ would exist and be the same as $\lim_{l\to\infty} \lim_{t\to\infty} \frac{1}{1+t}C_{t,l}$ when the latter limit exists.

To calculate $\lim_{l\to\infty} \lim_{t\to\infty} \frac{1}{1+t}C_{t,l}$, we fix l and consider $\lim_{t\to\infty} \frac{1}{1+t}C_{t,l}$. We view the $l+2$ consecutive states $(i_{k-l}, i_{k-l+1}, \ldots, i_k, i_{k+1})$ as the state of a Markov chain, which has a single recurrent class (since P is irreducible). Let us denote expected value with respect to its unique invariant distribution by $E_0\{\cdot\}$. Then, with probability 1, we have

$$\begin{aligned}
\lim_{t\to\infty} \frac{1}{1+t}C_{t,l} &= E_0\left\{ z_{l,l} \left(w_{l,1}\phi(i_{l+1}) - \phi(i_l) \right)' \right\} \\
&= E_0\left\{ \sum_{j=0}^{l} \lambda^j w_{l-j,j+1}\phi(i_{l-j})\phi(i_{l+1})' - \sum_{j=0}^{l} \lambda^j w_{l-j,j}\phi(i_{l-j})\phi(i_l)' \right\} \\
&= \sum_{j=0}^{l} \lambda^j \sum_{i_{l-j},\ldots,i_{l+1}} \xi_{i_{l-j}} \left(a_{i_{l-j}i_{l-j+1}} \cdots a_{i_l i_{l+1}} \right)\phi(i_{l-j})\,\phi(i_{l+1})' \\
&\quad - \sum_{j=0}^{l} \lambda^j \sum_{i_{l-j},\ldots,i_l} \xi_{i_{l-j}} \left(a_{i_{l-j}i_{l-j+1}} \cdots a_{i_{l-1}i_l} \right)\phi(i_{l-j})\phi(i_l)' \\
&= \sum_{j=0}^{l} \lambda^j \Phi' \Xi A^{j+1} \Phi - \sum_{j=0}^{l} \lambda^j \Phi' \Xi A^j \Phi \\
&= \Phi' \Xi \left(\sum_{j=0}^{l} (1-\lambda)\lambda^j A^{j+1} - I + \lambda^{l+1} A^{l+1} \right) \Phi.
\end{aligned}$$

[4] This is because $\lambda \max_{i,j} |a_{ij}|/p_{ij} < \beta$ for some positive $\beta < 1$ by our assumption, so by using Eq. (5.19), we can choose $\epsilon_l = \beta^l/(1-\beta)$.

The last expression converges to $\Phi'\Xi\left(A^{(\lambda)}-I\right)\Phi$ as $l\to\infty$, because

$$\lim_{l\to\infty}\sum_{j=0}^{l}(1-\lambda)\lambda^j A^{j+1}=A^{(\lambda)},\qquad \lim_{l\to\infty}\lambda^{l+1}A^{l+1}=0.$$

This proves that $\lim_{t\to\infty}\frac{1}{1+t}C_t=\Phi'\Xi\left(A^{(\lambda)}-I\right)\Phi$. The argument for showing that $\frac{1}{1+t}h_t\to h$ is similar. This implies the convergence of $\hat{r}_t$ and the APJ iteration. □

The assumption $\lambda\max_{i,j}|a_{ij}|/p_{ij}<1$ is important for the truncation argument in the preceding proof, as well as for the stability of the algorithms, as it makes z_t bounded. It would be worth trying to relax this assumption.

5.3. A generalization of TD(λ)

Finally, let us indicate a generalized version of the TD(λ) method of approximate DP [23]. It has the form

$$r_{t+1}=r_t+\gamma_t z_t d_t(i_t), \tag{5.21}$$

where γ_t is a diminishing positive scalar stepsize, z_t is given by Eq. (5.15), and $d_t(i_t)$ is the temporal difference given by Eq. (5.10). The analysis of TD(λ) that is most relevant to our work is the one in [24]. Much of this analysis generalizes easily. In particular, the idea of the convergence proof of [24] is to write the algorithm as

$$r_{t+1}=r_t+\gamma_t(Cr_t+h)+\gamma_t(V_t r_t+v_t),\quad t=0,1,\ldots,$$

where C and h are given by Eq. (5.18), and V_t and v_t are random matrices and vectors, respectively, which asymptotically have zero mean. The essence of the convergence proof of Tsitsiklis and Van Roy is that the matrix C is negative definite, in the sense that $r'Cr<0$ for all $r\neq 0$, so it has eigenvalues with negative real parts, which implies in turn that the matrix $I+\gamma_t C$ has eigenvalues strictly within the unit circle for sufficiently small γ_t. The following is a generalization of this key fact (Lemma 9 of [24]).

Proposition 5. *For all $\lambda\in[0,1)$, if $\Pi T^{(\lambda)}$ is a contraction on S with respect to $\|\cdot\|_\xi$, then the matrix C of Eq. (5.18) is negative definite.*

Proof. By the contraction assumption, we have for some $\alpha\in[0,1)$,

$$\|\Pi A^{(\lambda)}\Phi r\|_\xi\le\alpha\|\Phi r\|_\xi,\quad\forall\, r\in\Re^s. \tag{5.22}$$

Also, Π is given in matrix form as $\Pi=\Phi(\Phi'\Xi\Phi)^{-1}\Phi'\Xi$, from which it follows that

$$\Phi'\Xi(I-\Pi)=0. \tag{5.23}$$

Thus, we have for all $r\neq 0$,

$$\begin{aligned}
r'Cr &= r'\Phi'\Xi(A^{(\lambda)}-I)\Phi r\\
&= r'\Phi'\Xi((I-\Pi)A^{(\lambda)}+\Pi A^{(\lambda)}-I)\Phi r\\
&= r'\Phi'\Xi(\Pi A^{(\lambda)}-I)\Phi r\\
&= r'\Phi'\Xi\Pi A^{(\lambda)}\Phi r-\|\Phi r\|_\xi^2\\
&\le \|\Phi r\|_\xi\cdot\|\Pi A^{(\lambda)}\Phi r\|_\xi-\|\Phi r\|_\xi^2\\
&\le (\alpha-1)\|\Phi r\|_\xi^2\\
&< 0,
\end{aligned}$$

where the third equality follows from Eq. (5.23), the first inequality follows from the Cauchy–Schwarz inequality applied with the inner product $<x,y>=x'\Xi y$ that corresponds to the norm $\|\cdot\|_\xi$, and the second inequality follows from Eq. (5.22). □

The preceding proposition supports the validity of the algorithm (5.21), and provides a starting point for its analysis. However, the details are beyond the scope of the present paper.

6. Using basis functions involving powers of *A*

We have assumed in the preceding sections that the columns of Φ, the basis functions, are known, and the rows $\phi(i)'$ of Φ are explicitly available to use in the various simulation-based formulas. We will now discuss a class of basis functions

that may not be available, but may be approximated by simulation in the course of our algorithms. Let us first consider basis functions of the form $A^m g$, $m \geq 0$, where g is some vector in $\Re^n$. Such basis functions are implicitly used in the context of Krylov subspace methods; see e.g., [21]. A simple justification is that the fixed point of T has an expansion of the form

$$x^* = \sum_{t=0}^{\infty} A^t b,$$

provided the spectral radius of A is less than 1. Thus the basis functions $b, Ab, \ldots, A^s b$ yield an approximation based on the first $s+1$ terms of the expansion. Also a more general expansion is

$$x^* = \bar{x} + \sum_{t=0}^{\infty} A^t q,$$

where $\bar{x}$ is any vector in $\Re^n$ and q is the residual vector

$$q = T(\bar{x}) - \bar{x} = A\bar{x} + b - \bar{x};$$

this can be seen from the equation $x^* - \bar{x} = A(x^* - \bar{x}) + q$. Thus the basis functions $\bar{x}, q, Aq, \ldots, A^{s-1}q$ yield an approximation based on the first $s+1$ terms of the preceding expansion. Note that we have

$$A^m q = T^{m+1}(\bar{x}) - T^m(\bar{x}), \quad \forall\, m \geq 0,$$

so the subspace spanned by these basis functions is the subspace spanned by $\bar{x}, T(\bar{x}), \ldots, T^s(\bar{x})$.

Generally, to implement the methods of the preceding sections with basis functions of the form $A^m g$, $m \geq 0$, one would need to generate the ith components $(A^m g)(i)$ for any given i, but these may be hard to calculate. However, one can use instead single sample approximations of $(A^m g)(i)$, and rely on the formula

$$(A^m g)(i) \approx \frac{\sum_{k=0}^{t} \delta(i_k = i) w_{k,m} g_{i_{k+m}}}{\sum_{k=0}^{t} \delta(i_k = i)} \tag{6.1}$$

[cf. Eq. (5.4)]. Thus in principle, to approximate the algorithms of earlier sections using such basis functions, we only need to substitute each occurrence of $(A^m g)(i)$ in the vector $\phi(i)$ by a sample $w_{k,m} g_{i_{k+m}}$ generated independently of the "main" Markov chain trajectory.

It is possible to use, in addition to $g, Ag, \ldots, A^s g$, other basis functions, whose components are available with no error, or to use several sets of basis functions of the form $g, Ag, \ldots, A^s g$, corresponding to multiple vectors g. However, for simplicity in what follows in this section, we assume that the only basis functions are $g, Ag, \ldots, A^s g$ for a single given vector g, so the matrix Φ has the form

$$\Phi = \begin{pmatrix} g & Ag & \cdots & A^s g \end{pmatrix}. \tag{6.2}$$

The ith row of Φ is

$$\phi(i)' = \begin{pmatrix} g(i) & (Ag)(i) & \cdots & (A^s g)(i) \end{pmatrix}.$$

We will focus on a version of the equation approximation method of Section 2, which uses single sample approximations of these rows. The multistep methods of Section 5 admit similar versions, since the corresponding formulas involve powers of A multiplying vectors, which can be approximated using Eq. (6.1).

We recall [cf. Eq. (2.1)] that the projected equation $\Phi r = \Pi(A\Phi r + b)$ has the form

$$\sum_{i=1}^{n} \xi_i \phi(i) \left(\phi(i) - \sum_{j=1}^{n} a_{ij} \phi(j) \right)' r^* = \sum_{i=1}^{n} \xi_i \phi(i) b_i, \tag{6.3}$$

or equivalently, using Eq. (6.2),

$$\sum_{i=1}^{n} \xi_i \begin{pmatrix} g(i) \\ (Ag)(i) \\ \vdots \\ (A^s g)(i) \end{pmatrix} \begin{pmatrix} g(i) - (Ag)(i) & (Ag)(i) - (A^2 g)(i) & \cdots & (A^s g)(i) - (A^{s+1} g)(i) \end{pmatrix} r^* = \sum_{i=1}^{n} \xi_i \begin{pmatrix} g(i) \\ (Ag)(i) \\ \vdots \\ (A^s g)(i) \end{pmatrix} b_i. \tag{6.4}$$

To approximate this equation, we generate the index sequence $\{i_0, i_1, \ldots\}$ according to a distribution ξ with positive components. For each i_k, we also generate two additional mutually "independent" sequences

$$\left\{ (i_k, \hat{i}_{k,1}), (\hat{i}_{k,1}, \hat{i}_{k,2}), \ldots, (\hat{i}_{k,s-1}, \hat{i}_{k,s}) \right\}, \qquad \left\{ (i_k, \tilde{i}_{k,1}), (\tilde{i}_{k,1}, \tilde{i}_{k,2}), \ldots, (\tilde{i}_{k,s}, \tilde{i}_{k,s+1}) \right\},$$

of lengths s and $s+1$, respectively, according to transition probabilities p_{ij}. At time t, we form the following linear equation to approximate Eq. (6.4):

$$\sum_{k=0}^{t}\begin{pmatrix} g_{i_k} \\ \hat{w}_{k,1}g_{\hat{i}_{k,1}} \\ \vdots \\ \hat{w}_{k,s}g_{\hat{i}_{k,s}} \end{pmatrix}\left(g_{i_k}-\tilde{w}_{k,1}\,g_{\tilde{i}_{k+1}}\tilde{w}_{k,1}g_{\tilde{i}_{k+1}}-\tilde{w}_{k,2}g_{\tilde{i}_{k+2}}\cdots\tilde{w}_{k,s}g_{\tilde{i}_{k+s}}-\tilde{w}_{k,s+1}g_{\tilde{i}_{k+s+1}}\right)r=\sum_{k=0}^{t}\begin{pmatrix} g_{i_k} \\ \hat{w}_{k,1}g_{\hat{i}_{k,1}} \\ \vdots \\ \hat{w}_{k,s}g_{\hat{i}_{k,s}} \end{pmatrix}b_{i_k}, \tag{6.5}$$

where for all m,

$$\hat{w}_{k,m}=\frac{a_{i_k\hat{i}_{k,1}}}{p_{i_k\hat{i}_{k,1}}}\frac{a_{\hat{i}_{k,1}\hat{i}_{k,2}}}{p_{\hat{i}_{k,1}\hat{i}_{k,2}}}\cdots\frac{a_{\hat{i}_{k,m-1}\hat{i}_{k,m}}}{p_{\hat{i}_{k,m-1}\hat{i}_{k,m}}},\qquad \tilde{w}_{k,m}=\frac{a_{i_k\tilde{i}_{k,1}}}{p_{i_k\tilde{i}_{k,1}}}\frac{a_{\tilde{i}_{k,1}\tilde{i}_{k,2}}}{p_{\tilde{i}_{k,1}\tilde{i}_{k,2}}}\cdots\frac{a_{\tilde{i}_{k,m-1}\tilde{i}_{k,m}}}{p_{\tilde{i}_{k,m-1}\tilde{i}_{k,m}}};$$

[cf. Eq. (5.3)].

To verify the validity of this approximation, we can use Eq. (6.1), and a similar analysis to the one of Section 2. We omit the straightforward details. We can also construct a corresponding approximate Jacobi method along similar lines.

The preceding methodology can be extended in a few different ways. A similar method can be used in the case where the rows $\phi(i)'$ of Φ represent expected values with respect to some distribution depending on i, and can be calculated by simulation. Then, the terms $\phi(i)$ and $\phi(i)-\sum_{j=1}^{n}a_{ij}\phi(j)$ in Eq. (6.3) may be replaced by mutually independently generated samples, and the equation approximation formulas may be appropriately adjusted in similar spirit as Eq. (6.5). Furthermore, the device of using an extra independent sequence per time step, may also be used to construct l-step methods (cf. Section 5) where the index sequence $\{i_0,i_1,\ldots\}$ and the transition sequence $\{(i_0,j_0),(i_1,j_1),\ldots\}$ are generated by using different transition matrices. Note, however, that the ideas of the present section are harder to use in conjunction with the λ-methods of Section 5, because the corresponding mapping $T^{(\lambda)}$ involves an infinite number of steps.

We note that constructing basis functions for subspace approximation is an important research issue, and has received considerable attention recently in the approximate DP literature (see, e.g., [15,17,19,27]). However, the methods of the present section are new, even within the context of approximate DP, and to our knowledge, they are the first proposals to introduce sampling for basis function approximation directly within the TD(λ), LSTD, and LSPE-type methods.

Let us finally point out a generic difficulty associated with the method of this section: even if a solution $r^*=(r_0^*,r_1^*,\ldots,r_s^*)$ of the projected fixed point equation $\Phi r=\Pi T(\Phi r)$ is found, the approximation of the ith component of x^* has the form

$$\phi(i)'r^*=\sum_{m=0}^{s}r_m^*(A^mg)(i),$$

and requires the evaluation of the basis function components $(Ag)(i),\ldots,(A^sg)(i)$. For this, additional computation and simulation is needed, using the approximation formula (6.1).

7. Extensions and related methods

In this section, we briefly discuss how some of the ideas of earlier sections can be extended to address other types of problems, including some that are nonlinear.

7.1. Least squares problems

In a simple view of the methods of the preceding sections, we start from a least squares type of problem or a deterministic iterative algorithm for solving that problem, and we replace some of the exact expressions appearing in the problem or the algorithm by simulation-based approximations. This idea can be applied to several different least squares contexts and in a variety of ways. As illustration, we present a few examples.

Consider solving the problem

$$\min_{r\in\Re^s}\|A\Phi r-b\|_\zeta^2 \tag{7.1}$$

to obtain an approximation to the weighted least squares solution of a system $Ax=b$. Here A is an $m\times n$ matrix, ζ is a known probability distribution vector with positive components, b is a vector in $\Re^m$, and as before, Φ is an $n\times s$ matrix of basis functions. The solution is

$$r^*=(\Phi'A'ZA\Phi)^{-1}\Phi'A'Zb,$$

where Z is the diagonal $m\times m$ matrix having ζ along the diagonal, and we assume for simplicity that $\Phi'A'ZA\Phi$ is invertible. An approximation to this solution can be obtained by replacing $\Phi'A'ZA\Phi$ and $\Phi'A'Zb$ with estimates that are obtained by simulation and low-dimensional calculations.

In the most straightforward way to do this, we generate an infinite index sequence $i_0, i_1, \ldots$ by independently sampling the set $\{1, \ldots, m\}$ according to the distribution ζ. Simultaneously, we generate two independent sequences of corresponding transitions $\{(i_0, j_0), (i_1, j_1), \ldots\}$ and $\{(i_0, \hat{j}_0), (i_1, \hat{j}_1), \ldots\}$ according to transition probabilities p_{ij}, where $p_{ij} > 0$ whenever $a_{ij} \neq 0$. We then approximate r^* by $\hat{r} = C_t^{-1} c_t$, where

$$C_t = \frac{1}{t+1} \sum_{k=0}^{t} \frac{a_{i_k j_k}}{p_{i_k j_k}} \frac{a_{i_k \hat{j}_k}}{p_{i_k \hat{j}_k}} \phi(j_k) \phi(\hat{j}_k)', \qquad c_t = \frac{1}{t+1} \sum_{k=0}^{t} \frac{a_{i_k j_k}}{p_{i_k j_k}} \phi(j_k) b_{i_k}.$$

We may also use in place of C_t, its symmetrized version, which is $(C_t + C_t')/2$. It can be seen that $C_t \to \Phi' A' Z A \Phi$ and $c_t \to \Phi' A' Z b$ with probability 1, so $\hat{r}_t \to r^*$ as $t \to \infty$, with probability 1.

As another illustration, consider a discrete-time stochastic process with states $i = 1, \ldots, n$, which is ergodic in the sense that there exists $\xi = (\xi_1, \ldots, \xi_n)$, such that its generated trajectories $\{i_0, i_1, \ldots\}$ satisfy

$$\xi_i = \lim_{t \to \infty} \frac{\sum_{k=0}^{t} \delta(i_k = i)}{t+1}, \quad \forall\, i = 1, \ldots, n, \tag{7.2}$$

with probability 1. A special case is a Markov chain with a single recurrent class, in which case ξ is the invariant distribution of the chain. Other special cases, which are not easily modeled by a Markov chain, arise in queueing network applications. Consider approximating ξ with a mixture distribution Φr^*, where r^* solves the problem

$$\min_{e' r = 1, r \geq 0} \frac{1}{2} \|\xi - \Phi r\|_\zeta^2. \tag{7.3}$$

Here e is the unit vector, ζ is a known distribution with positive components, and Φ is an $n \times s$ matrix whose columns are basis functions that are distributions. To approximate the problem (7.3), we generate an infinitely long trajectory $\{i_0, i_1, \ldots\}$ of the process. We also generate an index sequence $\{\hat{i}_0, \hat{i}_1, \ldots\}$ by independently sampling the states of the process according to the distribution ζ. We then solve the problem

$$\min_{e' r = 1, r \geq 0} \frac{1}{2} r' C_t r - c_t' r, \tag{7.4}$$

where

$$C_t = \frac{1}{t+1} \sum_{k=0}^{t} \phi(\hat{i}_k) \phi(\hat{i}_k)', \qquad c_t = \frac{1}{t+1} \sum_{k=0}^{t} \zeta_{i_k} \phi(i_k).$$

It can be seen that asymptotically, as $t \to \infty$, the cost function of problem (7.4) converges with probability 1 to the cost function of problem (7.3) minus the constant $\frac{1}{2}\|\xi\|_\zeta^2$.

We also note that related methods may be used to calculate an approximation to the spectrum of a matrix $A'A$, and in particular the dominant eigenvalue and eigenvector of $A'A$ (see [5], and the related paper [1], which deals with the calculation of an approximation to the Perron–Frobenius eigenvector of a nonnegative matrix).

A potential difficulty with the preceding algorithms (including the ones of Sections 2–6) is the amount of simulation noise involved in forming reliable estimates of the approximated matrices and vectors. In special cases, the structure of the problem may be exploited to reduce the noise; for example, some components may be estimated separately from others (or computed exactly) to apply more effectively variance reduction techniques.

7.2. Differentiable nonlinear fixed point problems

One potential approach for the general fixed point equation $x = T(x)$, where T is a differentiable mapping, is to use Newton's method to solve the projected equation. In this approach, given r_k, we generate r_{k+1} by using one of the simulation-based methods given earlier to solve a linearized version (at r_k) of the projected equation $\Phi r = \Pi T(\Phi r)$. This is the linear equation

$$\Phi r_{k+1} = \Pi(T(\Phi r_k) + J_k \Phi(r_{k+1} - r_k)),$$

where J_k is the Jacobian matrix of T, evaluated at Φr_k. We do not discuss this approach further, and focus instead on a special case involving a contraction mapping, where convergence from any starting point r_0 is guaranteed.

7.3. Extension of Q-learning for optimal stopping

Let us consider a system of the form

$$x = T(x) = Af(x) + b,$$

where $f : \Re^n \mapsto \Re^n$ is a mapping with scalar function components of the form $f(x) = (f_1(x_1), \ldots, f_n(x_n))$. We assume that each of the mappings $f_i : \Re \mapsto \Re$ is nonexpansive in the sense that

$$|f_i(x_i) - f_i(\bar{x}_i)| \leq |x_i - \bar{x}_i|, \quad \forall\, i = 1, \ldots, n, x_i, \bar{x}_i \in \Re. \tag{7.5}$$

This guarantees that T is a contraction mapping with respect to any norm $\|\cdot\|$ with the property

$$\|y\| \leq \|z\| \quad \text{if } |y_i| \leq |z_i|, \quad \forall\, i = 1, \ldots, n,$$

whenever A is a contraction with respect to that norm. Such norms include weighted l_1 and l_∞ norms, the norm $\|\cdot\|_\xi$, as well as any scaled Euclidean norm $\|x\| = \sqrt{x'Dx}$, where D is a positive definite symmetric matrix with nonnegative components. Under the assumption (7.5), the theory of Section 3 applies and suggests appropriate choices of a Markov chain for simulation.

A special case has been studied in the context of an optimal stopping problem in [26], which gave a Q-learning algorithm that is similar in spirit to TD(0). The following example outlines this context.

Example 5 (*Optimal Stopping*). Consider the equation

$$x = T(x) = \alpha P f(x) + b,$$

where P is an irreducible transition probability matrix with invariant distribution ξ, $\alpha \in (0, 1)$ is a scalar discount factor, and f is a mapping with components

$$f_i(x_i) = \min\{c_i, x_i\}, \quad i = 1, \ldots, n,$$

where c_i are some scalars. This is the Q-factor equation corresponding to a discounted optimal stopping problem with states $i = 1, \ldots, n$, and a choice between two actions at each state i: stop at a cost c_i, or continue at a cost b_i and move to state j with probability p_{ij}. The optimal cost starting from state i is $\min\{c_i, x_i^*\}$, where x^* is the fixed point of T, which is unique because T is a sup-norm contraction, as shown in [26]. As a special case of Proposition 1, we obtain that ΠT is a contraction with respect to $\|\cdot\|_\xi$, and the associated error bounds apply. Similar results hold in the case where αP is replaced by a matrix A satisfying condition (2) of Proposition 1, or the conditions of Proposition 2. The case where $\sum_{j=1}^n |a_{\bar{i}j}| < 1$ for some index $\bar{i}$, and $0 \leq A \leq Q$, where Q is an irreducible transition probability matrix, corresponds to an optimal stopping problem where the stopping state will be reached from all other states with probability 1, even without applying the stopping action. In this case, by Proposition 1 under condition (3), ΠA is a contraction with respect to some norm, and hence $I - \Pi A$ is invertible. Using this fact, it follows by modifying the proof of Proposition 2 that $\Pi((1-\gamma)I + \gamma T)$ is a contraction with respect to $\|\cdot\|_\xi$.

We will now describe an approximate Jacobi algorithm that extends the method proposed in [30] for the optimal stopping problem of the preceding example. Similarly to Section 4, the projected Jacobi iteration

$$\Phi r_{t+1} = \Pi T(\Phi r_t), \quad t = 0, 1, \ldots,$$

takes the form

$$r_{t+1} = \left(\sum_{i=1}^n \xi_i \phi(i)\phi(i)'\right)^{-1} \sum_{i=1}^n \xi_i \phi(i) \left(\sum_{j=1}^n a_{ij} f_j\left(\phi(j)' r_t\right) + b_i\right).$$

We approximate this iteration with

$$r_{t+1} = \left(\sum_{k=0}^t \phi(i_k)\phi(i_k)'\right)^{-1} \sum_{k=0}^t \phi(i_k) \left(\frac{a_{i_k j_k}}{p_{i_k j_k}} f_{j_k}\left(\phi(j_k)' r_t\right) + b_{i_k}\right). \tag{7.6}$$

Here, as before, $\{i_0, i_1, \ldots\}$ is a state sequence, and $\{(i_0, j_0), (i_1, j_1), \ldots\}$ is a transition sequence satisfying Eqs. (2.2) and (2.4) with probability 1. The justification of this approximation is very similar to the ones given so far, and will not be discussed further.

A difficulty with iteration (7.6) is that the terms $f_{j_k}\left(\phi(j_k)' r_t\right)$ must be computed for all $k = 0, \ldots, t$, at every step t, thereby resulting in significant overhead. Methods to bypass this difficulty in the case of optimal stopping are given in [30], and can be extended to the more general context of this paper. We finally note that due to the nonlinearity of T, it is hard to implement the equation approximation methods of Section 2. Furthermore, there are no corresponding versions of the multistep methods of Section 5.

8. Conclusions

In this paper we have shown how linear fixed point equations can be solved approximately by projection on a low-dimensional subspace and simulation, thereby generalizing recent methods from the field of approximate DP. We have given

error bounds that apply to special types of contraction mappings, most prominently some involving diagonally dominant matrices. However, our methods apply to any linear system of equations whose projected solution is unique. While our simulation-based methods are likely not competitive with other methods for moderately-sized problems, they provide an approach for addressing extremely large problems, because they do not involve any vector operations or storage of size comparable to the original problem dimension. Our methods have been motivated by recent analysis and computational experience in approximate DP. Much remains to be done to apply them and to assess their potential in other fields.

Acknowledgments

Helpful discussions with N. Polydorides and P. Tseng are gratefully acknowledged. Dimitri Bertsekas' research was supported in part by NSF Grant ECCS-0801549. Huizhen Yu's research was supported in part by the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778.

References

[1] A. Basu, Bhattacharyya, V. Borkar, A learning algorithm for risk-sensitive cost, Tech. Report No. 2006/25, Dept. of Math., Indian Institute of Science, Bangalore, India, 2006.
[2] D.P. Bertsekas, V. Borkar, A. Nedić, Improved temporal difference methods with linear function approximation, in: J. Si, A. Barto, W. Powell (Eds.), Learning and Approximate Dynamic Programming, IEEE Press, NY, 2004.
[3] A.G. Barto, M. Duff, Monte Carlo matrix inversion and reinforcement learning, Advances in Neural Information Processing Systems 6 (1994) 687–694.
[4] D.P. Bertsekas, S. Ioffe, Temporal differences-based policy iteration and applications in neuro-dynamic programming, Lab. for Info. and Dec. Sys. Report LIDS-P-2349, MIT, Cambridge, MA, 1996.
[5] D.P. Bertsekas, H. Yu, Solution of large systems of equations using approximate dynamic programming methods, Lab. for Info. and Dec. Sys. Report LIDS-2754, MIT, Cambridge, MA, 2007.
[6] D.P. Bertsekas, Dynamic Programming and Optimal Control, Vol. II, 3rd ed., Athena Scientific, Belmont, MA, 2007.
[7] D.P. Bertsekas, J.N. Tsitsiklis, Neuro-Dynamic Programming, Athena Scientific, Belmont, MA, 1996.
[8] J.A. Boyan, Technical update: Least-squares temporal difference learning, Machine Learning 49 (2002) 1–15.
[9] S.J. Bradtke, A.G. Barto, Linear least-squares algorithms for temporal difference learning, Machine Learning 22 (1996) 33–57.
[10] D.S. Choi, B. Van Roy, A generalized Kalman filter for fixed point approximation and efficient temporal-difference learning, Discrete Event Dynamic Systems: Theory and Applications 16 (2006) 207–239.
[11] J.H. Curtiss, Monte Carlo methods for the iteration of linear operators, Journal of Mathematics and Physics 32 (1953) 209–232.
[12] J.H. Curtiss, A theoretical comparison of the efficiencies of two classical methods and a Monte Carlo method for computing one component of the solution of a set of linear algebraic equations, in: H.A. Meyer (Ed.), Symposium on Monte Carlo Methods, Wiley, New York, NY, 1954, pp. 191–233.
[13] G.E. Forsythe, R.A. Leibler, Matrix inversion by a Monte Carlo method, Mathematical Tables and Other Aids to Computation 4 (1950) 127–129.
[14] J.H. Halton, A retrospective and prospective survey of the Monte Carlo method, SIAM Review 12 (1970) 1–63.
[15] P. Keller, S. Mannor, D. Precup, Automatic basis function construction for approximate dynamic programming and reinforcement learning, in: Proceedings of the Twenty-third International Conference on Machine Learning, 2006.
[16] J.S. Liu, Monte Carlo Strategies in Scientific Computing, Springer, NY, 2001.
[17] I. Menache, S. Mannor, N. Shimkin, Basis function adaptation in temporal difference reinforcement learning, Annals of Operations Research 134 (2005).
[18] A. Nedić, D.P. Bertsekas, Least squares policy evaluation algorithms with linear function approximation, Discrete Event Dynamic Systems: Theory and Applications 13 (2003) 79–110.
[19] R. Parr, C. Painter-Wakefield, L. Li, M. Littman, Analyzing feature generation for value-function approximation, in: Proc. of the 24th International Conference on Machine Learning, Corvallis, OR. 2007.
[20] M.L. Puterman, Markov Decision Processes: Discrete Stochastic Dynamic Programming, J. Wiley, NY, 1994.
[21] Y. Saad, Iterative Methods for Sparse Linear Systems, 2nd ed., SIAM, Phila., PA, 2003.
[22] R.S. Sutton, A.G. Barto, Reinforcement Learning, MIT Press, Cambridge, MA, 1998.
[23] R.S. Sutton, Learning to predict by the methods of temporal differences, Machine Learning 3 (1988) 9–44.
[24] J.N. Tsitsiklis, B. Van Roy, An analysis of temporal-difference learning with function approximation, IEEE Transactions on Automatic Control 42 (1997) 674–690.
[25] J.N. Tsitsiklis, B. Van Roy, Average cost temporal-difference learning, Automatica 35 (1999) 1799–1808.
[26] J.N. Tsitsiklis, B. Van Roy, Optimal stopping of Markov processes: Hilbert space theory, approximation algorithms, and an application to pricing financial derivatives, IEEE Transactions on Automatic Control 44 (1999) 1840–1851.
[27] M.J. Valenti, Approximate dynamic programming with applications in multi-agent systems, Ph.D. Thesis, Dept. of Electrical Engineering and Computer Science, MIT, 2007.
[28] W.R. Wasow, A note on the inversion of matrices by random walks, Mathematical Tables and Other Aids to Computation 6 (1952) 78–81.
[29] H. Yu, D.P. Bertsekas, Convergence results for some temporal difference methods based on least squares, Lab. for Info. and Dec. Sys. Report 2697, MIT, 2006.
[30] H. Yu, D.P. Bertsekas, A least squares Q-learning algorithm for optimal stopping problems, Lab. for Info. and Dec. Sys. Report 2731, MIT, 2007.