Automatica 50 (2014) 1940–1943

Contents lists available at ScienceDirect

Automatica

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

Technical communique

Value set iteration for Markov decision processes☆

Hyeong Soo Chang [1]

Department of Computer Science and Engineering, Sogang University, Seoul, Republic of Korea

ARTICLE INFO

Article history:
Received 8 November 2013
Received in revised form
21 February 2014
Accepted 30 April 2014
Available online 4 June 2014

Keywords:
Markov decision processes
Value iteration
Dynamic programming
Constrained optimization

ABSTRACT

This communique presents an algorithm called "value set iteration" (VSI) for solving infinite horizon discounted Markov decision processes with finite state and action spaces as a simple generalization of value iteration (VI) and as a counterpart to Chang's policy set iteration. A sequence of value functions is generated by VSI based on manipulating a set of value functions at each iteration and it converges to the optimal value function. VSI preserves convergence properties of VI while converging no slower than VI and in particular, if the set used in VSI contains the value functions of independently generated sample-policies from a given distribution and a properly defined policy switching policy, a probabilistic exponential convergence rate of VSI can be established. Because the set used in VSI can contain the value functions of any policies generated by other existing algorithms, VSI is also a general framework of combining multiple solution methods.

1. Introduction

Consider a Markov decision process (MDP) (Puterman, 1994) (X, A, P, R), where X is a finite state set, $A(x)$ is a finite action set at $x \in X$ with $\bigcup_{x\in X} A(x) = A$, R is a reward function such that $R(x, a) \in \mathbb{R}, x \in X, a \in A(x)$, and P is a transition function that maps $\{(x, a)|x \in X, a \in A(x)\}$ to the set of probability distributions over X. We denote the probability of making a transition to state $y \in X$ when taking an action $a \in A(x)$ at state $x \in X$ by P^a_{xy}.

We define a (stationary Markovian) policy π as a mapping from X to A with $\pi(x) \in A(x), \forall x \in X$, and let Π be the set of all such policies. Define *the value function* V^π *of* $\pi \in \Pi$ over X such that

$$V^\pi(x) := E\left[\sum_{t=0}^{\infty} \gamma^t R(X_t, \pi(X_t))|X_0 = x\right], \quad x \in X,$$

where X_t is a random variable denoting state at time t by following π and $\gamma \in (0, 1)$ is a discounting factor.

Our goal is to find the optimal value function V^* where $V^*(x) = \max_{\pi\in\Pi} V^\pi(x), x \in X$, or to find an optimal policy $\pi^* \in \Pi$ that achieves $V^*(x)$ at all $x \in X$.

Let $B(X)$ be the set of all real-valued functions on X and define a mapping $L : B(X) \to B(X)$ such that for $x \in X$ and $u \in B(X)$,

$$L(u)(x) := \max_{a\in A(x)} \left(R(x, a) + \gamma \sum_{y\in X} P^a_{xy} u(y)\right).$$

It is well-known that V^* uniquely satisfies $L(V^*) = V^*$ and the sequence of value functions $\{U_k\}$ generated by iterative applications of L with an arbitrary initial value function $U_0 \in B(X)$ such that $L(U_k) = U_{k+1}, k \geq 0$, converges to V^* and this exact method is called value iteration (VI) (Puterman, 1994). Because L is a contraction mapping in $B(X)$ with γ-contraction, VI's convergence rate is linear with the rate of γ and VI produces an ϵ-optimal policy for any given $\epsilon > 0$. The ϵ-optimal policy is guaranteed to be exactly optimal for sufficiently small ϵ because Π is finite. VI's running time-complexity is polynomial in $|X|, |A|, 1/(1-\gamma)$, and the size of representing the inputs R and P (Blondel & Tsitsiklis, 2000).

VI is known to be one of the two exact dynamic-programming methods along with policy iteration (PI) (Puterman, 1994). Due to its simplicity and the exactness but the dimensional non-scalability, since VI was developed by Bellman (1957), a great body of works has been done to implement it in real applications and to improve or approximate it (to have an approximately optimal policy) over the decades (see, e.g., Bertsekas & Tsitsiklis, 1996, Chang, Hu, Fu, & Marcus, 2013, Powell, 2011 and Puterman, 1994 and the references therein). In particular, there exist (classical) variants of VI, e.g., Jacobi, Gauss–Seidel, action-elimination, etc., (see, e.g., Puterman, 1994 and other variants therein) aiming at reducing computational complexity of VI and possibly improving

☆ The material in this paper was not presented at any conference. This paper was recommended for publication in revised form by Associate Editor Nuno C. Martins under the direction of Editor André L. Tits.

E-mail address: hschang@sogang.ac.kr.

[1] Tel.: +82 2 705 8925; fax: +82 2 704 8273.

http://dx.doi.org/10.1016/j.automatica.2014.05.009
0005-1098/

the linear convergence rate by γ-contraction with no smaller contraction. These methods maintain the exactness of VI and are based on a single value-function manipulation. There are also some more recent efforts on devising an exact algorithm as a variant of VI to improve the convergence rate of VI by designing acceleration operator/estimator (Herzberg & Yechiali, 1994; Shlakhter, Lee, Khmelev, & Javer, 2010) or by using a sequence of truncated models (Arruda, Ourique, LaCombe, & Almudevar, 2013) with a single value-function manipulation but the degree of speeding up is not theoretically quantified even if some experimental results are provided to advocate these approaches.

In this communique, we present a novel *exact* algorithm called "value set iteration". (VSI) for solving MDPs as a simple *generalization* of VI and as a counterpart to policy set iteration (PSI) recently presented by Chang (2013). PSI is a generalization of PI by manipulating a set of policies at each iteration. As a counterpart to PSI, VSI generates a sequence of value functions based on manipulating *a set of value functions* at each iteration, as opposed to other existing exact variants of VI, and it converges to the optimal value function. VSI preserves convergence properties of VI while converging no slower than VI. In particular, if the value-function set used in VSI contains the value functions of $N \geq 1$ independently generated sample-policies from a given distribution and a properly defined policy switching policy (Chang et al., 2013), a probabilistic exponential convergence rate of VSI can be additionally established in terms of N but *independently of* γ, similar to PSI. This then potentially overcomes a major problem of the dependence on $1/(1-\gamma)$ in the running time-complexity of VI. Because the set used in VSI can contain the value functions of any policies generated by other existing algorithms, VSI is also a *general framework of combining multiple solution methods*.

We note that even if VSI manipulates a set of value functions as in PSI, PSI is based on extending the single-policy improvement step of PI into a multi-policy improvement step whereas VSI is based on a newly devised contraction-mapping operator in the space of value functions. The operator is defined for the first time in this work and iterative approximation by successive applications of the operator is totally different aspect from PSI. Each iteration of VSI requires $O((N+m+1)(|X|^2|A|+|X|^3))$ time-complexity if the set involved with VSI at each iteration contains N sample-policies and $m \geq 0$ additional arbitrarily chosen policies in Π. Therefore, we establish that by allowing an increment in the per-iteration time-complexity of VI by a factor of about N and by the amount of evaluating the value functions in the set, no slower convergence than VI in terms of the number of iterations is guaranteed while achieving a probabilistic exponential convergence rate. We provide a finite-time probabilistic error-bound in obtaining the optimal value function for a given initial state distribution (cf., Theorem 4). One of the key ideas for the analysis is based on a probability bound of sample-maximum estimate of a random variable (Campi & Calafiore, 2009) obtained from the scenario design method (Calafiore, 2010; Calafiore & Campi, 2006) to effectively solve control design problems that can be cast in the form of a convex optimization problem with uncertain constraints. In this sense, as in PSI, VSI takes the spirit of randomized methods in probabilistic robust control.

2. Value set iteration

2.1. General framework

Let $\mathcal{P}(\Pi)$ be the power set of Π. Define a mapping $T : B(X) \times \mathcal{P}(\Pi) \to B(X)$ such that for $x \in X$, $u \in B(X)$, and nonempty $\Delta \in \mathcal{P}(\Pi)$,

$$T(u,\Delta)(x) := \max_{a\in A(x)}\left(R(x,a)+\gamma\sum_{y\in X}P^a_{xy}\max\left\{u(y),\max_{\pi\in\Delta}V^\pi(y)\right\}\right)$$

and $T(u,\Delta)(x) := L(u)(x)$ if $\Delta = \emptyset$.

The following lemma states that similar to L, V^* is a unique fixed point of T for any $\Delta \in \mathcal{P}(\Pi)$ and T is also a contraction mapping in $B(X)$ for any Δ. In the sequel, the norm $\|\cdot\|$ denotes $\max_{x\in X}|f(x)|$ for $f \in B(X)$ and for $u, v \in B(X)$, $u \leq (\geq) v$ means $u(x) \leq (\geq) v(x)$ for all $x \in X$.

Lemma 1. *With the mapping T, the following holds for any $\Delta \in \mathcal{P}(\Pi)$:*

1. *V^* uniquely satisfies $T(V^*,\Delta) = V^*$.*
2. *For any $u, v \in B(X)$, $\|T(u,\Delta) - T(v,\Delta)\| \leq \gamma\|u-v\|$.*

Proof. The proof of (1) is from the definitions of T and V^* and Banach's fixed point theorem. For the part (2), if $\Delta = \emptyset$, it is trivial. If $\Delta \neq \emptyset$, then for any $x \in X$ and $u, v \in B(X)$,

$$\begin{aligned}
&T(u,\Delta)(x) - T(v,\Delta)(x) \leq \gamma\sum_{y\in X}P^{a^*}_{xy}\left(\max\left\{u(y),\max_{\pi\in\Delta}V^\pi(y)\right\}\right.\\
&\quad\left.-\max\left\{v(y),\max_{\pi\in\Delta}V^\pi(y)\right\}\right)\\
&\text{where } a^* \in \arg\max_{a\in A(x)}\left(R(x,a)\right.\\
&\quad\left.+\gamma\sum_{y\in X}P^a_{xy}\max\left\{u(y),\max_{\pi\in\Delta}V^\pi(y)\right\}\right)\\
&\leq \gamma\max_{z\in X}\left|\max\left\{u(z),\max_{\pi\in\Delta}V^\pi(z)\right\}-\max\left\{v(z),\max_{\pi\in\Delta}V^\pi(z)\right\}\right|\\
&\leq \gamma\max_{z\in X}\left|u(z)-v(z)\right|.
\end{aligned}$$

Changing the role of u and v, we have that $T(v,\Delta)(x) - T(u,\Delta)(x) \leq \gamma\max_{z\in X}|u(z)-v(z)|$. This concludes $\|T(u,\Delta)-T(v,\Delta)\| \leq \gamma\|u-v\|$.

We now provide VSI below. VSI degenerates to VI if $\Delta_k = \emptyset$ for all $k \geq 0$. The structure of the algorithm follows that of VI. A sequence of the value functions $\{V_k\}$ is generated by successive applications of T with $V_0 \in B(X)$ where an arbitrary $\Delta_k \in \mathcal{P}(\Pi)$ is employed at k.

Value set iteration (VSI)

1. **Initialization**: Select $\epsilon > 0$. Set $k = 0$ and choose any $V_0 \in B(X)$.
2. **Loop**:
   2.1 Select $\Delta_k \in \mathcal{P}(\Pi)$ and obtain $V_{k+1} = T(V_k, \Delta_k)$.
   2.2 If $\|V_{k+1} - V_k\| \leq \epsilon\cdot\frac{1-\gamma}{2\gamma}$, exit the loop. Otherwise, $k \leftarrow k+1$.

The parts of (1) and (2) of the following theorem establish the similar bounds on the performance of VSI to VI's and (3) shows that VSI terminates in a finite number of iterations. That is, VSI preserves the main convergence properties of VI. The part (1) states that $\{V_k\}$ converges to V^* with a linear convergence rate of γ and the part (2) states that the policy π_k defined with V_{k+1} is ϵ-optimal. In addition, the part (4) establishes that $V_{k+1}(x)$ is lower bounded by $\max_{\pi\in\Delta_k}V^\pi(x)$ for all $x \in X$ so that $\|V^* - V_{k+1}\| \leq \max_{x\in X}|V^*(x) - \max_{\pi\in\Delta_k}V^\pi(x)|$. We will further investigate the usefulness of this property later (cf., Theorems 3 and 4). Finally, by the part (5), VSI converges to V^* no slower than VI in terms of the number of iterations.

Theorem 2. *For the sequence $\{V_k\}$ generated by VSI, and the policy π_k defined such that for all $x \in X$,*

$$\pi_k(x) \in \arg\max_{a\in A(x)}\left(R(x,a)+\gamma\sum_{y\in X}P^a_{xy}V_{k+1}(y)\right),$$

and the sequence $\{U_k\}$ generated by VI, the following holds for any $\{\Delta_k\}$:

1. *If $V_0(x) = 0$ for all $x \in X$, then $\|V^* - V_k\| \le \frac{\gamma^k \max_{x,a} |R(x,a)|}{1-\gamma}$ for $k \ge 0$.*
2. *If $\|V_{k+1} - V_k\| \le \epsilon \cdot \frac{1-\gamma}{2\gamma}$, then $\|V^* - V^{\pi_k}\| \le \epsilon$.*
3. *There exists $N < \infty$ such that $\|V_{k+1} - V_k\| \le \epsilon \cdot \frac{1-\gamma}{2\gamma}$ for all $k \ge N$.*
4. *$V_{k+1}(x) \ge \max_{\pi \in \Delta_k} V^\pi(x)$ for all $x \in X$ and $k \ge 0$.*
5. *If $V_0 = U_0$, then $\|V^* - V_k\| \le \|V^* - U_k\|$ for $k \ge 0$.*

Proof. Throughout the proof, fix any $\{\Delta_k\}$. The statement (1) directly comes from the contraction property of T stated in Lemma 1. $\|V^* - V_k\| = \|T(V^*, \Delta_{k-1}) - T(V_{k-1}, \Delta_{k-1})\| \le \gamma \|V^* - V_{k-1}\| \le \gamma^2 \|V^* - V_{k-2}\| \le \cdots \le \gamma^k \|V^* - V_0\| \le \gamma^k \max_{x,a} |R(x,a)|/(1-\gamma)$ with $V_0(x) = 0$ for all $x \in X$.

For the proof of (2), we will use the contraction mapping $L_\pi : B(X) \to B(X)$ defined as $L_\pi(u)(x) = R(x, \pi(x)) + \gamma \sum_{y \in X} P_{xy}^{\pi(x)} u(y)$ for $u \in B(X)$, $x \in X$, and $\pi \in \Pi$. By using the triangular inequality then, we have that $\|V^{\pi_k} - V^*\| \le \|V^{\pi_k} - V_{k+1}\| + \|V_{k+1} - V^*\|$. For the first term of $\|V^{\pi_k} - V_{k+1}\|$ in the inequality,

$$\begin{aligned}
\|V^{\pi_k} - V_{k+1}\| &\le \|L_{\pi_k}(V^{\pi_k}) - L(V_{k+1})\| + \|L(V_{k+1}) - V_{k+1}\| \\
&= \|L_{\pi_k}(V^{\pi_k}) - L_{\pi_k}(V_{k+1})\| + \|L(V_{k+1}) - V_{k+1}\| \\
&\le \|L_{\pi_k}(V^{\pi_k}) - L_{\pi_k}(V_{k+1})\| + \|L(V_{k+1}) - L(V_k)\| \\
&\quad \text{because } V_{k+1} \ge L(V_k) \\
&\le \gamma \|V^{\pi_k} - V_{k+1}\| + \gamma \|V_{k+1} - V_k\|.
\end{aligned}$$

Therefore, we have that $\|V^{\pi_k} - V_{k+1}\| \le \frac{\gamma}{1-\gamma} \|V_{k+1} - V_k\|$. For the second term of $\|V_{k+1} - V^*\|$,

$$\begin{aligned}
\|V_{k+1} - V^*\| &\le \|V^* - L(V_{k+1})\| + \|L(V_{k+1}) - V_{k+1}\| \\
&\le \gamma \|V^* - V_{k+1}\| + \gamma \|V_{k+1} - V_k\| \\
&\le \frac{\gamma}{1-\gamma} \|V_{k+1} - V_k\|.
\end{aligned}$$

It follows that $\|V^{\pi_k} - V^*\| \le \frac{2\gamma}{1-\gamma} \|V_{k+1} - V_k\| \le \epsilon$.

The part (3) comes partly from the proof of (1). We showed that $\|V^* - V_k\| \le \gamma^k \|V^* - V_0\|$. Because V^* and V_0 are bounded, for any $\alpha > 0$, there exists $N < \infty$ such that for all $k \ge N$, $\|V^* - V_k\| \le \alpha$. Then with $V^* + c := V^*(x) + c, \forall x \in X$ and $c \in \mathbb{R}$, $V_{k+1} = T(V_k, \Delta_k) \le T(V^* + \alpha, \Delta_k) = T(V^*, \Delta_k) + \alpha\gamma = V^* + \alpha\gamma$ and similarly $V_{k+1} \ge V^* - \alpha\gamma$. That is, $\|V_{k+1} - V^*\| \le \alpha\gamma$ for all $k \ge N$. Therefore, by setting $\alpha = \epsilon(1-\gamma)/(2\gamma(1+\gamma))$, we have that $\|V_{k+1} - V_k\| \le \|V_{k+1} - V^*\| + \|V^* - V_k\| \le \alpha(1+\gamma) = \epsilon(1-\gamma)/(2\gamma)$ for all $k \ge N$.

The statement (4) follows from $V_{k+1} = T(V_k, \Delta_k) \ge L_\pi(V^\pi) = V^\pi$ for any $\pi \in \Delta_k$. Finally, the statement (5) can be proven by showing that $V_k \ge U_k$ for all k by the induction on k.

Because VSI converges with any sequence of $\{\Delta_k\}$, it provides a considerable freedom of designing convergent variants of VI. In the next subsection, we consider an example of $\{\Delta_k\}$ where Δ_k contains the value functions of randomly generated policies. This let us to establish a probabilistic exponential convergence rate of the resulting algorithm.

2.2. Example: value set iteration with policy switching

We first define the *policy switching* policy $\pi_{\text{ps}}(\Delta)$ for a given nonempty $\Delta \in \mathcal{P}(\Pi)$ such that for all $x \in X$,

$$\pi_{\text{ps}}(\Delta)(x) \in \left\{ \pi(x) | \pi \in \arg\max_{\pi' \in \Delta} V^{\pi'}(x) \right\}.$$

It is known that $\pi_{\text{ps}}(\Delta)$ improves all policies in Δ (Chang et al., 2013), i.e., $V^{\pi_{\text{ps}}(\Delta)}(x) \ge \max_{\pi \in \Delta} V^\pi(x)$ for all $x \in X$. Then the sequence $\{\Delta_k\}$ is selected such that for $k \ge 0$,

$$\Delta_k = \{\pi_{\text{ps}}(\Delta_{k-1})\} \cup \Psi_k,$$

where Ψ_k contains $N \ge 1$ policies independently sampled from a fixed distribution d over Π and possibly additional policies chosen in Π and $\{\pi_{\text{ps}}(\Delta_{-1})\} := \emptyset$.

VSI with policy switching (VSI–PS)

1. **Initialization**: Select $\epsilon > 0$. Set $k = 0$ and $\{\pi_{\text{ps}}(\Delta_{-1})\} := \emptyset$. Select any $V_0 \in B(X)$, $N \ge 1$, and an arbitrary distribution d over Π.
2. **Loop**:
   2.1 $\Delta_k \leftarrow \{\pi_{\text{ps}}(\Delta_{k-1})\} \cup \Psi_k$ where Ψ_k contains N policies independently sampled from d and possibly additional policies chosen in Π.
   2.2 Obtain $V_{k+1} = T(V_k, \Delta_k)$.
   2.3 If $\|V_{k+1} - V_k\| \le \epsilon \cdot \frac{1-\gamma}{2\gamma}$, exit the loop. Otherwise, $k \leftarrow k+1$.

Theorem 3. *For the sequence $\{V_k\}$ generated by VSI–PS, suppose that $V_0 = V^\pi$ for some $\pi \in \Pi$. Then for all $k \ge 0$ and $x \in X$,*

$$\max_{\pi \in \bigcup_{k'=0}^k \Delta_{k'}} V^\pi(x) \le V_{k+1}(x) \le V^*(x).$$

Proof. If $V_0 = V^\pi$ for some $\pi \in \Pi$, it is obvious that $V^* \ge V_0$. Assume that $V^* \ge V_k$. Then $V_{k+1} = T(V_k, \Delta_k) \le T(V^*, \Delta_k) = V^*$. Therefore for all k, $V_k \le V^*$.

From Theorem 2(3), $\max_{\pi \in \Delta_k} V^\pi(x) \le V_{k+1}(x)$ for all $x \in X$. Because $\pi_{\text{ps}}(\Delta_{k-1}) \in \Delta_k$, $\max_{\pi \in \Delta_k \cup \Delta_{k-1}} V^\pi(x) \le V_{k+1}(x)$. Continuing this way, we have that for all $x \in X$,

$$\max_{\pi \in \bigcup_{k'=0}^k \Delta_{k'}} V^\pi(x) \le V_{k+1}(x).$$

Note that all of the results in Theorem 2 apply to VSI–PS and in particular, it follows from the above theorem that for $k \ge 0$,

$$\|V^* - V_{k+1}\| \le \min\left\{ \frac{2\gamma^{k+1} \max_{x,a} |R(x,a)|}{1-\gamma}, \|V^* - \Phi_k\| \right\}, \tag{1}$$

where $\Phi_k(x) := \max_{\pi \in \bigcup_{k'=0}^k \Delta_{k'}} V^\pi(x)$, $x \in X$.

The following theorem establishes a probabilistic convergence rate of the lower bound function Φ_k in Theorem 3 to V^*. We show that the probability that the expected value of any sampled policy from d with respect to an initial state distribution is greater than that of Φ_k converges to zero with $O((N+1)^{-k})$ rate.

Theorem 4. *Consider a random variable Z defined on Π whose probability distribution is given by d. For a given initial state distribution δ over X, let $V_\delta^\pi := \sum_{x \in X} V^\pi(x)\delta(x)$, $\pi \in \Pi$. Then for $\Delta_0, \Delta_1, \ldots, \Delta_k, \Delta_{k+1}$ generated by VSI–PS,*

$$\Pr\left\{ V_\delta^Z > \sum_{x \in X} \Phi_{k+1}(x)\delta(x) \right\} \le \left(\frac{1}{N+1} \right)^{k+1}$$

for any δ and any d and $k \ge 0$.

We remark that the term Pr in the statement of the above theorem depends on the distribution d. Different choices of d can lead to different probabilities. However, the bound holds *regardless of the choices of d*.

Proof. The following proof technique is partly based on the proof of Theorem 4 in Chang (2013). Let $\{\psi_1^k, \ldots, \psi_N^k\}$ be the set of N independent sample-policies generated from d at k. We write Ψ_k as the union of $\{\psi_1^k, \ldots, \psi_N^k\}$ and α_k where α_k is the set of all policies that were not obtained by sampling from d at k. Fix any δ.

We first observe that for $k \geq 1$, $V_\delta^{\pi_{\rm ps}(\Delta_k)} \geq \max_{\pi\in\Delta_k} V_\delta^\pi$. It follows that for any $k \geq 1$, the event $\{V_\delta^Z > V_\delta^{\pi_{\rm ps}(\Delta_k)}\}$ implies $\{V_\delta^Z > \max_{\{\pi_{\rm ps}(\Delta_{k-1}),\psi_1^k,\ldots,\psi_N^k\}\cup\alpha_k} V_\delta^\pi\}$. Therefore, we have that

$$\begin{aligned}
\Pr\{V_\delta^Z > V_\delta^{\pi_{\rm ps}(\Delta_k)}\} &\leq \Pr\{V_\delta^Z > \max_{\{\pi_{\rm ps}(\Delta_{k-1}),\psi_1^k,\ldots,\psi_N^k\}\cup\alpha_k} V_\delta^\pi\} \\
&\leq \Pr\{V_\delta^Z > V_\delta^{\pi_{\rm ps}(\Delta_{k-1})}\} \times \Pr\{V_\delta^Z > \max_{\pi\in\alpha_k} V_\delta^\pi\} \\
&\quad \times \Pr\{V_\delta^Z > \max_{\{\psi_1^k,\ldots,\psi_N^k\}} V_\delta^\pi\} \\
&\leq \Pr\{V_\delta^Z > V_\delta^{\pi_{\rm ps}(\Delta_{k-1})}\} \times \Pr\{V_\delta^Z > \max_{\{\psi_1^k,\ldots,\psi_N^k\}} V_\delta^\pi\}.
\end{aligned}$$

By Campi and Calafiore (2009, Proposition 4), for a discrete random variable Y on a finite nonempty set Ω with any probability distribution d over Ω and $f : \Omega \to \Re$, $\Pr\{f(Y) > \max_{i=1,\ldots,N} f(y_i)\} \leq (N+1)^{-1}$, where y_i's are N independent, identically distributed samples of Y from d. Therefore for any d, $\Pr\{V_\delta^Z > \max_{\{\psi_1^k,\ldots,\psi_N^k\}} V_\delta^\pi\} \leq (N+1)^{-1}$ so that for $k \geq 1$ and any d,

$$\begin{aligned}
\Pr\{V_\delta^Z > V_\delta^{\pi_{\rm ps}(\Delta_k)}\} &\leq \frac{1}{N+1} \times \Pr\{V_\delta^Z > V_\delta^{\pi_{\rm ps}(\Delta_{k-1})}\} \\
&\leq \left(\frac{1}{N+1}\right)^k \times \Pr\{V_\delta^Z > V_\delta^{\pi_{\rm ps}(\Delta_0)}\}.
\end{aligned}$$

It follows that for $k \geq 0$ and any d, $\Pr\{V_\delta^Z > V_\delta^{\pi_{\rm ps}(\Delta_k)}\} \leq \left(\frac{1}{N+1}\right)^{k+1}$.

Because $\sum_{x\in X} \Phi_{k+1}(x)\delta(x) \geq V_\delta^{\pi_{\rm ps}(\Delta_k)}$ for $k \geq 0$, this finally implies that

$$\Pr\left\{V_\delta^Z > \sum_{x\in X} \Phi_{k+1}(x)\delta(x)\right\} \leq \left(\frac{1}{N+1}\right)^{k+1}$$

for any d and $k \geq 0$.

As for VSI, VSI–PS is a general framework for combining multiple solution methods and for converting any non-convergent heuristic method into an exact algorithm. Those policies generated by any other methods can be incorporated into $\{\Psi_k\}$. For example, if Gauss–Seidel value iteration (Puterman, 1994) separately generates a sequence of $\{W_k\}$ and $\tilde{\pi}_k$ defined greedily with W_{k+1} is included into Ψ_k of VSI–PS, then the bound of V_{k+1} relative to V^* is obtained by replacing γ with α in (1), where α is the convergence rate of the Gauss–Seidel method, which is less than or equal to γ (Puterman, 1994, Theorems 6.3.4 and 6.3.7). In other words, the convergence property of the method gets fused into VSI–PS. This kind of combination can be also achieved with other methods, e.g., Arruda et al. (2013) and Shlakhter et al. (2010), etc., by generating a sequence of policies from the value functions generated by the methods.

3. Concluding remark

Even if we provided the method of generating a sequence of $\{\Delta_k\}$ by using policy switching within the framework of VSI, the approach is a separate interest as a novel simulation-based algorithm for solving MDPs with a large action space. The complexity of computing the policy switching policy is independent of the size of the action space. Note that the sequence $\{\pi_{\rm ps}(\Delta_k)\}$ will converge to an optimal policy with probability one as $k \to \infty$ if d is selected such that any policy in Π can be generated with a positive probability. We used policy switching for an example of generating $\{\Delta_k\}$. As in the PSI case, we can employ parallel rollout (Chang et al., 2013) instead of policy switching, having another variant of VSI.

As in the PSI case (Chang, 2013), since no structural assumptions on the policy space is available in general, a general probability bound that holds for any choice of d was established in Theorem 4. Note also that the results of Theorem 2 hold for any d. In practice, the simplest choice of d would be the uniform distribution or heuristically designed one which incorporates some structure of the policy space, if available. How to set d related with the performance of VSI–PS is a good future research topic. There exist various "model-based" optimization algorithms that generate a sequence of the probability distributions over the set of the solutions of a given problem, i.e., "learn" the solution space (see, e.g., the references in Hu, Hu, & Chang, 2012). It would be an interesting work to incorporate (intermediate) results of those algorithms into VSI–PS.

This work focuses on developing an exact variant of VI, not touching on the issue of the curse of dimensionality on VI. A great body of works that deal with the issue exists in the MDP literature (see, e.g., Bertsekas & Tsitsiklis, 1996, Chang et al., 2013, Powell, 2011 and the references therein) by "approximating" the exactness of VI. Because VSI (or VSI–PS) has a similar structure to VI's with the added component of $\max_{\pi\in\Delta} V^\pi(x), x \in X$, those approximating ideas of VI can be similarly incorporated into VSI (with a proper approximation of $\max_{\pi\in\Delta} V^\pi(x), x \in X$).

Because VI has a lower per-iteration time-complexity than VSI, there would be a trade-off between performance and efficiency. Studying empirically and/or theoretically the performance of VSI compared with VI given a fixed computational budget with various sampling distributions is an important future research work.

References

Arruda, E. F., Ourique, F. O., LaCombe, J., & Almudevar, A. (2013). Accelerating the convergence of value iteration by using partial transition functions. *European Journal of Operational Research*, *229*, 190–198.

Bellman, R. (1957). A Markovian decision process. *Journal of Mathematics and Mechanics*, *6*, 679–684.

Bertsekas, D. P., & Tsitsiklis, J. N. (1996). *Neuro-dynamic programming*. Belmont, MA: Athena Scientific.

Blondel, V. D., & Tsitsiklis, J. N. (2000). A survey of computational complexity results in systems and control. *Automatica*, *36*, 1249–1274.

Calafiore, G. C. (2010). Random convex programs. *SIAM Journal on Optimization*, *20*, 3427–3464.

Calafiore, G. C., & Campi, M. C. (2006). The scenario approach to robust control design. *IEEE Transactions on Automatic Control*, *51*, 742–753.

Campi, M. C., & Calafiore, G. C. (2009). Notes on the scenario design approach. *IEEE Transactions on Automatic Control*, *54*, 382–385.

Chang, H. S. (2013). Policy set iteration for Markov decision processes. *Automatica*, *49*, 3687–3689.

Chang, H. S., Hu, J., Fu, M. C., & Marcus, S. I. (2013). *Simulation-based algorithms for Markov decision processes*.

Herzberg, M., & Yechiali, U. (1994). Accelerating procedures of the value iteration algorithm for discounted Markov decision processes, based on a one-step lookahead analysis. *Operations Research*, *42*, 940–946.

Hu, J., Hu, P., & Chang, H. S. (2012). A stochastic approximation framework for a class of randomized optimization algorithms. *IEEE Transactions on Automatic Control*, *57*, 165–178.

Powell, W. B. (2011). *Approximate dynamic programming: solving the curses of dimensionality*. Wiley.

Puterman, M. L. (1994). *Markov decision processes: discrete stochastic dynamic programming*. New York: Wiley.

Shlakhter, O., Lee, C., Khmelev, D., & Javer, N. (2010). Acceleration operators in the value iteration algorithms for Markov decision processes. *Operations Research*, *58*, 193–202.