On the Use of Non-Stationary Policies for Stationary Infinite-Horizon Markov Decision Processes

Bruno Scherrer
Inria, Villers-lès-Nancy, F-54600, France
bruno.scherrer@inria.fr

Boris Lesner
Inria, Villers-lès-Nancy, F-54600, France
boris.lesner@inria.fr

Abstract

We consider infinite-horizon stationary γ-discounted Markov Decision Processes, for which it is known that there exists a stationary optimal policy. Using Value and Policy Iteration with some error ϵ at each iteration, it is well-known that one can compute stationary policies that are $\frac{2\gamma}{(1-\gamma)^2}\epsilon$-optimal. After arguing that this guarantee is tight, we develop variations of Value and Policy Iteration for computing non-stationary policies that can be up to $\frac{2\gamma}{1-\gamma}\epsilon$-optimal, which constitutes a significant improvement in the usual situation when γ is close to 1. Surprisingly, this shows that the problem of "computing near-optimal non-stationary policies" is much simpler than that of "computing near-optimal stationary policies".

1 Introduction

Given an infinite-horizon stationary γ-discounted Markov Decision Process [24, 4], we consider approximate versions of the standard Dynamic Programming algorithms, Policy and Value Iteration, that build sequences of value functions v_k and policies π_k as follows

$$\text{Approximate Value Iteration (AVI):} \qquad v_{k+1} \leftarrow Tv_k + \epsilon_{k+1} \tag{1}$$

$$\text{Approximate Policy Iteration (API):} \qquad \left\{ \begin{array}{rcl} v_k & \leftarrow & v_{\pi_k} + \epsilon_k \\ \pi_{k+1} & \leftarrow & \text{any element of } \mathcal{G}(v_k) \end{array} \right. \tag{2}$$

where v_0 and π_0 are arbitrary, T is the Bellman optimality operator, v_{π_k} is the value of policy π_k and $\mathcal{G}(v_k)$ is the set of policies that are greedy with respect to v_k. At each iteration k, the term ϵ_k accounts for a possible approximation of the Bellman operator (for AVI) or the evaluation of v_{π_k} (for API). Throughout the paper, we will assume that error terms ϵ_k satisfy for all k, $\|\epsilon_k\|_\infty \leq \epsilon$ for some $\epsilon \geq 0$. Under this assumption, it is well-known that both algorithms share the following performance bound (see [25, 11, 4] for AVI and [4] for API):

Theorem 1. *For API (resp. AVI), the* loss *due to running policy π_k (resp. any policy π_k in $\mathcal{G}(v_{k-1})$) instead of the optimal policy π_* satisfies*

$$\limsup_{k\to\infty} \|v_* - v_{\pi_k}\|_\infty \leq \frac{2\gamma}{(1-\gamma)^2}\epsilon.$$

The constant $\frac{2\gamma}{(1-\gamma)^2}$ can be very big, in particular when γ is close to 1, and consequently the above bound is commonly believed to be conservative for practical applications. Interestingly, this very constant $\frac{2\gamma}{(1-\gamma)^2}$ appears in many works analyzing AVI algorithms [25, 11, 27, 12, 13, 23, 7, 6, 20, 21, 22, 9], API algorithms [15, 19, 16, 1, 8, 18, 5, 17, 10, 3, 9, 2] and in one of their generalization [26], suggesting that it cannot be improved. Indeed, the bound (and the $\frac{2\gamma}{(1-\gamma)^2}$ constant) are tight for API [4, Example 6.4], and we will show in Section 3 – to our knowledge, this has never been argued in the literature – that it is also tight for AVI.

Even though the theory of optimal control states that there exists a stationary policy that is optimal, the main contribution of our paper is to show that looking for a *non-stationary* policy (instead of a stationary one) may lead to a much better performance bound. In Section 4, we will show how to deduce such a non-stationary policy from a run of AVI. In Section 5, we will describe two original policy iteration variations that compute non-stationary policies. For all these algorithms, we will prove that we have a performance bound that can be reduced down to $\frac{2\gamma}{1-\gamma}\epsilon$. This is a factor $\frac{1}{1-\gamma}$ better than the standard bound of Theorem 1, which is significant when γ is close to 1. Surprisingly, this will show that the problem of "computing near-optimal non-stationary policies" is much simpler than that of "computing near-optimal stationary policies". Before we present these contributions, the next section begins by precisely describing our setting.

2 Background

We consider an infinite-horizon discounted Markov Decision Process [24, 4] $(\mathcal{S}, \mathcal{A}, P, r, \gamma)$, where $\mathcal{S}$ is a possibly infinite state space, $\mathcal{A}$ is a finite action space, $P(ds'|s,a)$, for all (s,a), is a probability kernel on $\mathcal{S}$, $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is a reward function bounded in max-norm by $R_{\max}$, and $\gamma \in (0,1)$ is a discount factor. A stationary deterministic policy $\pi : \mathcal{S} \to \mathcal{A}$ maps states to actions. We write $r_\pi(s) = r(s,\pi(s))$ and $P_\pi(ds'|s) = P(ds'|s,\pi(s))$ for the immediate reward and the stochastic kernel associated to policy π. The value v_π of a policy π is a function mapping states to the expected discounted sum of rewards received when following π from any state: for all $s \in \mathcal{S}$,

$$v_\pi(s) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_\pi(s_t) \middle| s_0 = s, s_{t+1} \sim P_\pi(\cdot|s_t)\right].$$

The value v_π is clearly bounded by $V_{\max} = R_{\max}/(1-\gamma)$. It is well-known that v_π can be characterized as the unique fixed point of the linear Bellman operator associated to a policy π: $T_\pi : v \mapsto r_\pi + \gamma P_\pi v$. Similarly, the Bellman optimality operator $T : v \mapsto \max_\pi T_\pi v$ has as unique fixed point the optimal value $v_* = \max_\pi v_\pi$. A policy π is greedy w.r.t. a value function v if $T_\pi v = Tv$, the set of such greedy policies is written $\mathcal{G}(v)$. Finally, a policy π_* is optimal, with value $v_{\pi_*} = v_*$, iff $\pi_* \in \mathcal{G}(v_*)$, or equivalently $T_{\pi_*} v_* = v_*$.

Though it is known [24, 4] that there always exists a deterministic stationary policy that is optimal, we will, in this article, consider non-stationary policies and now introduce related notations. Given a sequence $\pi_1, \pi_2, \ldots, \pi_k$ of k stationary policies (this sequence will be clear in the context we describe later), and for any $1 \le m \le k$, we will denote $\pi_{k,m}$ the *periodic non-stationary policy* that takes the first action according to π_k, the second according to π_{k-1}, ..., the m^{th} according to π_{k-m+1} and then starts again. Formally, this can be written as

$$\pi_{k,m} = \pi_k\ \pi_{k-1}\ \cdots\ \pi_{k-m+1}\ \pi_k\ \pi_{k-1}\ \cdots \pi_{k-m+1} \cdots$$

It is straightforward to show that the value $v_{\pi_{k,m}}$ of this periodic non-stationary policy $\pi_{k,m}$ is the unique fixed point of the following operator:

$$T_{k,m} = T_{\pi_k}\ T_{\pi_{k-1}}\ \cdots\ T_{\pi_{k-m+1}}.$$

Finally, it will be convenient to introduce the following discounted kernel:

$$\Gamma_{k,m} = (\gamma P_{\pi_k})(\gamma P_{\pi_{k-1}}) \cdots (\gamma P_{\pi_{k-m+1}}).$$

In particular, for any pair of values v and v', it can easily be seen that $T_{k,m}v - T_{k,m}v' = \Gamma_{k,m}(v - v')$.

3 Tightness of the performance bound of Theorem 1

The bound of Theorem 1 is tight for API in the sense that there exists an MDP [4, Example 6.4] for which the bound is reached. To the best of our knowledge, a similar argument has never been provided for AVI in the literature. It turns out that the MDP that is used for showing the tightness for API also applies to AVI. This is what we show in this section.

Example 1. *Consider the γ-discounted deterministic MDP from [4, Example 6.4] depicted on Figure 1. It involves states $1, 2, \ldots$. In state 1 there is only one self-loop action with zero reward, for each state $i > 1$ there are two possible choices: either* move *to state $i-1$ with zero reward or* stay

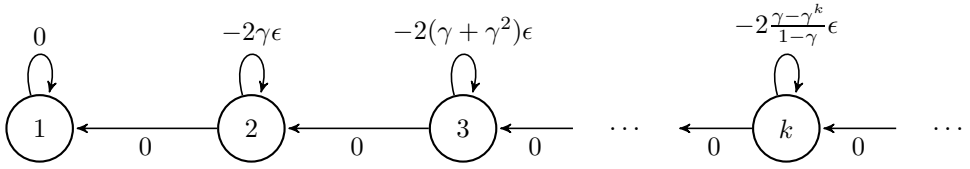

Figure 1: The deterministic MDP for which the bound of Theorem 1 is tight for Value and Policy Iteration.

with reward $r_i = -2\frac{\gamma-\gamma^i}{1-\gamma}\epsilon$ *with* $\epsilon \geq 0$*. Clearly the optimal policy in all states* $i > 1$ *is to move to* $i-1$ *and the optimal value function* v_* *is* 0 *in all states.*

Starting with $v_0 = v_*$*, we are going to show that for all iterations* $k \geq 1$ *it is possible to have a policy* $\pi_{k+1} \in \mathcal{G}(v_k)$ *which moves in every state but* $k+1$ *and thus is such that* $v_{\pi_{k+1}}(k+1) = \frac{r_{k+1}}{1-\gamma} = -2\frac{\gamma-\gamma^{k+1}}{(1-\gamma)^2}\epsilon$*, which meets the bound of Theorem 1 when* k *tends to infinity.*

To do so, we assume that the following approximation errors are made at each iteration $k > 0$*:*

$$\epsilon_k(i) = \begin{cases} -\epsilon & \text{if } i = k \\ \epsilon & \text{if } i = k+1 \\ 0 & \text{otherwise} \end{cases}.$$

With this error, we are now going to prove by induction on k *that for all* $k \geq 1$*,*

$$v_k(i) = \begin{cases} -\gamma^{k-1}\epsilon & \text{if } i < k \\ r_k/2 - \epsilon & \text{if } i = k \\ -(r_k/2-\epsilon) & \text{if } i = k+1 \\ 0 & \text{otherwise} \end{cases}.$$

Since $v_0 = 0$ *the best action is clearly to move in every state* $i \geq 2$ *which gives* $v_1 = v_0 + \epsilon_1 = \epsilon_1$ *which establishes the claim for* $k = 1$*.*

Assuming that our induction claim holds for k*, we now show that it also holds for* $k+1$*.*

For the move *action, write* q_k^{m} *its action-value function. For all* $i > 1$ *we have* $q_k^{\mathrm{m}}(i) = 0 + \gamma v_k(i-1)$*, hence*

$$q_k^{\mathrm{m}}(i) = \begin{cases} \gamma(-\gamma^{k-1}\epsilon) & = -\gamma^k\epsilon & \textit{if } i = 2,\ldots,k \\ \gamma(r_k/2-\epsilon) & = r_{k+1}/2 & \textit{if } i = k+1 \\ -\gamma(r_k/2-\epsilon) & = -r_{k+1}/2 & \textit{if } i = k+2 \\ 0 & & \textit{otherwise} \end{cases}.$$

For the stay *action, write* q_k^{s} *its action-value function. For all* $i > 0$ *we have* $q_k^{\mathrm{s}}(i) = r_i + \gamma v_k(i)$*, hence*

$$q_k^{\mathrm{s}}(i) = \begin{cases} r_i + \gamma(-\gamma^{k-1}\epsilon) & = r_i - \gamma^k\epsilon & \textit{if } i = 1,\ldots,k-1 \\ r_k + \gamma(r_k/2-\epsilon) & = r_k + r_{k+1}/2 & \textit{if } i = k \\ r_{k+1} - r_{k+1}/2 & = r_{k+1}/2 & \textit{if } i = k+1 \\ r_{k+2} + \gamma 0 & = r_{k+2} & \textit{if } i = k+2 \\ 0 & & \textit{otherwise} \end{cases}.$$

First, only the stay *action is available in state* 1*, hence, since* $r_0 = 0$ *and* $\epsilon_{k+1}(1) = 0$*, we have* $v_{k+1}(1) = q_k^{\mathrm{s}}(1) + \epsilon_{k+1}(1) = -\gamma^k\epsilon$*, as desired. Second, since* $r_i < 0$ *for all* $i > 1$ *we have* $q_k^{\mathrm{m}}(i) > q_k^{\mathrm{s}}(i)$ *for all these states but* $k+1$ *where* $q_k^{\mathrm{m}}(k+1) = q_k^{\mathrm{s}}(k+1) = r_{k+1}/2$*. Using the fact that* $v_{k+1} = \max(q_k^{\mathrm{m}}, q_k^{\mathrm{s}}) + \epsilon_{k+1}$ *gives the result for* v_{k+1}*.*

The fact that for $i > 1$ *we have* $q_k^{\mathrm{m}}(i) \geq q_k^{\mathrm{s}}(i)$ *with equality only at* $i = k+1$ *implies that there exists a policy* π_{k+1} *greedy for* v_k *which takes the optimal* move *action in all states but* $k+1$ *where the* stay *action has the same value, leaving the algorithm the possibility of choosing the suboptimal* stay *action in this state, yielding a value* $v_{\pi_{k+1}}(k+1)$*, matching the upper bound as* k *goes to infinity.*

Since Example 1 shows that the bound of Theorem 1 is tight, improving performance bounds imply to modify the algorithms. The following sections of the paper shows that considering non-stationary policies instead of stationary policies is an interesting path to follow.

4 Deducing a non-stationary policy from AVI

While AVI (Equation (1)) is usually considered as generating a sequence of values $v_0, v_1, \ldots, v_{k-1}$, it also implicitely produces a sequence[1] of policies $\pi_1, \pi_2, \ldots, \pi_k$, where for $i = 0, \ldots, k-1$, $\pi_{i+1} \in \mathcal{G}(v_i)$. Instead of outputing only the last policy π_k, we here simply propose to output the periodic non-stationary policy $\pi_{k,m}$ that loops over the last m generated policies. The following theorem shows that it is indeed a good idea.

Theorem 2. *For all iteration k and m such that $1 \leq m \leq k$, the loss of running the non-stationary policy $\pi_{k,m}$ instead of the optimal policy π_* satisfies:*

$$\|v_* - v_{\pi_{k,m}}\|_\infty \leq \frac{2}{1-\gamma^m}\left(\frac{\gamma - \gamma^k}{1-\gamma}\epsilon + \gamma^k \|v_* - v_0\|_\infty\right).$$

When $m = 1$ and k tends to infinity, one exactly recovers the result of Theorem 1. For general m, this new bound is a factor $\frac{1-\gamma^m}{1-\gamma}$ better than the standard bound of Theorem 1. The choice that optimizes the bound, $m = k$, and which consists in looping over all the policies generated *from the very start*, leads to the following bound:

$$\|v_* - v_{\pi_{k,k}}\|_\infty \leq 2\left(\frac{\gamma}{1-\gamma} - \frac{\gamma^k}{1-\gamma^k}\right)\epsilon + \frac{2\gamma^k}{1-\gamma^k}\|v_* - v_0\|_\infty,$$

that tends to $\frac{2\gamma}{1-\gamma}\epsilon$ when k tends to ∞.

The rest of the section is devoted to the proof of Theorem 2. An important step of our proof lies in the following lemma, that implies that for sufficiently big m, $v_k = Tv_{k-1} + \epsilon_k$ is a rather good approximation (of the order $\frac{\epsilon}{1-\gamma}$) of the value $v_{\pi_{k,m}}$ of the non-stationary policy $\pi_{k,m}$ (whereas in general, it is a much poorer approximation of the value v_{π_k} of the last stationary policy π_k).

Lemma 1. *For all m and k such that $1 \leq m \leq k$,*

$$\|Tv_{k-1} - v_{\pi_{k,m}}\|_\infty \leq \gamma^m \|v_{k-m} - v_{\pi_{k,m}}\|_\infty + \frac{\gamma - \gamma^m}{1-\gamma}\epsilon.$$

Proof of Lemma 1. The value of $\pi_{k,m}$ satisfies:

$$v_{\pi_{k,m}} = T_{\pi_k} T_{\pi_{k-1}} \cdots T_{\pi_{k-m+1}} v_{\pi_{k,m}}. \tag{3}$$

By induction, it can be shown that the sequence of values generated by AVI satisfies:

$$T_{\pi_k} v_{k-1} = T_{\pi_k} T_{\pi_{k-1}} \cdots T_{\pi_{k-m+1}} v_{k-m} + \sum_{i=1}^{m-1} \Gamma_{k,i} \epsilon_{k-i}. \tag{4}$$

By substracting Equations (4) and (3), one obtains:

$$Tv_{k-1} - v_{\pi_{k,m}} = T_{\pi_k} v_{k-1} - v_{\pi_{k,m}} = \Gamma_{k,m}(v_{k-m} - v_{\pi_{k,m}}) + \sum_{i=1}^{m-1} \Gamma_{k,i}\epsilon_{k-i}$$

and the result follows by taking the norm and using the fact that for all i, $\|\Gamma_{k,i}\|_\infty = \gamma^i$. □

We are now ready to prove the main result of this section.

Proof of Theorem 2. Using the fact that T is a contraction in max-norm, we have:

$$\begin{aligned}
\|v_* - v_k\|_\infty &= \|v_* - Tv_{k-1} + \epsilon_k\|_\infty \\
&\leq \|Tv_* - Tv_{k-1}\|_\infty + \epsilon \\
&\leq \gamma \|v_* - v_{k-1}\|_\infty + \epsilon.
\end{aligned}$$

[1] A given sequence of value functions may induce many sequences of policies since more than one greedy policy may exist for one particular value function. Our results holds for all such possible choices of greedy policies.

Then, by induction on k, we have that for all $k \geq 1$,

$$\|v_* - v_k\|_\infty \leq \gamma^k \|v_* - v_0\|_\infty + \frac{1-\gamma^k}{1-\gamma}\epsilon. \tag{5}$$

Using Lemma 1 and Equation (5) twice, we can conclude by observing that

$$\begin{aligned}
\|v_* - v_{\pi_{k,m}}\|_\infty &\leq \|Tv_* - Tv_{k-1}\|_\infty + \|Tv_{k-1} - v_{\pi_{k,m}}\|_\infty \\
&\leq \gamma\|v_* - v_{k-1}\|_\infty + \gamma^m \|v_{k-m} - v_{\pi_{k,m}}\|_\infty + \frac{\gamma - \gamma^m}{1-\gamma}\epsilon \\
&\leq \gamma\left(\gamma^{k-1}\|v_* - v_0\|_\infty + \frac{1-\gamma^{k-1}}{1-\gamma}\epsilon\right) \\
&\quad + \gamma^m\left(\|v_{k-m} - v_*\|_\infty + \|v_* - v_{\pi_{k,m}}\|_\infty\right) + \frac{\gamma - \gamma^m}{1-\gamma}\epsilon \\
&\leq \gamma^k\|v_* - v_0\|_\infty + \frac{\gamma - \gamma^k}{1-\gamma}\epsilon \\
&\quad + \gamma^m\left(\gamma^{k-m}\|v_* - v_0\|_\infty + \frac{1-\gamma^{k-m}}{1-\gamma}\epsilon + \|v_* - v_{\pi_{k,m}}\|_\infty\right) + \frac{\gamma - \gamma^m}{1-\gamma}\epsilon \\
&= \gamma^m\|v_* - v_{\pi_{k,m}}\|_\infty + 2\gamma^k\|v_* - v_0\|_\infty + \frac{2(\gamma - \gamma^k)}{1-\gamma}\epsilon \\
&\leq \frac{2}{1-\gamma^m}\left(\frac{\gamma - \gamma^k}{1-\gamma}\epsilon + \gamma^k\|v_* - v_0\|_\infty\right).
\end{aligned}$$

□

5 API algorithms for computing non-stationary policies

We now present similar results that have a Policy Iteration flavour. Unlike in the previous section where only the output of AVI needed to be changed, improving the bound for an API-like algorithm is slightly more involved. In this section, we describe and analyze two API algorithms that output non-stationary policies with improved performance bounds.

API with a non-stationary policy of growing period Following our findings on non-stationary policies AVI, we consider the following variation of API, where at each iteration, instead of computing the value of the last stationary policy π_k, we compute that of the periodic non-stationary policy $\pi_{k,k}$ that loops over all the policies $\pi_1, \ldots, \pi_k$ generated *from the very start*:

$$\begin{aligned}
v_k &\leftarrow v_{\pi_{k,k}} + \epsilon_k \\
\pi_{k+1} &\leftarrow \text{any element of } \mathcal{G}(v_k)
\end{aligned}$$

where the initial (stationary) policy $\pi_{1,1}$ is chosen arbitrarily. Thus, iteration after iteration, the non-stationary policy $\pi_{k,k}$ is made of more and more stationary policies, and this is why we refer to it as having a growing period. We can prove the following performance bound for this algorithm:

Theorem 3. *After k iterations, the loss of running the non-stationary policy $\pi_{k,k}$ instead of the optimal policy π_* satisfies:*

$$\|v_* - v_{\pi_{k,k}}\|_\infty \leq \frac{2(\gamma - \gamma^k)}{1-\gamma}\epsilon + \gamma^{k-1}\|v_* - v_{\pi_{1,1}}\|_\infty + 2(k-1)\gamma^k V_{\max}.$$

When k tends to infinity, this bound tends to $\frac{2\gamma}{1-\gamma}\epsilon$, and is thus again a factor $\frac{1}{1-\gamma}$ better than the original API bound.

Proof of Theorem 3. Using the facts that $T_{k+1,k+1}v_{\pi_{k,k}} = T_{\pi_{k+1}}T_{k,k}v_{\pi_{k,k}} = T_{\pi_{k+1}}v_{\pi_{k,k}}$ and $T_{\pi_{k+1}}v_k \geq T_{\pi_*}v_k$ (since $\pi_{k+1} \in \mathcal{G}(v_k)$), we have:

$$\begin{aligned}
&v_* - v_{\pi_{k+1,k+1}} \\
&= T_{\pi_*}v_* - T_{k+1,k+1}v_{\pi_{k+1,k+1}} \\
&= T_{\pi_*}v_* - T_{\pi_*}v_{\pi_{k,k}} + T_{\pi_*}v_{\pi_{k,k}} - T_{k+1,k+1}v_{\pi_{k,k}} + T_{k+1,k+1}v_{\pi_{k,k}} - T_{k+1,k+1}v_{\pi_{k+1,k+1}} \\
&= \gamma P_{\pi_*}(v_* - v_{\pi_{k,k}}) + T_{\pi_*}v_{\pi_{k,k}} - T_{\pi_{k+1}}v_{\pi_{k,k}} + \Gamma_{k+1,k+1}(v_{\pi_{k,k}} - v_{\pi_{k+1,k+1}}) \\
&= \gamma P_{\pi_*}(v_* - v_{\pi_{k,k}}) + T_{\pi_*}v_k - T_{\pi_{k+1}}v_k + \gamma(P_{\pi_{k+1}} - P_{\pi_*})\epsilon_k + \Gamma_{k+1,k+1}(v_{\pi_{k,k}} - v_{\pi_{k+1,k+1}}) \\
&\leq \gamma P_{\pi_*}(v_* - v_{\pi_{k,k}}) + \gamma(P_{\pi_{k+1}} - P_{\pi_*})\epsilon_k + \Gamma_{k+1,k+1}(v_{\pi_{k,k}} - v_{\pi_{k+1,k+1}}).
\end{aligned}$$

By taking the norm, and using the facts that $\|v_{\pi_{k,k}}\|_\infty \leq V_{\max}$, $\|v_{\pi_{k+1,k+1}}\|_\infty \leq V_{\max}$, and $\|\Gamma_{k+1,k+1}\|_\infty = \gamma^{k+1}$, we get:

$$\|v_* - v_{\pi_{k+1,k+1}}\|_\infty \leq \gamma\|v_* - v_{\pi_{k,k}}\|_\infty + 2\gamma\epsilon + 2\gamma^{k+1}V_{\max}.$$

Finally, by induction on k, we obtain:

$$\|v_* - v_{\pi_{k,k}}\|_\infty \leq \frac{2(\gamma - \gamma^k)}{1-\gamma}\epsilon + \gamma^{k-1}\|v_* - v_{\pi_{1,1}}\|_\infty + 2(k-1)\gamma^k V_{\max}. \qquad \square$$

Though it has an improved asymptotic performance bound, the API algorithm we have just described has two (related) drawbacks: 1) its finite iteration bound has a somewhat unsatisfactory term of the form $2(k-1)\gamma^k V_{\max}$, and 2) even when there is no error (when $\epsilon = 0$), we cannot guarantee that, similarly to standard Policy Iteration, it generates a sequence of policies of increasing values (it is easy to see that in general, we do not have $v_{\pi_{k+1,k+1}} \geq v_{\pi_{k,k}}$). These two points motivate the introduction of another API algorithm.

API with a non-stationary policy of fixed period We consider now another variation of API parameterized by $m \geq 1$, that iterates as follows for $k \geq m$:

$$\begin{aligned}
v_k &\leftarrow v_{\pi_{k,m}} + \epsilon_k \\
\pi_{k+1} &\leftarrow \text{any element of } \mathcal{G}(v_k)
\end{aligned}$$

where the initial non-stationary policy $\pi_{m,m}$ is built from a sequence of m arbitrary stationary policies $\pi_1, \pi_2, \cdots, \pi_m$. Unlike the previous API algorithm, the non-stationary policy $\pi_{k,m}$ here only involves the last m greedy stationary policies instead of all of them, and is thus of fixed period. This is a strict generalization of the standard API algorithm, with which it coincides when $m = 1$. For this algorithm, we can prove the following performance bound:

Theorem 4. *For all m, for all $k \geq m$, the loss of running the non-stationary policy $\pi_{k,m}$ instead of the optimal policy π_* satisfies:*

$$\|v_* - v_{\pi_{k,m}}\|_\infty \leq \gamma^{k-m}\|v_* - v_{\pi_{m,m}}\|_\infty + \frac{2(\gamma - \gamma^{k+1-m})}{(1-\gamma)(1-\gamma^m)}\epsilon.$$

When $m = 1$ and k tends to infinity, we recover exactly the bound of Theorem 1. When $m > 1$ and k tends to infinity, this bound coincides with that of Theorem 2 for our non-stationary version of AVI: it is a factor $\frac{1-\gamma^m}{1-\gamma}$ better than the standard bound of Theorem 1.

The rest of this section develops the proof of this performance bound. A central argument of our proof is the following lemma, which shows that similarly to the standard API, our new algorithm has an (approximate) policy improvement property.

Lemma 2. *At each iteration of the algorithm, the value $v_{\pi_{k+1,m}}$ of the non-stationary policy*

$$\pi_{k+1,m} = \pi_{k+1}\ \pi_k\ \ldots\ \pi_{k+2-m}\ \pi_{k+1}\ \pi_k\ \ldots\ \pi_{k-m+2}\ldots$$

cannot be much worse than the value $v_{\pi'_{k,m}}$ of the non-stationary policy

$$\pi'_{k,m} = \pi_{k-m+1}\ \pi_k\ \ldots\ \pi_{k+2-m}\ \pi_{k-m+1}\ \pi_k\ \ldots\ \pi_{k-m+2}\ldots$$

in the precise following sense:

$$v_{\pi_{k+1,m}} \geq v_{\pi'_{k,m}} - \frac{2\gamma}{1-\gamma^m}\epsilon.$$

The policy $\pi'_{k,m}$ differs from $\pi_{k+1,m}$ in that every m steps, it chooses the oldest policy π_{k-m+1} instead of the newest one π_{k+1}. Also $\pi'_{k,m}$ is related to $\pi_{k,m}$ as follows: $\pi'_{k,m}$ takes the first action according to π_{k-m+1} and then runs $\pi_{k,m}$; equivalently, since $\pi_{k,m}$ loops over $\pi_k\pi_{k-1}\dots\pi_{k-m+1}$, $\pi'_{k,m} = \pi_{k-m+1}\pi_{k,m}$ can be seen as a 1-step right rotation of $\pi_{k,m}$. When there is no error (when $\epsilon = 0$), this shows that the new policy $\pi_{k+1,m}$ is better than a "rotation" of $\pi_{k,m}$. When $m = 1$, $\pi_{k+1,m} = \pi_{k+1}$ and $\pi'_{k,m} = \pi_k$ and we thus recover the well-known (approximate) policy improvement theorem for standard API (see for instance [4, Lemma 6.1]).

Proof of Lemma 2. Since $\pi'_{k,m}$ takes the first action with respect to π_{k-m+1} and then runs $\pi_{k,m}$, we have $v_{\pi'_{k,m}} = T_{\pi_{k-m+1}} v_{\pi_{k,m}}$. Now, since $\pi_{k+1} \in \mathcal{G}(v_k)$, we have $T_{\pi_{k+1}} v_k \geq T_{\pi_{k-m+1}} v_k$ and

$$\begin{aligned}
v_{\pi'_{k,m}} - v_{\pi_{k+1,m}} &= T_{\pi_{k-m+1}} v_{\pi_{k,m}} - v_{\pi_{k+1,m}} \\
&= T_{\pi_{k-m+1}} v_k - \gamma P_{\pi_{k-m+1}} \epsilon_k - v_{\pi_{k+1,m}} \\
&\leq T_{\pi_{k+1}} v_k - \gamma P_{\pi_{k-m+1}} \epsilon_k - v_{\pi_{k+1,m}} \\
&= T_{\pi_{k+1}} v_{\pi_{k,m}} + \gamma(P_{\pi_{k+1}} - P_{\pi_{k-m+1}})\epsilon_k - v_{\pi_{k+1,m}} \\
&= T_{\pi_{k+1}} T_{k,m} v_{\pi_{k,m}} - T_{k+1,m} v_{\pi_{k+1,m}} + \gamma(P_{\pi_{k+1}} - P_{\pi_{k-m+1}})\epsilon_k \\
&= T_{k+1,m} T_{\pi_{k-m+1}} v_{\pi_{k,m}} - T_{k+1,m} v_{\pi_{k+1,m}} + \gamma(P_{\pi_{k+1}} - P_{\pi_{k-m+1}})\epsilon_k \\
&= \Gamma_{k+1,m}(T_{\pi_{k-m+1}} v_{\pi_{k,m}} - v_{\pi_{k+1,m}}) + \gamma(P_{\pi_{k+1}} - P_{\pi_{k-m+1}})\epsilon_k \\
&= \Gamma_{k+1,m}(v_{\pi'_{k,m}} - v_{\pi_{k+1,m}}) + \gamma(P_{\pi_{k+1}} - P_{\pi_{k-m+1}})\epsilon_k.
\end{aligned}$$

from which we deduce that:

$$v_{\pi'_{k,m}} - v_{\pi_{k+1,m}} \leq (I - \Gamma_{k+1,m})^{-1}\gamma(P_{\pi_{k+1}} - P_{\pi_{k-m+1}})\epsilon_k$$

and the result follows by using the facts that $\|\epsilon_k\|_\infty \leq \epsilon$ and $\|(I - \Gamma_{k+1,m})^{-1}\|_\infty = \frac{1}{1-\gamma^m}$. □

We are now ready to prove the main result of this section.

Proof of Theorem 4. Using the facts that 1) $T_{k+1,m+1} v_{\pi_{k,m}} = T_{\pi_{k+1}} T_{k,m} v_{\pi_{k,m}} = T_{\pi_{k+1}} v_{\pi_{k,m}}$ and 2) $T_{\pi_{k+1}} v_k \geq T_{\pi_*} v_k$ (since $\pi_{k+1} \in \mathcal{G}(v_k)$), we have for $k \geq m$,

$$\begin{aligned}
&v_* - v_{\pi_{k+1,m}} \\
&= T_{\pi_*} v_* - T_{k+1,m} v_{\pi_{k+1,m}} \\
&= T_{\pi_*} v_* - T_{\pi_*} v_{\pi_{k,m}} + T_{\pi_*} v_{\pi_{k,m}} - T_{k+1,m+1} v_{\pi_{k,m}} + T_{k+1,m+1} v_{\pi_{k,m}} - T_{k+1,m} v_{\pi_{k+1,m}} \\
&= \gamma P_{\pi_*}(v_* - v_{\pi_{k,m}}) + T_{\pi_*} v_{\pi_{k,m}} - T_{\pi_{k+1}} v_{\pi_{k,m}} + \Gamma_{k+1,m}(T_{\pi_{k-m+1}} v_{\pi_{k,m}} - v_{\pi_{k+1,m}}) \\
&\leq \gamma P_{\pi_*}(v_* - v_{\pi_{k,m}}) + T_{\pi_*} v_k - T_{\pi_{k+1}} v_k + \gamma(P_{\pi_{k+1}} - P_{\pi_*})\epsilon_k + \Gamma_{k+1,m}(T_{\pi_{k-m+1}} v_{\pi_{k,m}} - v_{\pi_{k+1,m}}) \\
&\leq \gamma P_{\pi_*}(v_* - v_{\pi_{k,m}}) + \gamma(P_{\pi_{k+1}} - P_{\pi_*})\epsilon_k + \Gamma_{k+1,m}(T_{\pi_{k-m+1}} v_{\pi_{k,m}} - v_{\pi_{k+1,m}}). \qquad (6)
\end{aligned}$$

Consider the policy $\pi'_{k,m}$ defined in Lemma 2. Observing as in the beginning of the proof of Lemma 2 that $T_{\pi_{k-m+1}} v_{\pi_{k,m}} = v_{\pi'_{k,m}}$, Equation (6) can be rewritten as follows:

$$v_* - v_{\pi_{k+1,m}} \leq \gamma P_{\pi_*}(v_* - v_{\pi_{k,m}}) + \gamma(P_{\pi_{k+1}} - P_{\pi_*})\epsilon_k + \Gamma_{k+1,m}(v_{\pi'_{k,m}} - v_{\pi_{k+1,m}}).$$

By using the facts that $v_* \geq v_{\pi_{k,m}}$, $v_* \geq v_{\pi_{k+1,m}}$ and Lemma 2, we get

$$\begin{aligned}
\|v_* - v_{\pi_{k+1,m}}\|_\infty &\leq \gamma\|v_* - v_{\pi_{k,m}}\|_\infty + 2\gamma\epsilon + \frac{\gamma^m(2\gamma\epsilon)}{1-\gamma^m} \\
&= \gamma\|v_* - v_{\pi_{k,m}}\|_\infty + \frac{2\gamma}{1-\gamma^m}\epsilon.
\end{aligned}$$

Finally, we obtain by induction that for all $k \geq m$,

$$\|v_* - v_{\pi_{k,m}}\|_\infty \leq \gamma^{k-m}\|v_* - v_{\pi_{m,m}}\|_\infty + \frac{2(\gamma - \gamma^{k+1-m})}{(1-\gamma)(1-\gamma^m)}\epsilon. \quad \square$$

6 Discussion, conclusion and future work

We recalled in Theorem 1 the standard performance bound when computing an approximately optimal stationary policy with the standard AVI and API algorithms. After arguing that this bound is tight – in particular by providing an original argument for AVI – we proposed three new dynamic programming algorithms (one based on AVI and two on API) that output non-stationary policies for which the performance bound can be significantly reduced (by a factor $\frac{1}{1-\gamma}$).

From a bibliographical point of view, it is the work of [14] that made us think that non-stationary policies may lead to better performance bounds. In that work, the author considers problems with a finite-horizon T for which one computes *non-stationary* policies with performance bounds in $O(T\epsilon)$, and infinite-horizon problems for which one computes *stationary* policies with performance bounds in $O(\frac{\epsilon}{(1-\gamma)^2})$. Using the informal equivalence of the horizons $T \simeq \frac{1}{1-\gamma}$ one sees that non-stationary policies look better than stationary policies. In [14], non-stationary policies are only computed in the context of finite-horizon (and thus non-stationary) problems; the fact that non-stationary policies can also be useful in an infinite-horizon stationary context is to our knowledge completely new.

The best performance improvements are obtained when our algorithms consider periodic non-stationary policies of which the period grows to infinity, and thus require an infinite memory, which may look like a practical limitation. However, in two of the proposed algorithm, a parameter m allows to make a trade-off between the quality of approximation $\frac{2\gamma}{(1-\gamma^m)(1-\gamma)}\epsilon$ and the amount of memory $O(m)$ required. In practice, it is easy to see that by choosing $m = \left\lceil \frac{1}{1-\gamma} \right\rceil$, that is a memory that scales linearly with the horizon (and thus the difficulty) of the problem, one can get a performance bound of[2] $\frac{2\gamma}{(1-e^{-1})(1-\gamma)}\epsilon \leq \frac{3.164\gamma}{1-\gamma}\epsilon$.

We conjecture that our asymptotic bound of $\frac{2\gamma}{1-\gamma}\epsilon$, and the non-asymptotic bounds of Theorems 2 and 4 are tight. The actual proof of this conjecture is left for future work. Important recent works of the literature involve studying performance bounds when the errors are controlled in L_p norms instead of max-norm [19, 20, 21, 1, 8, 18, 17] which is natural when supervised learning algorithms are used to approximate the evaluation steps of AVI and API. Since our proof are based on componentwise bounds like those of the pioneer works in this topic [19, 20], we believe that the extension of our analysis to L_p norm analysis is straightforward. Last but not least, an important research direction that we plan to follow consists in revisiting the many implementations of AVI and API for building stationary policies (see the list in the introduction), turn them into algorithms that look for non-stationary policies and study them precisely analytically as well as empirically.

References

[1] A. Antos, Cs. Szepesvári, and R. Munos. Learning near-optimal policies with Bellman-residual minimization based fitted policy iteration and a single sample path. *Machine Learning*, 71(1):89–129, 2008.

[2] M. Gheshlaghi Azar, V. Gmez, and H.J. Kappen. Dynamic Policy Programming with Function Approximation. In *14th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 15, Fort Lauderdale, FL, USA, 2011.

[3] D.P. Bertsekas. Approximate policy iteration: a survey and some new methods. *Journal of Control Theory and Applications*, 9:310–335, 2011.

[4] D.P. Bertsekas and J.N. Tsitsiklis. *Neuro-Dynamic Programming*. Athena Scientific, 1996.

[5] L. Busoniu, A. Lazaric, M. Ghavamzadeh, R. Munos, R. Babuska, and B. De Schutter. Least-squares methods for Policy Iteration. In M. Wiering and M. van Otterlo, editors, *Reinforcement Learning: State of the Art*. Springer, 2011.

[6] D. Ernst, P. Geurts, and L. Wehenkel. Tree-based batch mode reinforcement learning. *Journal of Machine Learning Research (JMLR)*, 6, 2005.

[2] With this choice of m, we have $m \geq \frac{1}{\log 1/\gamma}$ and thus $\frac{2}{1-\gamma^m} \leq \frac{2}{1-e^{-1}} \leq 3.164$.

[7] E. Even-dar. Planning in pomdps using multiplicity automata. In *Uncertainty in Artificial Intelligence (UAI*, pages 185–192, 2005.

[8] A.M. Farahmand, M. Ghavamzadeh, Cs. Szepesvári, and S. Mannor. Regularized policy iteration. *Advances in Neural Information Processing Systems*, 21:441–448, 2009.

[9] A.M. Farahmand, R. Munos, and Cs. Szepesvári. Error propagation for approximate policy and value iteration (extended version). In *NIPS*, December 2010.

[10] V. Gabillon, A. Lazaric, M. Ghavamzadeh, and B. Scherrer. Classification-based Policy Iteration with a Critic. In *International Conference on Machine Learning (ICML)*, pages 1049–1056, Seattle, États-Unis, June 2011.

[11] G.J. Gordon. Stable Function Approximation in Dynamic Programming. In *ICML*, pages 261–268, 1995.

[12] C. Guestrin, D. Koller, and R. Parr. Max-norm projections for factored MDPs. In *International Joint Conference on Artificial Intelligence*, volume 17-1, pages 673–682, 2001.

[13] C. Guestrin, D. Koller, R. Parr, and S. Venkataraman. Efficient Solution Algorithms for Factored MDPs. *Journal of Artificial Intelligence Research (JAIR)*, 19:399–468, 2003.

[14] S.M. Kakade. *On the Sample Complexity of Reinforcement Learning*. PhD thesis, University College London, 2003.

[15] S.M. Kakade and J. Langford. Approximately Optimal Approximate Reinforcement Learning. In *International Conference on Machine Learning (ICML)*, pages 267–274, 2002.

[16] M.G. Lagoudakis and R. Parr. Least-squares policy iteration. *Journal of Machine Learning Research (JMLR)*, 4:1107–1149, 2003.

[17] A. Lazaric, M. Ghavamzadeh, and R. Munos. Finite-Sample Analysis of Least-Squares Policy Iteration. *To appear in Journal of Machine learning Research (JMLR)*, 2011.

[18] O.A. Maillard, R. Munos, A. Lazaric, and M. Ghavamzadeh. Finite Sample Analysis of Bellman Residual Minimization. In Masashi Sugiyama and Qiang Yang, editors, *Asian Conference on Machine Learpning. JMLR: Workshop and Conference Proceedings*, volume 13, pages 309–324, 2010.

[19] R. Munos. Error Bounds for Approximate Policy Iteration. In *International Conference on Machine Learning (ICML)*, pages 560–567, 2003.

[20] R. Munos. Performance Bounds in Lp norm for Approximate Value Iteration. *SIAM J. Control and Optimization*, 2007.

[21] R. Munos and Cs. Szepesvári. Finite time bounds for sampling based fitted value iteration. *Journal of Machine Learning Research (JMLR)*, 9:815–857, 2008.

[22] M. Petrik and B. Scherrer. Biasing Approximate Dynamic Programming with a Lower Discount Factor. In *Twenty-Second Annual Conference on Neural Information Processing Systems -NIPS 2008*, Vancouver, Canada, 2008.

[23] J. Pineau, G.J. Gordon, and S. Thrun. Point-based value iteration: An anytime algorithm for POMDPs. In *International Joint Conference on Artificial Intelligence*, volume 18, pages 1025–1032, 2003.

[24] M. Puterman. *Markov Decision Processes*. Wiley, New York, 1994.

[25] S. Singh and R. Yee. An Upper Bound on the Loss from Approximate Optimal-Value Functions. *Machine Learning*, 16-3:227–233, 1994.

[26] C. Thiery and B. Scherrer. Least-Squares λ Policy Iteration: Bias-Variance Trade-off in Control Problems. In *International Conference on Machine Learning*, Haifa, Israel, 2010.

[27] J.N. Tsitsiklis and B. Van Roy. Feature-Based Methods for Large Scale Dynamic Programming. *Machine Learning*, 22(1-3):59–94, 1996.