HAL open science

Performance Bounds for Lambda Policy Iteration and Application to the Game of Tetris

Bruno Scherrer

▶ To cite this version:

Bruno Scherrer. Performance Bounds for Lambda Policy Iteration and Application to the Game of Tetris. [Research Report] RR-6348, 2011, pp.53. inria-00185271v3

HAL Id: inria-00185271

https://inria.hal.science/inria-00185271v3

Submitted on 13 Nov 2007 (v3), last revised 11 Oct 2011 (v5)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

inria-00185271, version 3 - 13 Nov 2007

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Performance Bounds for Lambda Policy Iteration

Bruno Scherrer

N° 6348

Novembre 2007

Thème COG

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--6348--FR+ENG

Performance Bounds for Lambda Policy Iteration

Bruno Scherrer

Thème COG — Systèmes cognitifs
Projet Maia

Rapport de recherche n° 6348 — Novembre 2007 — 39 pages

Abstract: We consider the discrete-time infinite-horizon discounted stationary optimal control problem formalized by Markov Decision Processes. We study Lambda Policy Iteration, a family of algorithms parameterized by lambda, originally introduced by Ioffe and Bertsekas. Lambda Policy Iteration generalizes the standard algorithms Value Iteration and Policy Iteration, and is closely related to TD(lambda) introduced by Sutton and Barto. We deepen the original theory developped by Ioffe and Bertsekas by providing convergence rate bounds which generalize standard bounds for Value Iteration described for instance by Puterman. We also develop the theory of this algorithm when it is used in an approximate form. Doing so, we extend and unify the separate analyses developped by Munos for Approximate Value Iteration and Approximate Policy Iteration. The main contribution of this paper is that we show that doing Approximate Lambda Policy Iteration is sound.

Key-words: Optimal Control, Reinforcement Learning, Markov Decision Processes, Analysis of Algorithms, Convergence Rate, Error Bounds, Value Iteration, Policy Iteration, Temporal Differences, Span seminorm

Unité de recherche INRIA Lorraine
LORIA, Technopôle de Nancy-Brabois, Campus scientifique,
615, rue du Jardin Botanique, BP 101, 54602 Villers-Lès-Nancy (France)
Téléphone : +33 3 83 59 30 00 — Télécopie : +33 3 83 27 83 19

Bornes sur les performances de l'algorithme lambda Itérations sur les politiques

Résumé : Nous considérons le problème du contrôle optimal stationnaire à temps discret, à horizon infini, avec le critère actualisé. Nous étudions lambda Itérations sur les politiques, une famille d'algorithmes paramétrées par lambda, originellement proposée par Ioffe et Bertsekas. Lambda Itérations sur les politiques généralise les algorithmes standards Itérations sur les valeurs et Itérations sur les politiques, et est intimiment lié à l'algorithme TD(lambda) proposé par Sutton et Barto. Nous approfondissons l'analyse originelle développée par Ioffe et Bertsekas en décrivant des bornes sur la vitesse de convergence qui généralisent des résultats standards concernant Itérations sur les valeurs décrits par exemple par Puterman. Nous développons également l'analyse de cet algorithme lorsqu'il est utilisé de manière approchée. Ce faisant, nous étendons et unifions les analyses développées séparément par Munos pour les versions approximatives d'Itérations sur les valeurs et d'Itérations sur les politiques. La contribution principale de cet article est de montrer qu'utiliser une version approximative de Lambda Policy Iteration est fondé.

Mots-clés : Contrôle optimal, Apprentissage par renforcement, Analyse d'algorithmes, Vitesse de convergence, Bornes d'erreurs, Processus de décision markovien, Itérations sur les valeurs, Itérations sur les politiques, Différences temporelles, Apprentissage par renforcement, Semi-norme span

Contents

1 Definition of norms and seminorms **4**

2 Markov Decision Processes **5**
2.1 The Stochastic Optimal Control Problem . . . 5
2.2 Value Iteration . . . 6
2.3 Policy Iteration . . . 9
2.4 λ Policy Iteration . . . 11
2.5 Our contributions . . . 15

3 Componentwise Performance bounds for λ Policy Iteration **15**
3.1 Performance bounds for Exact λ Policy Iteration . . . 16
3.2 Performance bounds for Approximate λ Policy Iteration . . . 16

4 Span seminorm Performance bounds for λ Policy Iteration **19**
4.1 From Componentwise bounds to span seminorm bounds . . . 19
4.2 Rates of convergence for Exact λ Policy Iteration . . . 21
4.3 Bounds for Approximate λ Policy Iteration . . . 22

Appendix **25**

A Componentwise bounds with respect to the Bellman residuals **25**
A.1 Bounds with respect to Policy Bellman residual . . . 25
A.2 Bounds with respect to Bellman residual . . . 26

B General componentwise analysis of λ Policy Iteration **27**
B.1 Overview of the analysis of the componentwise loss bound . . . 27
B.2 Proof of Lemma B.1: a relation between the shift and the Bellman residual . . . 29
B.3 Proof of Lemma B.2: a lower bound of the Bellman residual . . . 29
B.4 Proof of Lemma B.3: an upper bound of the distance . . . 30

C Componentwise rate of convergence for Exact λ Policy Iteration **31**
C.1 A bound with respect to the Bellman residual . . . 31
C.2 A bound with respect to the distance . . . 32
C.3 A bound with respect to the distance and the loss of the greedy policy . . . 32

D Asymptotic componentwise loss bounds with respect to the approximation error **33**
D.1 General analysis . . . 33
D.2 When the value converges . . . 35
D.3 When the policy converges . . . 36

E From componentwise bounds to span seminorm bounds **37**
E.1 Proof of lemma 4.1 . . . 37
E.2 Proof of lemma 4.2 . . . 38

References **39**

Introduction

We consider the discrete-time infinite-horizon discounted stationary optimal control problem formalized by Markov Decision Processes. We study λ Policy Iteration, a family of algorithms parameterized by λ, originally introduced by Ioffe and Bertsekas [1]. λ Policy Iteration generalizes the standard algorithms Value Iteration and Policy Iteration, and has some connections with TD(λ) introduced by Sutton & Barto [9]. We deepen the original theory developped by Ioffe and Bertsekas [1] by providing convergence rate bounds which generalize standard bounds for Value Iteration described for instance by Puterman [7]. We also develop the theory of this algorithm when it is used in an approximate form. Doing so, we extend and unify the separate analyses developped by Munos for Approximate Value Iteration [5] and Approximate Policy Iteration [4]. The main contribution of this paper is to show that doing Approximate Lambda Policy Iteration is sound.

1 Definition of norms and seminorms

The analysis we will describe in this article relies on several norms and seminorms which we define here.

Let X be a finite space. In this section, u denotes a real-valued function on X, which can be seen as a vector of dimension $|X|$. Let e denote the vector of which all components are 1. μ denotes a distribution on X. We consider the weighted L_p norm:

$$\|u\|_{p,\mu} := \left(\sum_x \mu(x)|u(x)|^p \right)^{1/p}.$$

We will write $\|.\|_p$ the unweighted L_p norm (i.e. with uniform distribution μ). $\|.\|_\infty$ is the max-norm:

$$\|u\|_\infty := \max_x |u(x)| = \lim_{p\to\infty} \|u\|_p .$$

We write $\text{span}_\infty [.]$ the span seminorm (as for instance defined in [7]):

$$\text{span}_\infty [u] := \max_x u(x) - \min_x u(x).$$

It can be seen that

$$\text{span}_\infty [u] = 2 \min_a \|u - ae\|_\infty .$$

It is thus natural to generalize the span seminorm definition as follows:

$$\text{span}_{p,\mu} [u] := 2 \min_a \|u - ae\|_{p,\mu}$$

It is clear that it is a seminorm (i.e. it is non-negative, it satisfies the triangle inequality and $\text{span}_* [au] = |a|\text{span}_* [u]$). It is not a norm because it is zero for all constant functions.

The error bounds we will derive in this paper are expressed in terms of some span seminorm. The following relations

$$\begin{cases} \text{span}_p [u] & \leq & 2\|u\|_p & \leq & 2\|u\|_\infty \\ \text{span}_{p,\mu} [u] & \leq & 2\|u\|_{p,\mu} & \leq & 2\|u\|_\infty \\ \text{span}_\infty [u] & \leq & 2\|u\|_\infty & & \end{cases} \tag{1}$$

show how to deduce error bounds involving the (more standard) L_p and max norms. Since the span seminorm can be zero for non zero (constant) vectors, there is no relation that would enable us to derive error bounds in span seminorm from a L_p or a max norm. Bounding an error with the span seminorm is in this sense stronger and this constitutes our motivation for using it.

2 Markov Decision Processes

This paper is about the discrete-time infinite-horizon discounted stationary optimal control problem, which we describe now.

2.1 The Stochastic Optimal Control Problem

This paper is about the discrete-time infinite-horizon discounted stationary optimal control problem, which we describe now. We consider a discrete-time dynamic system whose state transition depends on a control. We assume that there is a **state space** X of finite size N. When at state i, the control is chosen from a finite **control space** A. The control $a \in A$ specifies the **transition probability** $p_{ij}(a)$ to the next state j. At the kth iteration, the system is given a reward $\gamma^k r(i, a, j)$ where r is the instantaneous **reward function**, and $0 < \gamma < 1$ is a discount factor. The tuple $\langle X, A, p, r, \gamma \rangle$ is known as a **Markov Decision Process** [7].

We are interested in stationary deterministic policies, that is functions $\pi : X \to A$ which map states into controls[1]. Writing i_k the state at time k, the **value of policy** π at state i is defined as the total expected return while following a policy π from i, that is

$$v^\pi(i) := \lim_{N\to\infty} E_\pi \left[\sum_{k=0}^{N-1} \gamma^k r(i_k, \pi(i_k), i_{k+1}) \middle| i_0 = i \right] \tag{2}$$

where E_π denotes the expectation conditional on the fact that the actions are selected with the policy π. The **optimal value** starting from state i is defined as

$$v_*(i) := \max_\pi v^\pi(i).$$

We write P^π the $N \times N$ stochastic matrix whose elements are $p_{ij}(\pi(i))$ and r^π the vector whose components are $\sum_j p_{ij}(\pi(i)) r(i, \pi(i), j)$. v^π and v_* can be seen as vectors on X. It is well-known that v^π solves the following Bellman equation:

$$v^\pi = r^\pi + \gamma P^\pi v^\pi.$$

v^π is a fixed point of the linear backup operator $\mathcal{T}^\pi v := r^\pi + \gamma P^\pi v$. As P^π is a stochastic matrix, its eigenvalues cannot be greater than 1, and consequently $I - \gamma P^\pi$ is invertible. This implies that

$$v^\pi = (I - \gamma P^\pi)^{-1} r^\pi = \sum_{i=0}^{\infty} (\gamma P^\pi)^i r^\pi. \tag{3}$$

It is also well-known that v_* satisfies the following Bellman equation:

$$v_* = \max_\pi (r^\pi + \gamma P^\pi v_*) = \max_\pi \mathcal{T}^\pi v_*$$

v_* is a fixed point of the nonlinear backup operator $\mathcal{T} v := \max_\pi \mathcal{T}^\pi v$. Once the optimal value v_* is computed, deriving an optimal policy is straightforward. For any value vector v, we call a **greedy policy with respect to the value** v a policy π that satisfies:

$$\pi \in \arg\max_{\pi'} \mathcal{T}^{\pi'} v$$

or equivalently $\mathcal{T}^\pi v = \mathcal{T} v$. We will write, with some abuse of notation[2] greedy(v) any policy that is greedy with respect to v. The notions of optimal value function and greedy policies are fundamental to optimal

[1] Restricting our attention to stationary deterministic policies is not a limitation. Indeed, for the optimality criterion to be defined soon, it can indeed be shown that there exists at least one stationary deterministic policy which is optimal [7].

[2] There might be several policies that are greedy with respect to some value v.

control because of the following property: any policy π_* that is greedy with respect to the optimal value is an **optimal policy** and its value v^{π_*} equals v_*.

It is well-known that the backup operators $\mathcal{T}^\pi$ and $\mathcal{T}$ are γ-contraction mappings with respect to the max-norm. In what follows we only write what this means for the Bellman operator $\mathcal{T}$ but the same holds for $\mathcal{T}^\pi$. Being a γ-contraction mapping for the max-norm means that for all pairs of vectors (v, w),

$$\|\mathcal{T}v - \mathcal{T}w\|_\infty \leq \gamma \|v - w\|_\infty .$$

This implies that the corresponding fixed point exists and is unique and that for any initial vector v_0,

$$\lim_{k\to\infty} (\mathcal{T})^k v_0 = v_*. \tag{4}$$

2.2 Value Iteration

The Value Iteration algorithms for computing the value of a policy π and the value of the optimal policy π_* rely on equation 4. Algorithm 1 provides a description of Value Iteration for computing an optimal policy (replace $\mathcal{T}$ by $\mathcal{T}^\pi$ in it and one gets Value Iteration for computing the value of policy π). In this description,

Algorithm 1 `Value Iteration`

Input: An MDP, an initial value v_0
Output: An (approximately) optimal policy
  $k \leftarrow 0$
  repeat
    $v_{k+1} \leftarrow \mathcal{T}v_k + \epsilon_{k+1}$ // Update the value
    $k \leftarrow k + 1$
  until some stopping criterion
Return greedy(v_k)

we have introduced a term ϵ_k which stands for several possible sources of error at each iteration: this error might be the computer round off, the fact that we use an approximate architecture for representing v, a stochastic approximation of P^{π_k}, etc... or a combination of these. In what follows, when we talk about the *Exact* version of an algorithm, this means that $\epsilon_k = 0$ for all k.

Properties of Exact Value Iteration It is well-known that the contraction property induces some interesting properties for Exact Value Iteration. We have already mentioned that contraction implies the asymptotic convergence (equation 4). It can also be inferred that there is at least a linear rate of convergence: for all reference iteration k_0, and for all $k \geq k_0$

$$\|v_* - v_k\|_\infty \leq \gamma^{k-k_0} \|v_* - v_{k_0}\|_\infty .$$

Even more interestingly, it is possible to derive a performance bound, that is a bound of the difference between the real value of a policy produced by the algorithm and the value of the optimal policy π_* (see for instance [7]). Let π_k denote the policy that is greedy with respect to v_{k-1}. Then, for all reference iteration k_0, and for all $k \geq k_0$,

$$\|v_* - v^{\pi_k}\|_\infty \leq \frac{2\gamma^{k-k_0}}{1-\gamma} \|\mathcal{T}v_{k_0} - v_{k_0}\|_\infty = \frac{2\gamma^{k-k_0}}{1-\gamma} \|v_{k_0+1} - v_{k_0}\|_\infty .$$

This fact is of considerable importance computationally since it provides a stopping criterion: taking $k = k_0 + 1$, we see that if $\|v_{k_0+1} - v_{k_0}\|_\infty < \frac{1-\gamma}{2\gamma}\epsilon$, then $\|v_* - v^{\pi_{k_0+1}}\|_\infty < \epsilon$.

It is somewhat less known that the Bellamn operators $\mathcal{T}$ and $\mathcal{T}^\pi$ are also contraction mapping with respect to the span_∞ seminorm [7]. This means that there exists a variant of the above equation involving the span seminorm instead of the max-norm. For instance, such a fact provides the following stopping criterion [7]:

Proposition 2.1 (Stopping Condition for Exact Value Iteration [7]).
If at some iteration k_0, the difference between two subsequent iterations satisfies

$$span_\infty \left[v_{k_0+1} - v_{k_0}\right] < \frac{1-\gamma}{\gamma}\epsilon$$

then the greedy policy π_{k_0+1} with respect to v_{k_0} is ϵ-optimal: $\|v_* - v^{\pi_{k_0+1}}\|_\infty < \epsilon$.

This latter stopping criterion is better since, from the relation between the span seminorm and the norm (equation 1) it implies the former.

Properties of Approximate Value Iteration When considering large Markov Decision Processes, one cannot usually implement an exact version of Value Iteration. In such a case $\epsilon_k \neq 0$. In general, the algorithm does not converge anymore but it is possible to study its asymptotic behaviour. The most well-known result is due to Bertsekas and Tsitsiklis [2]: If the approximation errors are uniformly bounded $\|\epsilon_k\|_\infty \leq \epsilon$, then the difference between the asymptotic performance of policies π_{k+1} greedy with respect to v_k satisfies

$$\limsup_{k\to\infty} \|v_* - v^{\pi_k}\|_\infty \leq \frac{2\gamma}{(1-\gamma)^2}\epsilon. \tag{5}$$

Munos has recently argued in [4, 5] that, since most supervised learning algorithms (such as least square regression) that are used in practice for approximating each iterate of Value Iteration minimize an empirical approximation in some L_p norm, it would be more interesting to have an analogue of the above result where the approximation error ϵ is expressed in terms of the L_p norm. Munos actually showed how to do this in [5]. The idea is to analyze the *componentwise* asymptotic behaviour of Approximate Value Iteration, from which it is rather easy to derive L_p analysis for any p. Write $P_k = P^{\pi_k}$ the stochastic matrix corresponding to the policy π_k which is greedy with respect to v_{k-1}, P_* the stochastic matrix corresponding to the (unknown) optimal policy π_*.

Lemma 2.2 (Componentwise Asymptotic Performance of Approximate Value Iteration [5]).
The following matrices

$$\begin{aligned} Q_{kj} &:= (1-\gamma)(I-\gamma P_k)^{-1} P_k P_{k-1} ... P_{j+1} \\ Q'_{kj} &:= (1-\gamma)(I-\gamma P_k)^{-1} (P_*)^{k-j} \end{aligned}$$

are stochastic and the asymptotic performance of the policies generated by Approximate Value Iteration satisfies

$$\limsup_{k\to\infty} v_* - v^{\pi_k} \leq \limsup_{k\to\infty} \frac{1}{1-\gamma} \sum_{j=0}^{k-1} \gamma^{k-j} \left[Q_{kj} - Q'_{kj}\right] \epsilon_j.$$

From the above componentwise bound, it is possible[3] to derive the following L_p bounds.

[3] This result is not explicitely stated by Munos in [5], but using a technique of another of his articles [4], it is straightforward to derive from Lemma 2.2. The current paper will anyway generalize this result (in Proposition 4.8 page 23).

Proposition 2.3 (Asymptotic Performance of Approximate Value Iteration I).
Choose any p any distribution μ. Consider the notations of Lemma 2.2 and 3.4. If the approximation errors are uniformly bounded for the following set of norms

$$\forall k \geq j, \quad \|\epsilon_k\|_{p,\frac{1}{2}\mu\left(Q_{jk}+Q'_{jk}\right)} \leq \epsilon$$

then the asymptotic performance of policies generated by Value Iteration satisfies

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

Munos also introduces some **concentration coefficient** [4, 5]: Assume there exists a distribution ν and a real number $C(\nu)$ such that

$$C(\nu) := \max_{i,j,a} \frac{p_{ij}(a)}{\nu(j)}. \tag{6}$$

For instance, if one chooses the uniform law ν, then there always exists such a $C(\nu) \in (1, N)$ where N is the size of the state space. See [4, 5] for more discussion on this coefficient. This concentration coefficient is interesting because it allows to derive the following performance bounds on the max-norm of the loss.

Proposition 2.4 (Asymptotic Performance of Approximate Value Iteration II [5]).
Let $C(\nu)$ be the concentration coefficient defined in equation 6. If the approximation errors are uniformly bounded

$$\|\epsilon_k\|_{p,\nu} \leq \epsilon$$

then the asymptotic performance of the policies generated by Approximate Value Iteration satisfies

$$\limsup_{k\to\infty} \ \|v_* - v^{\pi_k}\|_{\infty} \leq \frac{2\gamma\left(C(\nu)\right)^{1/p}}{(1-\gamma)^2}\epsilon.$$

The main difference between the bounds of Propositions 2.3 and 2.4 and that of Bertsekas and Tsitsiklis (equation 5) is that the approximation error ϵ is controlled by the weighted L_p norm. As $\lim_{p\to\infty} \|.\|_{p,\mu} \leq \|.\|_{\infty}$ and $\left(C(\nu)\right)^{1/p} \overset{p\to\infty}{\longrightarrow} 1$, Munos's results are strictly better.

There is in general no guarantee that AVI converges. AVI is known to converge for specific approximation architectures known as averagers [3] which include state aggregation. Also, convergence may just occur experimentally. Suppose (v_k) tends to some v. Write π the corresponding greedy policy. Note also that (ϵ_k) tends to $v - \mathcal{T}v$, which is known as the **Bellman residual**. The above bounds apply, but in this specific case, they can be improved by a factor $\frac{1}{1-\gamma}$. It is indeed known (e.g. [10]) that

$$\|v_* - v^{\pi}\|_{\infty} \leq \frac{2\gamma}{(1-\gamma)} \|v - \mathcal{T}v\|_{\infty}$$

and, with the same notations as above, Munos derived the analogous better L_p bound [5]:

Corollary 2.5 (Performance of Approximate Value Iteration in case of convergence [5]).
Let $C(\nu)$ be the concentration coefficient defined in equation 6. Suppose (v_k) tends to some v. Write π the corresponding greedy policy. Then

$$\|v_* - v^{\pi}\|_{\infty} \leq \frac{2\gamma\left(C(\nu)\right)^{1/p}}{(1-\gamma)} \|v - \mathcal{T}v\|_{p,\nu}$$

Eventually, let us mention that in [5], Munos also shows some *finer* performance bounds (in weighted L_p norm) using some *finer* concentration coefficients. We won't discuss them in this paper and we recommend the interested reader to go through [5].

2.3 Policy Iteration

Policy Iteration is an alternative method for computing an optimal policy for an infinite-horizon discounted Markov Decision Process. This algorithm is based on the following property: if π is some policy, then any

Algorithm 2 `Policy Iteration`

Input: An MDP, an initial policy π_0
Output: An (approximately) optimal policy
  $k \leftarrow 0$
  repeat
    $v_k \leftarrow (I - \gamma P^{\pi_k})^{-1} r^{\pi_k} + \epsilon_k$ // Estimate the value of π_k
    $\pi_{k+1} \leftarrow \text{greedy}(v_k)$ // Update the policy
    $k \leftarrow k + 1$
  until some stopping criterion
Return π_k

policy π' that is greedy with respect to the value of π, i.e. any π' satisfying $\pi' = \text{greedy}(v^\pi)$, is better than π in the sense that $v^{\pi'} \geq v^\pi$. Policy Iteration exploits this property in order to generate a sequence of policies with increasing values. It is described in Algorithm 2. Note that we use the analytical form of the value of a policy given by equation 3. Also, as for Value Iteration, our description includes a potential error ϵ_k term each time the value of a policy is estimated.

Properties of Exact Policy Iteration When the state space and the control spaces are finite, it is well-known that Exact Policy Iteration converges to an optimal policy π_* in a finite number of iterations. It is known that the rate of convergence is at least linear [7]. If the function $v \mapsto P^{\text{greedy}(v)}$ is Lipschitz, then it can be shown that Policy Iteration has a quadratic convergence [7]. However, we did not find any stopping condition in the literature that is similar to the one of Proposition 2.1.

Properties of Approximate Policy Iteration For problems of interest, one usually uses Policy Iteration in an approximate form, i.e. with $\epsilon_k \neq 0$. Results similar to those we presented for Approximate Value Iteration exist for Approximate Policy Iteration. As soon as there is some error $\epsilon_k \neq 0$, the algorithm does not necessarily converge anymore but there is an analog of equation 5 which is also due to Bertsekas and Tsitsiklis [2]: If the approximation errors are uniformly bounded ($\forall k, \|\epsilon_k\|_\infty \leq \epsilon$), then the difference between the asymptotic performance of policies π_{k+1} greedy with respect to v_k and the optimal policy is

$$\limsup_{k\to\infty} \|v_* - v^{\pi_k}\|_\infty \leq \frac{2\gamma}{(1-\gamma)^2}\epsilon. \tag{7}$$

As for Value Iteraton, Munos has extended this result so that one can get bounds involving the L_p norm. He also showed how to relate the performance analysis to the Bellman residual $v_k - \mathcal{T}^{\pi_k} v_k$ that says how much v_k approximates the real value of the policy π_k; this is interesting when the evaluation step of Approximate Policy Iteration involves the minimization of this Bellman residual. It is important to note that this Bellman residual is different from the one we introduced in the previous section (we then considered $v_k - \mathcal{T} v_k = v_k - \mathcal{T}^{\pi_{k+1}} v_k$ where π_{k+1} is greedy with respect to v_k). To avoid confusion, and because it is related to some specific policy, we will call $v_k - \mathcal{T}^{\pi_k} v_k$ the **Policy Bellman residual**. Munos started by deriving a componentwise analysis. Write $P_k = P^{\pi_k}$ the stochastic matrix corresponding to the policy π_k which is greedy with respect to v_{k-1}, P_* the stochastic matrix corresponding to the (unknown) optimal policy π_*.

Lemma 2.6 (Componentwise Asymptotic Performance of Approximate Policy Iteration [4]). *The following matrices*

$$\begin{aligned} R_k &:= (1-\gamma)^2 (I-\gamma P_*)^{-1} P_{k+1} (I-\gamma P_{k+1})^{-1} \\ R'_k &:= (1-\gamma)^2 (I-\gamma P_*)^{-1} \left[P_* + \gamma P_{k+1} (I-\gamma P_{k+1})^{-1} P_k \right] \\ R''_k &:= (1-\gamma)^2 (I-\gamma P_*)^{-1} P_* (I-\gamma P_k) \end{aligned}$$

are stochastic and the asymptotic performance of the policies generated by Approximate Policy Iteration satisfies

$$\begin{aligned} \limsup_{k\to\infty} \; v_* - v^{\pi_k} &\leq \frac{2\gamma}{(1-\gamma)^2} \limsup_{k\to\infty} \; [R_k - R'_k]\, \epsilon_k \\ \limsup_{k\to\infty} \; v_* - v^{\pi_k} &\leq \frac{2\gamma}{(1-\gamma)^2} \limsup_{k\to\infty} \; [R_k - R''_k]\, (v_k - \mathcal{T}^{\pi_k} v_k). \end{aligned}$$

As for Value Iteration, the above componentwise bound leads to the following L_p bounds.

Proposition 2.7 (Asymptotic Performance of Approximate Policy Iteration I [4]). *Choose any p and any distribution μ. With the notations of Lemma 2.6, the asymptotic performance of the policies generated by Approximate Policy Iteration satisfies*

$$\begin{aligned} \limsup_{k\to\infty} \; \|v_* - v^{\pi_k}\|_{p,\mu} &\leq \frac{2\gamma}{(1-\gamma)^2} \limsup_{k\to\infty} \; \|\epsilon_k\|_{p,\frac{1}{2}\mu\left(R_{jk}+R'_k\right)} \\ \limsup_{k\to\infty} \; \|v_* - v^{\pi_k}\|_{p,\mu} &\leq \frac{2\gamma}{(1-\gamma)^2} \limsup_{k\to\infty} \; \|v_k - \mathcal{T}^{\pi_k} v_k\|_{p,\frac{1}{2}\mu\left(R_{jk}+R''_k\right)}. \end{aligned}$$

Using the concentration coefficient $C(\nu)$ introduced in the previous section (equation 6), it is also possible to show[4] the following weighted L_p bounds:

Proposition 2.8 (Asymptotic Performance of Approximate Policy Iteration II). *Let $C(\nu)$ be the concentration coefficient defined in equation 6. The asymptotic performance of the policies generated by Approximate Policy Iteration satisfies*

$$\begin{aligned} \limsup_{k\to\infty} \; \|v_* - v^{\pi_k}\|_{\infty} &\leq \frac{2\gamma \left(C(\nu)\right)^{1/p}}{(1-\gamma)^2} \limsup_{k\to\infty} \; \|\epsilon_k\|_{p,\nu} \\ \limsup_{k\to\infty} \; \|v_* - v^{\pi_k}\|_{\infty} &\leq \frac{2\gamma \left(C(\nu)\right)^{1/p}}{(1-\gamma)^2} \limsup_{k\to\infty} \; \|v_k - \mathcal{T}^{\pi_k} v_k\|_{p,\nu}. \end{aligned}$$

Again, the bounds of Propositions 2.7 and 2.8 with respect to the approximation error ϵ are better than that of Bertsekas and Tsitsiklis (equation 7). Compared to the analog result for Approximate Value Iteration (Propositions 2.3 and 2.4) where the bound depends on a *uniform* error bound ($\forall k, \|\epsilon_k\|_{p,\nu} \leq \epsilon$), the above bounds have the nice property that they only depend on *asymptotic* errors/residuals.

Finally, as for Approximate Value Iteration, a better bound (by a factor $\frac{1}{1-\gamma}$) might be obtained if the sequence of policies happens to converge. It can be shown (from [4], Remark 4 page 7) that:

[4]Similarly to footnote 3, this result is not explicitely stated by Munos in [4] but using techniques of another of his articles [5], it is straightforward to derive from Lemma 2.6. The current paper will anyway generalize this result (in Proposition 4.10 page 23).

Corollary 2.9 (Performance of Approximate Policy Iteration in case of convergence). *Let $C(\nu)$ be the concentration coefficient defined in equation 6. If the sequence of policies (π_k) converges to some π, then*

$$\begin{aligned} v_* - v^\pi &\leq \frac{2\gamma\left(C(\nu)\right)^{1/p}}{(1-\gamma)} \limsup_{k\to\infty} \|\epsilon_k\|_{p,\nu} \\ v_* - v^\pi &\leq \frac{2\gamma\left(C(\nu)\right)^{1/p}}{(1-\gamma)} \limsup_{k\to\infty} \|v_k - \mathcal{T}^{\pi_k} v_k\|_{p,\nu} . \end{aligned}$$

2.4 λ Policy Iteration

Value Iteration and Policy Iteration are often considered as unrelated algorithms. Though all the results we have emphasized so far are strongly related (and even sometimes identical), they were proved independently for each algorithm. In this section, we describe a family of algorithms called "λ Policy Iteration" (originally introduced in [1]) parameterized by a coefficient $\lambda \in (0,1)$, that generalizes them both. When $\lambda = 0$, λ Policy Iteration will reduce to Value Iteration while it will reduce to Policy Iteration when $\lambda = 1$. We will also briefly discuss the fact that λ Policy Iteration draws some connections with Temporal Difference algorithms which one finds in the Reinforcement Learning literature [9].

We begin by giving some intuition about how one can make a connection between Value Iteration and Policy Iteration. For the moment let us forget about the error term ϵ_k. Value Iteration seems to build a sequence of value functions and Policy Iteration a sequence of policies. Both algorithms can in fact be seen as updating a sequence of value-policy pairs. With some little rewriting — by decomposing the (nonlinear) Bellman operator $\mathcal{T}$ into 1) the maximization step and 2) the application of the (linear) Bellman operator — it can be seen that each iterate of Value Iteration is equivalent to the two following updates:

$$\left\{ \begin{array}{lcl} \pi_{k+1} & \leftarrow & \text{greedy}(v_k) \\ v_{k+1} & \leftarrow & \mathcal{T}^{\pi_{k+1}} v_k \end{array} \right. \Leftrightarrow \left\{ \begin{array}{lcl} \pi_{k+1} & \leftarrow & \text{greedy}(v_k) \\ v_{k+1} & \leftarrow & r^{\pi_{k+1}} + \gamma P^{\pi_{k+1}} v_k . \end{array} \right.$$

The left hande side of the above equation uses the operator $\mathcal{T}^{\pi_{k+1}}$ while the right hande side uses its definition. Similarly — by inversing in Algorithm 2 the order of 1) the estimation of the value of the current policy and 2) the update of the policy, and by using the fact that the value of policy π_{k+1} is the fixed point of $\mathcal{T}^{\pi_{k+1}}$ (equation 4) — it can be argued that every iteration of Policy Iteration does the following:

$$\left\{ \begin{array}{lcl} \pi_{k+1} & \leftarrow & \text{greedy}(v_k) \\ v_{k+1} & \leftarrow & (\mathcal{T}^{\pi_{k+1}})^\infty v_k \end{array} \right. \Leftrightarrow \left\{ \begin{array}{lcl} \pi_{k+1} & \leftarrow & \text{greedy}(v_k) \\ v_{k+1} & \leftarrow & (I - \gamma P^{\pi_{k+1}})^{-1} r^{\pi_{k+1}} \end{array} \right.$$

Thanks to this little rewriting, both algorithms now look close to each other. Both can be seen as having an estimate v_k of the value of policy π_k, from which they deduce a potentially better policy π_{k+1}. The corresponding value $v^{\pi_{k+1}}$ of this better policy may be regarded as a target which is going to be tracked by the next estimate v_{k+1}. The difference is in the update that enables to go from v_k to v_{k+1}: while Policy Iteration directly *jumps to* the value of π_{k+1} (by applying the Bellman operator $\mathcal{T}^{\pi_{k+1}}$ an infinite number of times), Value Iteration only *makes one step* towards it (by applying $\mathcal{T}^{\pi_{k+1}}$ only once). From this common view of Value Iteration, it would be possible to naturally introduce the well-known Modified Policy Iteration algorithm (see e.g. [7]) which *makes n steps* at each update:

$$\left\{ \begin{array}{lcl} \pi_{k+1} & \leftarrow & \text{greedy}(v_k) \\ v_{k+1} & \leftarrow & (\mathcal{T}^{\pi_{k+1}})^n v_k \end{array} \right. \Leftrightarrow \left\{ \begin{array}{lcl} \pi_{k+1} & \leftarrow & \text{greedy}(v_k) \\ v_{k+1} & \leftarrow & \left[I + \gamma P^{\pi_{k+1}} + ... + (\gamma P^{\pi_{k+1}})^{n-1}\right] r^{\pi_{k+1}} + (\gamma P^{\pi_{k+1}})^n v_k \end{array} \right.$$

The above common view is actually here interesting because it leads to a natural introduction of λ Policy Iteration. λ Policy Iteration is doing a λ-*adjustable step* towards the value of π_{k+1}:

$$\left\{ \begin{array}{lcl} \pi_{k+1} & \leftarrow & \text{greedy}(v_k) \\ v_{k+1} & \leftarrow & (1-\lambda)\sum_{j=0}^\infty \lambda^j (\mathcal{T}^{\pi_{k+1}})^{j+1} v_k \end{array} \right. \Leftrightarrow \left\{ \begin{array}{lcl} \pi_{k+1} & \leftarrow & \text{greedy}(v_k) \\ v_{k+1} & \leftarrow & (I - \lambda\gamma P^{\pi_{k+1}})^{-1}(r^{\pi_{k+1}} + (1-\lambda)\gamma P^{\pi_{k+1}} v_k) \end{array} \right.$$

More formally, λ Policy Iteration (see the above left hand side) consists in doing a λ-geometric average of the different number of applications of the Bellman operator $(\mathcal{T}^{\pi_{k+1}})^j$ to v_k. The right hand side is here interesting because it clearly shows that λ Policy Iteration generalizes Value Iteration (when $\lambda = 0$) and Policy Iteration (when $\lambda = 1$). The actual equivalence between the left and the right can be proved as follows:

$$\begin{aligned}
(1-\lambda)\sum_{j=0}^{\infty}\lambda^j(\mathcal{T}^{\pi_{k+1}})^{j+1}v_k &= (1-\lambda)\sum_{j=0}^{\infty}\lambda^j\left\{\left[\sum_{l=0}^{j}(\gamma P^{\pi_{k+1}})^l\right]r^{\pi_{k+1}} + (\gamma P^{\pi_{k+1}})^{j+1}v_k\right\} \\
&= \sum_{j=0}^{\infty}\sum_{l=0}^{j}(1-\lambda)\lambda^j(\gamma P^{\pi_{k+1}})^l r^{\pi_{k+1}} + (1-\lambda)\sum_{j=0}^{\infty}\lambda^j(\gamma P^{\pi_{k+1}})^{j+1}v_k \\
&= \sum_{l=0}^{\infty}\sum_{j=l}^{\infty}(1-\lambda)\lambda^j(\gamma P^{\pi_{k+1}})^l r^{\pi_{k+1}} + (1-\lambda)\sum_{j=0}^{\infty}\lambda^j(\gamma P^{\pi_{k+1}})^{j+1}v_k \\
&= \sum_{l=0}^{\infty}\left[\sum_{j=l}^{\infty}\left(\lambda^j(\gamma P^{\pi_{k+1}})^l r^{\pi_{k+1}} - \lambda^{j+1}(\gamma P^{\pi_{k+1}})^l r^{\pi_{k+1}}\right)\right] + (1-\lambda)\sum_{j=0}^{\infty}\lambda^j(\gamma P^{\pi_{k+1}})^{j+1}v_k \\
&= \sum_{l=0}^{\infty}\lambda^l(\gamma P^{\pi_{k+1}})^l r^{\pi_{k+1}} + (1-\lambda)\sum_{j=0}^{\infty}\lambda^j(\gamma P^{\pi_{k+1}})^{j+1}v_k \\
&= \sum_{l=0}^{\infty}(\lambda\gamma P^{\pi_{k+1}})^l\left(r^{\pi_{k+1}} + (1-\lambda)\gamma P^{\pi_{k+1}}v_k\right) \\
&= (I-\lambda\gamma P^{\pi_{k+1}})^{-1}\left(r^{\pi_{k+1}} + (1-\lambda)\gamma P^{\pi_{k+1}}v_k\right).
\end{aligned}$$

In order to describe the λ Policy Iteration algorithm, it is useful to introduce a new operator. For any value v and any policy π, define (the following four formulations are equivalent up to some little linear algebra manipulations):

$$\begin{aligned}
\mathcal{T}_\lambda^\pi v &:= v + (I-\lambda\gamma P^\pi)^{-1}(\mathcal{T}^\pi v - v) && (8)\\
&= (I-\lambda\gamma P^\pi)^{-1}(\mathcal{T}^\pi v - \lambda\gamma P^\pi v) \\
&= (I-\lambda\gamma P^\pi)^{-1}(r^\pi + (1-\lambda)\gamma P^\pi v) && (9)\\
&= (I-\lambda\gamma P^\pi)^{-1}(\lambda r^\pi + (1-\lambda)\mathcal{T}^\pi v) && (10)
\end{aligned}$$

λ Policy Iteration is formally described in Algorithm 3. Once again, our description includes a potential

Algorithm 3 λ `Policy Iteration`

Input: An MDP, $\lambda \in (0,1)$, an initial value v_0
Output: An (approximately) optimal policy
  $k \leftarrow 0$
  repeat
    $\pi_{k+1} \leftarrow \text{greedy}(v_k)$ // Update the policy
    $v_{k+1} \leftarrow \mathcal{T}_\lambda^{\pi_{k+1}} v_k + \epsilon_{k+1}$ // Update the estimate of the value of policy π_{k+1}
    $k \leftarrow k+1$
  until some convergence criterion
Return $\text{greedy}(v_k)$

error term each time the value is updated. Even with this error term, it is straightforward to see that the

algorithm reduces to Value Iteration (Algorithm 1) when $\lambda = 0$ and to Policy Iteration[5] (Algorithm 2) when $\lambda = 1$.

Relation with Reinforcement Learning The definition of the operator $\mathcal{T}_\lambda^\pi$ given by equation 9 is the form we have used for the introduction of λ Policy Iteration as an intermediate algorithm between Value Iteration and Policy Iteration. The equivalent form given by equation 8 can be used to make a connection with the TD(λ) algorithms[6] that one finds in the Reinforcement Learning literature, of which a reference is the book by Sutton & Barto [9]. Indeed, equation 9 can be seen as an incremental additive procedure:

$$v_{k+1} \leftarrow v_k + \delta_k$$

where

$$\delta_k := (I - \lambda\gamma P^{\pi_{k+1}})^{-1}(\mathcal{T}^{\pi_{k+1}} v_k - v_k)$$

is zero if and only if the value v_k is equal to the optimal value v_*. It can be shown (see [1] for a proof or simply look at the equivalence between equations 2 and 3 for the intuition) that the vector δ_k has components given by:

$$\Delta_k(i) = \lim_{N\to\infty} E_{\pi_{k+1}} \left[\sum_{j=0}^{N-1} (\lambda\gamma)^j d_k(i_j, i_{j+1}) \middle| i_0 = i \right]$$

with

$$d_k(i,j) := r(i, \pi_{k+1}(i), j) + \gamma V(j) - V(i)$$

being the temporal difference associated to transition $i \to j$, as introduced by Sutton & Barto [9]. When one uses a stochastic approximation of λ Policy Iteration, that is when the expectation $E_{\pi_{t+1}}$ is approximated by sampling, λ Policy Iteration reduces to the algorithm TD(λ) which is described in chapter 7 of [9]. Also, Bertsekas and Ioffe [1] showed that Approximate TD(λ) with a linear feature architecture, as described in chapter 8.2 of [9], corresponds to Approximate λ Policy Iteration where the value is updated by least square fitting using a gradient-type iteration after each sample. Eventually, the interested reader might notice that the "unified view" of Reinforcement Learning algorithms which is depicted in chapter 10.1 of [9] is in fact a picture of λ Policy Iteration.

Properties of Exact λ Policy Iteration In the original article introducing λ Policy Iteration [1], Bertsekas and Ioffe provide an analysis of the algorithm when it is run exactly (when $\epsilon_k = 0$). Define the following factor

$$\beta = \frac{(1-\lambda)\gamma}{1-\lambda\gamma}. \tag{11}$$

We have $0 \le \beta \le \gamma < 1$. If $\lambda = 0$ (Value Iteration) then $\beta = \gamma$, and if $\lambda = 1$ (Policy Iteration) then $\beta = 0$. They give some insight on what happens at each iteration. For all k, define the operator

$$\forall v, \quad M_k v := (1-\lambda)\mathcal{T}^{\pi_{k+1}} v_k + \lambda \mathcal{T}^{\pi_{k+1}} v \tag{12}$$

Assume $\mathcal{T}^{\pi_{k+1}}$ is a contraction mapping of modulus α for some norm $\|.\|$; this assumption is always true with $\alpha = \gamma$ and the max-norm. Then [1]

- M_k is a contraction mapping of modulus $\beta\alpha$ for the same norm $\|.\|$.

[5] Policy Iteration starts with an initial policy while λ Policy Iteration starts with some initial value. To be precise, 1 Policy Iteration starting with v_0 is equivalent to Policy Iteration starting with the greedy policy with respect to v_0.

[6] TD stands for Temporal Difference. The connection with TD algorithms was one of the motivation of the original article about λ Policy Iteration by Bertsekas and Ioffe [1]. Indeed, λ Policy Iteration is there also called "Temporal Difference Based Policy Iteration" and the presentation the authors give starts from the formulation of equation 8 (which is close to TD(λ)) and then makes the connection with Value Iteration and Policy Iteration.

- The next iterate v_{k+1} is the (unique) fixed point of M_k.

The operator M_k gives some insight on how to concretely implement one iteration of λ Policy Iteration: it can for instance be done through a Value Iteration-like algorithm which applies M_k iteratively. Also, the fact that its contraction factor β can be tuned is of particular importance because finding the corresponding fixed point can be much easier [1] than that of $\mathcal{T}^{\pi_{k+1}}$, which is only γ-contracting.

Bertsekas and Ioffe also show the convergence and provide an asymptotic rate of convergence:

Proposition 2.10 (Convergence and Rate of Convergence of Exact λ Policy Iteration [1]). *If the discount factor $\gamma < 1$, then v_k converges to v_*. Furthermore, after some index k_*, the rate of convergence is linear in β as defined in equation 11, that is*

$$\forall k \geq k_*, \quad \|v_{k+1} - v_*\| \leq \beta \|v_k - v_*\|.$$

By making λ close to 1, β can be arbitrarily close to 0 so the above rate of convergence might look impressive. This needs to be put into perspective: the index k_* is the index after which the policy π_k does not change anymore (and is equal to the optimal policy π_*). As we said when we introduced the algorithm, λ controls the speed at which one wants v_k to "track the target" $v^{\pi_{k+1}}$; when $\lambda = 1$, this is done in one step (and if $\pi_{k+1} = \pi_*$ then $v_{k+1} = v_*$). However, the bigger the value of λ, the less the operator M_k is contracting (recall that the contraction factor is β defined in equation 11), and the more time it might take to compute the next iterate v_{k+1} (its fixed point).

Analyses are usually simpler for Value Iteration than for Policy Iteration. Analyses of Value Iteration is based on the fact that it is an algorithm that computes the fixed point of the Bellman operator which is a γ-contraction mapping in max-norm (see e.g. [2]). Unfortunately, it can be shown that the operator by which Policy Iteration updates the value from one iteration to the next is in general not a contraction in max-norm; Analyses of Policy Iteration rely on some other properties, like the fact that the sequence values is (approximately) non-decreasing (see [2, 4]). In fact, this observation can be drawn for λ Policy Iteration as soon as it does not reduce to Value Iteration:

Proposition 2.11. *As soon as $\lambda > 0$, there exists no norm for which the operator by which λ Policy Iteration updates the value from one iteration to the next is a contraction.*

Proof. To see this, consider the following deterministic MDP with two states $\{1, 2\}$ and two actions $\{change, stay\}$: $r_1 = 0$, $r_2 = 1$, $P_{change}(s_2|s_1) = P_{change}(s_1|s_2) = P_{stay}(s_1|s_1) = P_{stay}(s_2|s_2) = 1$. Consider the following two value functions $v = (\epsilon, 0)$ and $v' = (0, \epsilon)$ with $\epsilon > 0$. Their corresponding greedy policies are $\pi = (stay, change)$ and $\pi' = (change, stay)$. Then, we can compute the next iterates of v and v' (using equation 9):

$$\begin{aligned}
r^\pi + (1-\lambda\gamma)P^\pi v &= ((1-\lambda)\gamma\epsilon, 1 + (1-\lambda)\gamma\epsilon) \\
\mathcal{T}_\lambda^\pi v &= \left(\frac{(1-\lambda)\gamma\epsilon}{1-\lambda\gamma}, 1 + \frac{(1-\lambda)\gamma\epsilon}{1-\lambda\gamma}\right) \\
r^{\pi'} + (1-\lambda\gamma)P^{\pi'} v' &= ((1-\lambda)\gamma\epsilon, 1 + (1-\lambda)\gamma\epsilon) \\
\mathcal{T}_\lambda^{\pi'} v' &= \left(\frac{1 + (1-\lambda)\gamma\epsilon}{1-\lambda\gamma} - 1, \frac{1 + (1-\lambda)\gamma\epsilon}{1-\lambda\gamma}\right)
\end{aligned}$$

Then

$$\mathcal{T}_\lambda^{\pi'} v' - \mathcal{T}_\lambda^\pi v = \left(\frac{1}{1-\lambda\gamma} - 1, \frac{1}{1-\lambda\gamma} - 1\right)$$

while

$$v' - v = (-\epsilon, \epsilon).$$

As all norms are equivalent, and as ϵ can be arbitrarily small, the norm of $\mathcal{T}_\lambda^\pi v - \mathcal{T}_\lambda^{\pi'} v'$ can be arbitrarily larger than norm of $v - v'$ when $\lambda > 0$. □

Approximate λ Policy Iteration In their original article [1], Bertsekas and Ioffe describe a case study involving an instance of Approximate λ Policy Iteration (with $(\epsilon_k \neq 0)$. However, to the best of our knowledge, there is no work studying the theoretical soundness of doing Approximate λ Policy Iteration.

2.5 Our contributions

Now that we have described the algorithms and some of their known properties, motivating the remaining of this paper is straightforward. λ Policy Iteration is conceptually a very nice algorithm since it generalizes the two most-well known algorithms for solving discounted infinite-horizon Markov Decision Processes. The natural question that arises is whether one can generalizes the results we have described so far to λ Policy Iteration(uniformly for all λ). The answer is yes:

- We provide a componentwise analysis of Exact and Approximate λ Policy Iteration.
- We show that the convergence rate of Exact λ Policy Iteration is linear (Proposition 2.10 only showed the *asymptotic* linear convergence) and we generalize the stopping criterion described for Value Iteration (Proposition 2.1).
- We give componentwise and span seminorm bounds of the asymptotic error of Approximate λ Policy Iteration with respect to the *asymptotic* approximation error, Bellman residual, and Policy Bellman residual. This generalizes Lemmas 2.2 and 2.6, and Propositions 2.3,2.4, 2.7 and 2.8. Our analysis actually implies that doing Approximate λ Policy Iteration is sound (when the approximation error tends to 0, the algorithm finds the optimal solution).
- We provide specific (better) bounds for the case when the value or the policy converges, which generalizes Corollaries 2.5 and 2.9.
- Last but not least, we provide all our results using the span seminorms we have introduced in section 1, and using the relations between this span semi-norms and the standard L_p norms (equation 1), it can be seen that our results are slightly stronger than all the previously described results.

Conceptually, we provide a unified vision (unified proofs, unified results) for all the mentioned algorithms.

3 Componentwise Performance bounds for λ Policy Iteration

This section contains our main results, which take the form of componentwise bounds of the performance bounds when using λ Policy Iteration. For clarity, most proofs are deferred to the appendix. The core of our work is the complete analysis of λ Policy Iteration (Appendix B). It serves as a basis for computing the rate of convergence of Exact λ Policy Iteration (Section 3.1, proof in Appendix C) and the asymptotic performance of Approximate λ Policy Iteration with respect to the approximation error (Section 3.2, proof in Appendix D). The asymptotic performance of Approximate λ Policy Iteration with respect to the Bellman residuals is somewhat simpler and is proved independently (Section 3.2, proof in Appendix A).

Notations For clarity, we use the following lighter notations: $P_k := P^{\pi_k}$, $\mathcal{T}_k := \mathcal{T}^{\pi_k}$, $P_* := P^{\pi_*}$. We will refer to the factor β as introduced by Bertsekas and Ioffe (equation 11 page 13). Also, the following stochastic matrix will play a recurrent role in our analysis:

$$A_k := (1 - \lambda\gamma)(I - \lambda\gamma P_k)^{-1} P_k.$$

3.1 Performance bounds for Exact λ Policy Iteration

Consider Exact λ Policy Iteration. We have the following convergence rate bounds:

Lemma 3.1 (Componentwise Rate of Convergence bounds for Exact λ Policy Iteration). *The following matrices*

$$\begin{aligned}
E_{kk_0} &:= (1-\gamma)(P_*)^{k-k_0}(I-\gamma P_*)^{-1} \\
E'_{kk_0} &:= \left(\frac{1-\gamma}{\gamma^{k-k_0}}\right)\left(\frac{\lambda\gamma}{1-\lambda\gamma}\sum_{j=k_0}^{k-1}\gamma^{k-1-j}\beta^{j-k_0}(P_*)^{k-1-j}A_{j+1}A_j...A_{k_0+1}+\beta^{k-k_0}(I-\gamma P_k)^{-1}A_kA_{k-1}...A_{k_0+1}\right) \\
F_{kk_0} &:= (1-\gamma)P_*^{k-k_0}+\gamma E'_{kk_0}P_*
\end{aligned}$$

are stochastic and

$$\begin{aligned}
v_* - v^{\pi_k} &\leq \frac{\gamma^{k-k_0}}{1-\gamma}\left[F_{kk_0}-E'_{kk_0}\right](v_*-v_{k_0}) \\
v_* - v^{\pi_k} &\leq \frac{\gamma^{k-k_0}}{1-\gamma}\left[E_{kk_0}-E'_{kk_0}\right](\mathcal{T}v_{k_0}-v_{k_0}). \\
v_* - v^{\pi_k} &\leq \gamma^{k-k_0}(P_*)^{k-k_0}\left[(v_*-v_{k_0})-\min_s[v_*(s)-v_{k_0}(s)]e\right]+\|v_*(s)-v^{\pi_{k_0+1}}\|_\infty e.
\end{aligned}$$

These bounds imply that λ Policy Iteration converges to the optimal value function at least at a linear rate.

3.2 Performance bounds for Approximate λ Policy Iteration

Let us now consider Approximate λ Policy Iteration. We provide componentwise bounds of the loss $v_* - v^{\pi_k}$ of using policy π_k instead of using the optimal policy, with respect to the approximation error ϵ_k, the Policy Bellman residual $\mathcal{T}_k v_k - v_k$ and the Bellman residual $\mathcal{T}v_k - v_k = \mathcal{T}_{k+1}v_k - v_k$. Recall the subtle difference between these two Bellman residuals: the Policy Bellman residual says how much v_k differs from the value of π_k while the Bellman residual says how much v_k differs from the value of the policies π_{k+1} and π_*.

Lemma 3.2 (Componentwise Performance bounds for Approximate λ Policy Iteration). *The following matrices*

$$\begin{aligned}
B_{jk} &:= \frac{1-\gamma}{\gamma^{k-j}}\left[\frac{\lambda\gamma}{1-\lambda\gamma}\sum_{i=j}^{k-1}\gamma^{k-1-i}\beta^{i-j}(P_*)^{k-1-i}A_{i+1}A_i...A_{j+1}+\beta^{k-j}(I-\gamma P_k)^{-1}A_kA_{k-1}...A_{j+1}\right]\\
B'_{jk} &:= \gamma B_{jk}P_j+(1-\gamma)(P_*)^{k-j}\\
C_k &:= (1-\gamma)^2(I-\gamma P_*)^{-1}\left(P_*(I-\gamma P_k)^{-1}\right)\\
C'_k &:= (1-\gamma)^2(I-\gamma P_*)^{-1}\left(P_{k+1}(I-\gamma P_{k+1})^{-1}\right)\\
D &:= (1-\gamma)P_*(I-\gamma P_*)^{-1}\\
D'_k &:= (1-\gamma)P_k(I-\gamma P_k)^{-1}
\end{aligned}$$

are stochastic and

$$\begin{aligned}
\forall k_0,\quad \limsup_{k\to\infty}\ v_*-v^{\pi_k} &\le \frac{1}{1-\gamma}\limsup_{k\to\infty}\sum_{j=k_0}^{k-1}\gamma^{k-j}\left[B_{jk}-B'_{jk}\right]\epsilon_j\\
\limsup_{k\to\infty}\ v_*-v^{\pi_k} &\le \frac{\gamma}{(1-\gamma)^2}\limsup_{k\to\infty}\ [C_k-C'_k](\mathcal{T}_kv_k-v_k)\\
\forall k,\quad v_*-v^{\pi_k} &\le \frac{\gamma}{1-\gamma}\left[D-D'_k\right](\mathcal{T}v_{k-1}-v_{k-1}).
\end{aligned}$$

We can look at the relation between our bound for general λ and the bounds derived by Munos (Lemmas 2.2 and 2.6):

- Let us consider the case where $\lambda=0$. Then $\beta=\gamma$, $A_k=P_k$ and

$$B_{jk}=(1-\gamma)(I-\gamma P_k)^{-1}P_kP_{k-1}...P_{j+1}.$$

  Then we have just shown that $\limsup_{k\to\infty}\ v_*-v^{\pi_k}$ is upper bounded by:

$$\limsup_{k\to\infty}\sum_{j=k_0}^{k-1}\gamma^{k-j}\left[(I-\gamma P_k)^{-1}P_kP_{k-1}...P_{j+1}-\left(\gamma(I-\gamma P_k)^{-1}P_kP_{k-1}...P_j+(P_*)^{k-j}\right)\right]\epsilon_j. \tag{13}$$

  The bound derived by Munos for Approximate Value Iteration (Lemma 2.2 page 7) is

$$\begin{aligned}
&\limsup_{k\to\infty}\ (I-\gamma P_k)^{-1}\sum_{j=0}^{k-1}\gamma^{k-j}\left[P_kP_{k-1}...P_{j+1}-(P_*)^{k-j}\right]\epsilon_j\\
=\ &\limsup_{k\to\infty}\sum_{j=0}^{k-1}\gamma^{k-j}\left[(I-\gamma P_k)^{-1}P_kP_{k-1}...P_{j+1}-(I-\gamma P_k)^{-1}(P_*)^{k-j}\right]\epsilon_j\\
=\ &\limsup_{k\to\infty}\sum_{j=0}^{k-1}\gamma^{k-j}\left[(I-\gamma P_k)^{-1}P_kP_{k-1}...P_{j+1}-\left((I-\gamma P_k)^{-1}\gamma P_k(P_*)^{k-j}+(P_*)^{k-j}\right)\right]\epsilon_j. \qquad (14)
\end{aligned}$$

  The above bounds are very close to each other: we go from equation 13 to equation 14 by replacing $P_{k-1}...P_j$ by $(P_*)^{k-j}$.

- When $\lambda=1$, $\beta=0$, $A_k=(1-\gamma)(I-\gamma P_k)^{-1}P_k$ and

$$B_{jk}=(1-\gamma)(P_*)^{k-1-j}P_{j+1}(I-\gamma P_{j+1})^{-1}.$$

Our bound is

$$\limsup_{k\to\infty} \; v_* - v^{\pi_k} \leq \limsup_{k\to\infty} \; \sum_{j=k_0}^{k-1} \gamma^{k-1-j} (P_*)^{k-1-j} u_j$$

with

$$u_j := \left[\gamma P_{j+1}(I-\gamma P_{j+1})^{-1}(I-\gamma P_j) - \gamma P_*\right] \epsilon_j.$$

By definition of the supremum limit, for all $\epsilon > 0$, there exists an index k_1 such that for all $j \geq k_1$,

$$u_j \leq \limsup_{l\to\infty} \; u_l + \epsilon e.$$

Then:

$$\begin{aligned}
\limsup_{k\to\infty} \; \sum_{j=k_1}^{k-1} \gamma^{k-1-j} (P_*)^{k-1-j} u_j &\leq \limsup_{k\to\infty} \; \sum_{j=k_1}^{k-1} \gamma^{k-1-j} (P_*)^{k-1-j} \left(\limsup_{l\to\infty} \; u_l + \epsilon \right) \\
&= (I-\gamma P_*)^{-1} \left(\limsup_{l\to\infty} \; u_l + \epsilon e \right).
\end{aligned}$$

As this is true for all $\epsilon > 0$, we eventually find the bound of Munos for Approximate Policy Iteration (Lemma 2.6 page 10).

Thus, up to some little details, our componentwise analysis unifies those of Munos. It is actually not surprising that we be able to find the result of Munos for Approximate Policy Iteration because the proof we develop in appendix B is a generalization of Munos's proof for Approximate Policy Iteration in [4]. Similarly, the reason why we don't exactly fall back on the componentwise of Munos for Approximate Value Iteration is that our proof uses a technique that is slightly different from that of Munos in [5]. It would in fact be possible to write a proof that generalizes that of Approximate Value Iteration but this is not really fundamental as most of the results we are going to deduce won't be affected.

The bound with respect to the approximation error can be improved if we know or observe that the value or the policy converges. Note that the former condition implies the latter.

Corollary 3.3. *Suppose the value converges to some v. Write π its greedy policy and P the corresponding stochastic matrix. Consider the following stochastic matrices:*

$$\begin{aligned}
B_v &:= (1-\gamma)\left[(1-\lambda)(I-\gamma P)^{-1}P + \lambda(I-\gamma P_*)^{-1}P\right] \\
D &:= (1-\gamma)P_*(I-\gamma P_*)^{-1}
\end{aligned}$$

Then the error necessarily converges to some ϵ and

$$v_* - v^\pi \leq \frac{\gamma}{1-\gamma}\left[B_v - D\right]\epsilon$$

Corollary 3.4. *Suppose the policy converges to some π. Write P the corresponding stochastic matrix. The following matrices*

$$\begin{aligned}
A^\pi &:= (1-\lambda\gamma)P(I-\lambda\gamma P)^{-1} \\
{A^\pi}_{jk} &:= \frac{1-\gamma}{\gamma^{k-j}}\left[\frac{\lambda\gamma}{1-\lambda\gamma}\sum_{i=j}^{k-1}\gamma^{k-1-i}\beta^{i-j}(P_*)^{k-1-i}(A^\pi)^{i-j}+\beta^{k-j}(I-\gamma P)^{-1}(A^\pi)^{k-1-j}\right] \\
B^\pi_{jk} &:= {A^\pi}_{jk}P \\
B'^\pi_{jk} &:= \frac{1-\gamma}{1-\lambda\gamma}\left[(P_*)^{k-j}+\frac{(1-\lambda\gamma)\gamma(1-\lambda)}{1-\gamma}{A^\pi}_{jk}(I-\lambda\gamma P)^{-1}(P)^2\right].
\end{aligned}$$

are stochastic and

$$v_* - v^\pi \leq \limsup_{k\to\infty} \frac{1-\lambda\gamma}{1-\gamma}\sum_{j=k_0}^{k-1}\gamma^{k-j}\left[B^\pi_{jk}-B'^\pi_{jk}\right]\epsilon_j$$

4 Span seminorm Performance bounds for λ Policy Iteration

Following the lines of Munos [5, 4], we here show that the componentwise bounds we have derived in the previous section enable us to derive span seminorms bounds. We first describe some lemmas that enable us to make this derivation. Then we present the results for λ Policy Iteration.

4.1 From Componentwise bounds to span seminorm bounds

We here present four lemmas that show how to derive span seminorms bounds from componentwise bounds. The proofs are close to those of [5, 4] and deferred to appendix E for clarity. The main difference is that we come up with span seminorm bounds instead of L_p/max-norms.

The first two lemmas are:

Lemma 4.1. *Let x_k, y_k be sequences of vectors and X_k, X'_k sequences of stochastic matrices satisfying*

$$\limsup_{k\to\infty} |x_k| \leq K \limsup_{k\to\infty} (X_k - X'_k)y_k$$

where $|x|$ denotes the componentwise absolute value of x. For all distribution μ, $\mu_k := \frac{1}{2}\mu(X_k + X'_k)$ is a distribution and

$$\limsup_{k\to\infty} \|x_k\|_{p,\mu} \leq K \limsup_{k\to\infty} span_{p,\mu_k}[y_k]$$

$$\limsup_{k\to\infty} \|x_k\|_\infty \leq K \limsup_{k\to\infty} span_\infty[y_k]$$

Lemma 4.2. *Let x_k, y_k be sequences of vectors and X_{jk}, X'_{jk} sequences of stochastic matrices satisfying*

$$\forall k_0, \quad \limsup_{k\to\infty} |x_k| \leq K \limsup_{k\to\infty} \sum_{j=k_0}^{k-1} \gamma^{k-j}(X_{kj} - X'_{kj})y_j$$

For all distribution μ,

$$\mu_{kj} := \frac{1}{2}\mu(X_{kj} + X'_{kj})$$

are distributions and

$$\limsup_{k\to\infty} \|x_k\|_{p,\mu} \leq \frac{K\gamma}{1-\gamma} \lim_{k_0\to\infty} \left[\sup_{k\geq j\geq k_0} span_{p,\mu_{kj}}[y_j]\right].$$

$$\limsup_{k\to\infty} \|x_k\|_\infty \leq \frac{K\gamma}{1-\gamma} \limsup_{k\to\infty} span_\infty[y_k]$$

The two other lemmas consider the **concentration coefficient** introduced by Munos in [4, 5] and already mentioned before (equation 6 page 8). We recall its definition. We assume there exists a distribution ν and a real number $C(\nu)$ such that

$$C(\nu) := \max_{i,j,a} \frac{p_{ij}(a)}{\nu(j)}.$$

If X is an average of products of stochastic matrices of the MDP, it can be seen that for any non-negative vector y $\mu X y \leq C(\nu)\nu y$. Thus, with the notation of Lemma 4.1, we have for all k and all vector y,

$$\begin{aligned}
\left(\text{span}_{p,\mu_k}[y]\right)^p &= \min_a \left(\|y - ae\|_{p,\mu_k}\right)^p \\
&= \min_a \mu_k |y - ae|^p \\
&= \min_a \left(\frac{1}{2}\mu(X_k + X'_k)\right)|y - ae|^p \\
&\leq C(\nu) \min_a \nu|y - ae|^p \\
&= C(\nu) \min_a \left(\|y - ae\|_{p,\nu}\right)^p \\
&= C(\nu)\left(\text{span}_{p,\nu}[y]\right)^p.
\end{aligned}$$

This leads to an analogue of Lemma 4.1 involving this concentration coefficient:

Lemma 4.3. *Let x_k, y_k be sequences of vectors and X_k, X'_k sequences of stochastic matrices (that are averages of products of stochastic matrices of the MDP) satisfying*

$$\limsup_{k\to\infty} |x_k| \leq K \limsup_{k\to\infty} (X_k - X'_k)y_k.$$

Then

$$\limsup_{k\to\infty} \|x_k\|_\infty \leq K\left[C(\nu)\right]^{1/p} \limsup_{k\to\infty} span_{p,\nu}[y_k].$$

Similarly, we get the following analogue of Lemma 4.2:

Lemma 4.4. *Let x_k, y_k be sequences of vectors and X_{jk}, X'_{jk} sequences of stochastic matrices (that are averages of products of stochastic matrices of the MDP) satisfying*

$$\forall k_0, \quad \limsup_{k\to\infty} |x_k| \leq K \limsup_{k\to\infty} \sum_{j=k_0}^{k-1} \gamma^{k-j}(X_{kj} - X'_{kj})y_j.$$

Then

$$\limsup_{k\to\infty} \|x_k\|_\infty \leq \frac{K\gamma}{1-\gamma} [C(\nu)]^{1/p} \limsup_{k\to\infty} span_{p,\nu}[y_k].$$

We are now ready to give some span seminorm bounds for λ Policy Iteration.

4.2 Rates of convergence for Exact λ Policy Iteration

Consider Exact λ Policy Iteration. Using Lemma 4.1, Lemma 3.1 becomes:

Proposition 4.5 (Rates of convergence for Exact λ Policy Iteration in span seminorm). *With the notations of Lemma 3.1, and for all distribution μ and indices $k > k_0$,*

$$\begin{aligned}
\|v_* - v^{\pi_k}\|_{p,\mu} &\leq \frac{\gamma^{k-k_0}}{1-\gamma} span_{p,\frac{1}{2}\mu\left(F_{kk_0}+E'_{kk_0}\right)}[v_* - v_{k_0}] \\
\|v_* - v^{\pi_k}\|_\infty &\leq \frac{\gamma^{k-k_0}}{1-\gamma} span_\infty[v_* - v_{k_0}] \\
\|v_* - v^{\pi_k}\|_{p,\mu} &\leq \frac{\gamma^{k-k_0}}{1-\gamma} span_{p,\frac{1}{2}\mu\left(E_{kk_0}+E'_{kk_0}\right)}[\mathcal{T}v_{k_0} - v_{k_0}] \\
\|v_* - v^{\pi_k}\|_\infty &\leq \frac{\gamma^{k-k_0}}{1-\gamma} span_\infty[\mathcal{T}v_{k_0} - v_{k_0}] \\
\|v_* - v^{\pi_k}\|_{p,\mu} &\leq \gamma^{k-k_0}\left(\left\|(v_* - v_{k_0}) - \min_s[v_*(s) - v_{k_0}(s)]e\right\|_{p,\mu(P_*)^{k-k_0}} + \|v_*(s) - v^{\pi_{k_0+1}}\|_\infty\right) \\
\|v_* - v^{\pi_k}\|_\infty &\leq \gamma^{k-k_0}\left(span_\infty[v_* - v_{k_0}] + \|v_*(s) - v^{\pi_{k_0+1}}\|_\infty\right) \qquad (15)
\end{aligned}$$

The first pair of rate is expressed in terms of the distance between the value function and the optimal value function at some iteration k_0. The second pair of inequalities can be used as a stopping criterion. Indeed, taking $k = k_0 + 1$ it implies for the following Stopping condition, which generalizes that of Proposition 2.1:

Proposition 4.6 (Stopping condition for Exact λ Policy Iteration). *If at some iteration k_0, the value v_{k_0} satisfies:*

$$span_\infty[\mathcal{T}v_{k_0} - v_{k_0}] \leq \frac{1-\gamma}{\gamma}\epsilon$$

then the greedy policy π_{k_0+1} with respect to v_{k_0} is ϵ-optimal: $\|v_* - v^{\pi_{k_0+1}}\|_\infty < \epsilon$.

The last pair of inequalities rely on the distance between the value function and the optimal value function and the value difference between the optimal policy and the first greedy policy; compared to the others, it has the advantage of not containing explicitely a $\frac{1}{1-\gamma}$ factor (though this factor is hidden in $v_* - v^{\pi_{k_0+1}}$). To the best of our knowledge, this bound is even new for the specific cases of Value Iteration and Policy Iteration.

Using Lemma 4.3, Lemma 3.1 gives the following analysis with respect to the concentration coefficient.

Proposition 4.7 (Rates of convergence for Exact λ Policy Iteration in span seminorm). *Let $C(\nu)$ be the concentration coefficient defined in equation 6. For all p and $k \geq k_0$,*

$$\begin{aligned}
\|v_* - v^{\pi_k}\|_\infty &\leq \frac{\gamma^{k-k_0}}{1-\gamma} [C(\nu)]^{1/p} \, span_{p,\nu} [v_* - v_{k_0}] \\
\|v_* - v^{\pi_k}\|_\infty &\leq \frac{\gamma^{k-k_0}}{1-\gamma} [C(\nu)]^{1/p} \, span_{p,\nu} [\mathcal{T} v_{k_0} - v_{k_0}]
\end{aligned}$$

Also:

$$\|v_* - v^{\pi_k}\|_\infty \leq \gamma^{k-k_0} [C(\nu)]^{1/p} \left(span_{p,\mu} [v_* - v_{k_0}] + \|v_*(s) - v^{\pi_{k_0+1}}\|_\infty \right)$$

4.3 Bounds for Approximate λ Policy Iteration

We can derive the same kind of bounds for Approximate λ Policy Iteration. Using Lemmas 4.1 and 4.2, Lemma 3.2 gives the following Proposition.

Proposition 4.8 (Performance bounds for Approximate λ Policy Iteration in span seminorm). *With the notations of Lemma 3.2, and for all p and all distribution μ*

$$\begin{aligned}
\limsup_{k\to\infty} \|v_* - v^{\pi_k}\|_{p,\mu} &\leq \frac{\gamma}{(1-\gamma)^2} \lim_{k_0\to\infty} \left[\sup_{k\geq j\geq k_0} span_{p,\frac{1}{2}\mu\left(B_{jk}+B'_{jk}\right)} [\epsilon_j] \right] \\
\limsup_{k\to\infty} \|v_* - v^{\pi_k}\|_\infty &\leq \frac{\gamma}{(1-\gamma)^2} \limsup_{j\to\infty} \; span_\infty [\epsilon_j] \\
\limsup_{k\to\infty} \|v_* - v^{\pi_k}\|_{p,\mu} &\leq \frac{\gamma}{(1-\gamma)^2} \limsup_{k\to\infty} \; span_{p,\frac{1}{2}\mu\left(C_k+C'_k\right)} [\mathcal{T}_k v_k - v_k] \\
\limsup_{k\to\infty} \|v_* - v^{\pi_k}\|_\infty &\leq \frac{\gamma}{(1-\gamma)^2} \limsup_{k\to\infty} \; span_\infty [\mathcal{T}_k v_k - v_k] \\
\forall k, \quad \|v_* - v^{\pi_k}\|_{p,\mu} &\leq \frac{\gamma}{1-\gamma} span_{p,\frac{1}{2}\mu\left(D+D'_k\right)} [\mathcal{T} v_{k-1} - v_{k-1}] \\
\forall k, \quad \|v_* - v^{\pi_k}\|_\infty &\leq \frac{\gamma}{1-\gamma} span_\infty [\mathcal{T} v_{k-1} - v_{k-1}]
\end{aligned}$$

Similarly, Corollaries 2.5 and 2.9 give:

Corollary 4.9 (Performance bounds for App.λ Policy Iteration in case of convergence). *If the value converges to some v, then the approximation error converges to some ϵ. With the notations of Corollaries 3.3 and 3.4, the corresponding greedy policy π satisfies for all p*

$$\begin{aligned}
\|v_* - v^\pi\|_{p,\mu} &\leq \frac{\gamma}{1-\gamma} span_{p,\frac{1}{2}\mu(B_v+D)} [\epsilon] \\
\|v_* - v^\pi\|_\infty &\leq \frac{\gamma}{1-\gamma} span_\infty [\epsilon]
\end{aligned}$$

If the policy converges to some π, then:

$$\begin{aligned}
\|v_* - v^\pi\|_{p,\mu} &\leq \frac{\gamma(1-\lambda\gamma)}{(1-\gamma)^2} \lim_{k_0\to\infty} \left[\sup_{k\geq j\geq k_0} span_{p,\frac{1}{2}\mu\left(B^\pi_{jk}+B'^\pi_{jk}\right)} [\epsilon_j] \right] \\
\|v_* - v^\pi\|_\infty &\leq \frac{\gamma(1-\lambda\gamma)}{(1-\gamma)^2} \limsup_{j\to\infty} \; span_\infty [\epsilon_j]
\end{aligned}$$

Eventually, using Lemmas 4.3 and 4.4, we have the following bounds involving the concentration coefficient.

Proposition 4.10 (Performance bounds for Approximate λ Policy Iteration in span seminorm).

Let $C(\nu)$ be the concentration coefficient defined in equation 6. For all p and all k,

$$\begin{aligned}
\limsup_{k\to\infty} \ \|v_* - v^{\pi_k}\|_\infty &\leq \frac{\gamma}{(1-\gamma)^2}\left[C(\nu)\right]^{1/p} \limsup_{j\to\infty} \ span_{p,\nu}\left[\epsilon_j\right] \\
\limsup_{k\to\infty} \ \|v_* - v^{\pi_k}\|_\infty &\leq \frac{\gamma}{(1-\gamma)^2}\left[C(\nu)\right]^{1/p} \limsup_{k\to\infty} \ span_{p,\nu}\left[\mathcal{T}_k v_k - v_k\right] \\
\forall k, \quad \|v_* - v^{\pi_k}\|_\infty &\leq \frac{\gamma}{1-\gamma}\left[C(\nu)\right]^{1/p} span_{p,\nu}\left[\mathcal{T} v_{k-1} - v_{k-1}\right].
\end{aligned}$$

Corollary 4.11 (Performance bounds for App. λ Policy Iteration in case of convergence).
If the value converges to some v, then the approximation error converges to some ϵ, and the corresponding greedy policy π satisfies:

$$\|v_* - v^{\pi}\|_\infty \leq \frac{\gamma}{1-\gamma}\left[C(\nu)\right]^{1/p} span_{p,\nu}\left[\epsilon\right]$$

If the policy converges to some π, then:

$$\|v_* - v^{\pi}\|_\infty \leq \frac{\gamma(1-\lambda\gamma)}{(1-\gamma)^2}\left[C(\nu)\right]^{1/p} \limsup_{j\to\infty} \ span_{p,\nu}\left[\epsilon_j\right]$$

When comparing the specific bounds of Munos for Approximate Value Iteration (Propositions 2.3 and 2.4) and Approximate Policy Iteration (Propositions 2.7 and 2.8), we wrote that the latter had the nice property that the bounds only depend on *asymptotic* errors/residuals (while the former depends on all errors). Our bounds for λ Policy Iteration have this nice property too. Considering the relations between the span seminorms and the other standard norms (equation 1 page 4), we see that our results are more general than those of Munos.

Conclusion and Future work

We have considered the λ Policy Iteration algorithm which generalizes the standard algorithms Value Iteration and Policy Iteration. We have reviewed some results by Puterman [7], Bertsekas and Tsitsiklis [2] and Munos [4, 5], concerning the rate of convergence of these standard algorithms in the exact and approximate cases. We have extended these results to λ Policy Iteration and proposed some new convergence rate in the exact case (equation 15 page 21). The performance analysis we have described slightly improves the previous results in two ways. 1) As suggested by the results of Puterman in the exact case, the use of the span seminorm has enabled us to derive tighter bounds in general[7]. 2) Our analysis of Approximate λ Policy Iteration relates the asymptotic performance of the algorithm to the asymptotic errors/residuals instead of a uniform bound of the errors/residuals and this might be of practical interest. More importantly, the main contribution of this paper has been to provide a unified view of the approximate versions of optimal control and reinforcement learning algorithms and to show that it is sound to use them.

We now describe some future research directions. Munos introduced in [5] some concentration coefficients that are finer than the one we used throughout the paper and a natural track would be to consider them in order to derive finer performance bounds. From what we have achieved here, this not trivial since the componentwise analysis we derived for λ Policy Iteration is significanlty more intricate than the ones we find in the specific limit cases $\lambda = 0$ (Value Iteration) and $\lambda = 1$ (Policy Iteration). Analogously to a recent work

[7] If the approximation of λ Policy Iteration is *only* due to the use of a Least-square Feature Based Linear Architecture, and if the constant e belongs to the set of features, then the span seminorm bound is equivalent to the standard L_p norm bound.

by Munos and Szepesvari [6] for Approximate Value Iteration, a potential use of these coefficients would be to derive an analysis of Approximate λ Policy Iteration involving PAC-style polynomial bounds on the number of samples and a quantitative measure of the power of the approximation architecture (like the well-known VC dimension).

Another important direction is to study the implications of the choice of the parameter λ, as for instance is done by Singh and Dayan in [8]. The original analysis by Bertsekas and Ioffe [1] shows how one can concretely implement Exact λ Policy Iteration. Each iteration requires the computation of the fixed point of a β-contracting operator (see equation 12 page 13). We plan to study the tradeoff between the ease for computing this fixed point (the smaller β the faster) and the time for λ Policy Iteration to converge to the optimal policy (the bigger β the faster). In parallel, the reader might have noticed that most of the bounds we have provided do not depend on λ. We believe that the finer concentration coefficients of Munos we have just discussed should also help keeping track of the influence of λ on the performance of the exact or approximate algorithm. We expect that we should be able to derive results relating the smoothness of the MDP problem to the choice of λ.

Appendix

The following Appendices contains all the proofs concerning the analysis of λ Policy Iteration. We write $P_k = P^{\pi_k}$ the stochastic matrix corresponding to the policy π_k which is greedy with respect to v_{k-1}, P_* the stochastic matrix corresponding to the optimal policy π_*. Similarly we write $\mathcal{T}_k$ and $\mathcal{T}_*$ the associated Bellman operators.

The proof techniques we have developped are strongly inspired by those of Munos in the articles [4, 5]. Most of the inequalities will appear from the definition of the greedy operator:

$$\pi = \text{greedy} \Leftrightarrow \forall \pi', \mathcal{T}^{\pi'} v \leq \mathcal{T}^{\pi} v.$$

We will often use the property that an average of stochastic matrices is also a stochastic matrix. A recurrent instance of this property is: if P is some stochastic matrix, then the geometric average

$$(1-\alpha)\sum_{i=0}^{\infty}(\alpha P)^i = (1-\alpha)(I-\alpha P)^{-1}$$

with $0 \leq \alpha < 1$ is also a stochastic matrix. Finally, we will use the property that if some vectors x and y are such that $x \leq y$, then $Px \leq Py$ for any stochastic matrix.

A Componentwise bounds with respect to the Bellman residuals

In this Appendix, we study the loss

$$l_k := v_* - v^{\pi_k}$$

with respect to the two following **Bellman residuals**:

$$b'_k := \mathcal{T}_k v_k - v_k$$

$$b_k := \mathcal{T}_{k+1} v_k - v_k = \mathcal{T} v_k - v_k$$

b'_k says how much v_k differs from the value of π_k while b_k says how much v_k differs from the value of the policies π_{k+1} and π_*.

A.1 Bounds with respect to Policy Bellman residual

Our analysis relies on the following lemma

Lemma A.1. *Suppose we have a policy π, a function v that is an approximation of the value v^π of π in the sense that its residual $b' := T^\pi v - v$ is small. Taking the greedy policy π' with respect to v will reduce the loss as follows:*

$$v_* - v^{\pi'} \leq \gamma P_*(v_* - v^\pi) + \left(\gamma P_*(I-\gamma P)^{-1} - \gamma P'(I-\gamma P')^{-1}\right) b'$$

where P and P' are the stochastic matrices which correspond to π and π'.

Proof. We have:

$$\begin{aligned} v_* - v^{\pi'} &= \mathcal{T}_* v_* - \mathcal{T}^{\pi'} v^{\pi'} \\ &= \mathcal{T}_* v_* - \mathcal{T}_* v^\pi + \mathcal{T}_* v^\pi - \mathcal{T}_* v + \mathcal{T}_* v - \mathcal{T}^{\pi'} v + \mathcal{T}^{\pi'} v - \mathcal{T}^{\pi'} v^{\pi'} \\ &\leq \gamma P_*(v_* - v^\pi) + \gamma P_*(v^\pi - v) + \gamma P'(v - v^{\pi'}) \end{aligned} \tag{16}$$

where we used the fact that $\mathcal{T}_* v \leq \mathcal{T}^{\pi'} v$. One can see that:

$$\begin{aligned} v^\pi - v &= \mathcal{T}^\pi v^\pi - v \\ &= \mathcal{T}^\pi v^\pi - \mathcal{T}^\pi v + \mathcal{T}^\pi v - v \\ &= \gamma P(v^\pi - v) + b' \\ &= (I - \gamma P)^{-1} b' \end{aligned} \tag{17}$$

and that

$$\begin{aligned} v - v^{\pi'} &= v - \mathcal{T}^{\pi'} v^{\pi'} \\ &= v - \mathcal{T}^\pi v + \mathcal{T}^\pi v - \mathcal{T}^{\pi'} v + \mathcal{T}^{\pi'} v - \mathcal{T}^{\pi'} v^{\pi'} \\ &\leq -b' + \gamma P'(v - v^{\pi'}) \\ &\leq (I - \gamma P')^{-1}(-b'). \end{aligned} \tag{18}$$

where we used the fact that $\mathcal{T}^\pi v \leq \mathcal{T}^{\pi'} v$. We get the result by putting back equations 17 and 18 into equation 16. □

To derive a bound for λ Policy Iteration, we simply apply the above lemma to $\pi = \pi_k$, $v = v_k$ and $\pi' = \pi_{k+1}$. We thus get:

$$l_{k+1} \leq \gamma P_* l_k + \left(\gamma P_*(I - \gamma P_k)^{-1} - \gamma P_{k+1}(I - \gamma P_{k+1})^{-1}\right) b'_k$$

Introduce the following stochastic matrices:

$$C_k := (1-\gamma)^2 (I - \gamma P_*)^{-1} \left(P_*(I - \gamma P_k)^{-1}\right)$$

$$C'_k := (1-\gamma)^2 (I - \gamma P_*)^{-1} \left(P_{k+1}(I - \gamma P_{k+1})^{-1}\right)$$

This leads to the following componentwise bound:

$$\limsup_{k\to\infty} \; l_k \leq \frac{\gamma}{(1-\gamma)^2} \limsup_{k\to\infty} \; [C_k - C'_k]\, b'_k$$

A.2 Bounds with respect to Bellman residual

We rely on the following lemma (which is for instance proved by Munos in [5])

Lemma A.2. *Suppose we have a function v. Let π be the greedy policy with respect to v. Then*

$$v_* - v^\pi \leq \gamma \left[P_*(I - \gamma P_*)^{-1} - P^\pi (I - \gamma P^\pi)^{-1}\right] (\mathcal{T}^\pi v - v)$$

We provide a proof for the sake of completeness:

Proof. Using the fact that $\mathcal{T}_* v \leq \mathcal{T}^\pi v$, we see that

$$\begin{aligned} v_* - v^\pi &= \mathcal{T}_* v_* - \mathcal{T}^\pi v^\pi \\ &= \mathcal{T}_* v_* - \mathcal{T}_* v + \mathcal{T}_* v - \mathcal{T}^\pi v + \mathcal{T}^\pi v - \mathcal{T}^\pi v^\pi \\ &\leq \mathcal{T}_* v_* - \mathcal{T}_* v + \mathcal{T}^\pi v - \mathcal{T}^\pi v^\pi \\ &= \gamma P_*(v_* - v) + \gamma P^\pi (v - v^\pi) \\ &= \gamma P_*(v_* - v^\pi) + \gamma P_*(v^\pi - v) \gamma P^\pi (v - v^\pi) \\ &\leq (I - \gamma P_*)^{-1} (\gamma P_* - \gamma P^\pi)(v^\pi - v). \end{aligned}$$

Using equation 17 we see that:

$$v^\pi - v = (I - \gamma P^\pi)^{-1}(\mathcal{T}^\pi v - v).$$

Thus

$$\begin{aligned} v_* - v^\pi &\leq (I - \gamma P_*)^{-1}(\gamma P_* - \gamma P^\pi)(I - \gamma P^\pi)^{-1}(\mathcal{T}^\pi v - v) \\ &= (I - \gamma P_*)^{-1}(\gamma P_* - I + I - \gamma P^\pi)(I - \gamma P^\pi)^{-1}(\mathcal{T}^\pi v - v) \\ &= \left[(I - \gamma P_*)^{-1} - (I - \gamma P^\pi)^{-1}\right](\mathcal{T}^\pi v - v) \\ &= \gamma\left[P_*(I - \gamma P_*)^{-1} - P^\pi(I - \gamma P^\pi)^{-1}\right](\mathcal{T}^\pi v - v). \quad \square \end{aligned}$$

To derive a bound for λ Policy Iteration, we simply apply the above lemma to $v = v_{k-1}$ and $\pi = \pi_k$. We thus get:

$$l_k \leq \frac{\gamma}{1-\gamma}\left[D - D'_k\right] b_{k-1} \tag{19}$$

where

$$\begin{aligned} D &:= (1-\gamma)P_*(I - \gamma P_*)^{-1} \\ D'_k &:= (1-\gamma)P_k(I - \gamma P_k)^{-1} \end{aligned}$$

are stochastic matrices.

B General componentwise analysis of λ Policy Iteration

This Appendix contains the core of all the remaining results. We show how to compute an upper bound of the loss for (approximate) λ Policy Iteration in general. It will be the basis for the derivation of componentwise bounds for approximate λ Policy Iteration (section 3.2) and Exact λ Policy Iteration (section 3.1).

B.1 Overview of the analysis of the componentwise loss bound

We define:

- the **loss** of using policy π_k instead of the optimal policy:

$$l_k := v_* - v^{\pi_k}$$

- the **value** of the kth iterate b.a. (before approximation):

$$w_k := v_k - \epsilon_k$$

- the **distance** between the optimal value and the kth value b.a. :

$$d_k := v_* - w_k$$

- the **shift** between the kth value b.a. and the value of the kth policy:

$$s_k := w_k - v^{\pi_k}$$

- the **Bellman residual** between the $k-1^{th}$ and the kth values b.a.:

$$b_k := \mathcal{T}_{k+1} v_k - v_k = \mathcal{T} v_k - v_k$$

All our results come from a series of relations involving the above quantities. In this Appendix, we will use the notation $\overline{x}$ for an upper bound of x and $\underline{x}$ for a lower bound.

Let us define the following stochastic matrix:

$$A_k := (1-\lambda\gamma)P_k(I-\lambda\gamma P_k)^{-1}.$$

Also Define the following factor

$$\beta = \frac{(1-\lambda)\gamma}{1-\lambda\gamma}.$$

Then

Lemma B.1. *The shift is related to the Bellman residual:*

$$s_k = \beta(I-\gamma P_k)^{-1}A_k(-b_{k-1}).$$

Lemma B.2. *The Bellman residual at iteration $k+1$ cannot be much lower than the Bellman residual at iteration k:*

$$b_{k+1} \geq \beta A_{k+1}b_k + x_{k+1}$$

where $x_k := (\gamma P_k - I)\epsilon_k$ only depends on the approximation error.

As a consequence, a lower bound of the Bellman residual is:

$$b_k \geq \sum_{j=k_0+1}^{k} \beta^{k-j}\left(A_kA_{k-1}...A_{j+1}\right)x_j + \beta^{k-k_0}\left(A_kA_{k-1}...A_{k_0+1}\right)b_{k_0} := \underline{b_k}.$$

Using Lemma B.1, the bound on the Bellman residual also provides an upper on the shift:

$$s_k \leq \beta(I-\gamma P_k)^{-1}A_k(-\underline{b_{k-1}}) := \overline{s_k}$$

Lemma B.3. *The distance tends to reduce:*

$$d_{k+1} \leq \gamma P_* d_k + y_k$$

where $y_k := \frac{\lambda\gamma}{1-\lambda\gamma}A_{k+1}(-\underline{b_k}) - \gamma P_*\epsilon_k$ depends on the lower bound of the Bellman residual and the approximation error.

Then, an upper bound of the distance is:

$$d_k \leq \sum_{j=k_0}^{k-1}\gamma^{k-1-j}(P_*)^{k-1-j}y_j + \gamma^{k-k_0}(P_*)^{k-k_0}d_{k_0} = \overline{d_k}.$$

Eventually, as

$$l_k = d_k + s_k \leq \overline{d_k} + \overline{s_k},$$

the upper bounds on the distance and the shift will enable us to derive the upper bound on the loss.

The above proof is a generalization of that of Munos in [4] for Approximate Policy Iteration. When $\lambda = 1$, that is when both proofs coincide, since $\beta = 0$ Lemmas B.1 and B.2 have a particularly trivial form.

B.2 Proof of Lemma B.1: a relation between the shift and the Bellman residual

We have:

$$\begin{aligned}
(I-\gamma P_k)s_k &= (I-\gamma P_k)(w_k - v^{\pi_k}) \\
&= (I-\gamma P_k)w_k - r_k \\
&= (I-\lambda\gamma P_k + \lambda\gamma P_k - \gamma P_k)w_k - r_k \\
&= (I-\lambda\gamma P_k)w_k + (\lambda\gamma P_k - \gamma P_k)w_k - r_k \\
&= r_k + (1-\lambda)\gamma P_k v_{k-1} + (\lambda-1)\gamma P_k w_k - r_k \\
&= (1-\lambda)\gamma P_k(v_{k-1} - w_k) \\
&= (1-\lambda)\gamma P_k(I-\lambda\gamma P_k)^{-1}(v_{k-1} - \mathcal{T}_k v_{k-1}) \\
&= (1-\lambda)\gamma P_k(I-\lambda\gamma P_k)^{-1}(-b_{k-1})
\end{aligned}$$

Therefore

$$s_k = \beta(I-\gamma P_k)^{-1}A_k(-b_{k-1})$$

with

$$A_k := (1-\lambda\gamma)P_k(I-\lambda\gamma P_k)^{-1}.$$

Suppose we have a lower bound of the Bellman residual: $b_k \geq \underline{b_k}$ (we will derive one soon). Since $(I-\gamma P_k)^{-1}A_k$ only has non-negative elements then

$$s_k \leq \beta(I-\gamma P_k)^{-1}A_k(-\underline{b_k}) := \overline{s_k}.$$

B.3 Proof of Lemma B.2: a lower bound of the Bellman residual

From the definition of the algorithm, and using the fact that $\mathcal{T}_k v^{\pi_k} = v^{\pi_k}$ we see that:

$$\begin{aligned}
b_k &= \mathcal{T}_{k+1}v_k - v_k \\
&= \mathcal{T}_{k+1}v_k - \mathcal{T}_k v_k + \mathcal{T}_k v_k - v_k \\
&\geq \mathcal{T}_k v_k - v_k \\
&= \mathcal{T}_k v_k - \mathcal{T}_k v^{\pi_k} + v^{\pi_k} - v_k \\
&= \gamma P_k(v_k - v^{\pi_k}) + v^{\pi_k} - v_k \\
&= (\gamma P_k - I)(s_k + \epsilon_k). \\
&= \beta A_k b_{k-1} + (\gamma P_k - I)\epsilon_k. \qquad (20)
\end{aligned}$$

where we eventually used the relation between s_k and b_k (Lemma B.1). In other words:

$$b_{k+1} \geq \beta A_{k+1} b_k + x_{k+1}$$

with

$$x_k := (\gamma P_k - I)\epsilon_k.$$

Since A_k is a stochastic matrix and $\beta \geq 0$, we get by induction:

$$b_k \geq \sum_{j=k_0+1}^{k} \beta^{k-j}\left(A_k A_{k-1} ... A_{j+1}\right) x_j + \beta^{k-k_0}\left(A_k A_{k-1} ... A_{k_0+1}\right) b_{k_0} := \underline{b_k}.$$

B.4 Proof of Lemma B.3: an upper bound of the distance

Given that $\mathcal{T}_* v_* = v_*$, we have

$$\begin{aligned} v_* &= v_* + (I - \lambda\gamma P_{k+1})^{-1}(\mathcal{T}_* v_* - v_*) \\ &= (I - \lambda\gamma P_{k+1})^{-1}(\mathcal{T}_* v_* - \lambda\gamma P_{k+1} v_*) \end{aligned}$$

Therefore the distance satisfies:

$$\begin{aligned} d_{k+1} &= v_* - w_{k+1} \\ &= (I - \lambda\gamma P_{k+1})^{-1}[(\mathcal{T}_* v_* - \lambda\gamma P_{k+1} v_*) - (\mathcal{T}_{k+1} v_k - \lambda\gamma P_{k+1} v_k)] \\ &= (I - \lambda\gamma P_{k+1})^{-1}[\mathcal{T}_* v_* - \mathcal{T}_{k+1} v_k + \lambda\gamma P_{k+1}(v_k - v_*)] \\ &= \lambda\gamma P_{k+1} d_{k+1} + \mathcal{T}_* v_* - \mathcal{T}_{k+1} v_k + \lambda\gamma P_{k+1}(v_k - v_*) \\ &= \lambda\gamma P_{k+1} d_{k+1} + \mathcal{T}_* v_* - \mathcal{T}_{k+1} v_k + \lambda\gamma P_{k+1}(w_k + \epsilon_k - v_*) \\ &= \lambda\gamma P_{k+1} d_{k+1} + \mathcal{T}_* v_* - \mathcal{T}_{k+1} v_k + \lambda\gamma P_{k+1}(\epsilon_k - d_k) \\ &= \mathcal{T}_* v_* - \mathcal{T}_{k+1} v_k + \lambda\gamma P_{k+1}(\epsilon_k + d_{k+1} - d_k) \end{aligned}$$

Since π^{k+1} is greedy with respect to v_k, we have $\mathcal{T}_{k+1} v_k \geq \mathcal{T}_* v_k$ and therefore:

$$\begin{aligned} \mathcal{T}_* v_* - \mathcal{T}_{k+1} v_k &= \mathcal{T}_* v_* - \mathcal{T}_* v_k + \mathcal{T}_* v_k - \mathcal{T}_{k+1} v_k \\ &\leq \mathcal{T}_* v_* - \mathcal{T}_* v_k \\ &= \gamma P_*(v_* - v_k) \\ &= \gamma P_*(v_* - (w_k + \epsilon_k)) \\ &= \gamma P_* d_k - \gamma P_* \epsilon_k \end{aligned}$$

As a consequence, the distance satisfies:

$$d_{k+1} \leq \gamma P_* d_k + \lambda\gamma P_{k+1}(\epsilon_k + d_{k+1} - d_k) - \gamma P_* \epsilon_k$$

Noticing that:

$$\begin{aligned} \epsilon_k + d_{k+1} - d_k &= \epsilon_k + w_k - w_{k+1} \\ &= v_k - w_{k+1} \\ &= -(I - \lambda\gamma P_{k+1})^{-1}(\mathcal{T}_{k+1} v_k - v_k) \\ &= (I - \lambda\gamma P_{k+1})^{-1}(-b_k) \\ &\leq (I - \lambda\gamma P_{k+1})^{-1}(-\underline{b_k}) \end{aligned}$$

we get:

$$d_{k+1} \leq \gamma P_* d_k + y_k$$

where

$$y_k := \frac{\lambda\gamma}{1 - \lambda\gamma} A_{k+1}(-\underline{b_k}) - \gamma P_* \epsilon_k.$$

Since P_* is a stochastic matrix and $\gamma \geq 0$, we have by induction:

$$d_k \leq \sum_{j=k_0}^{k-1} \gamma^{k-1-j} (P_*)^{k-1-j} y_j + \gamma^{k-k_0} (P_*)^{k-k_0} d_{k_0} = \overline{d_k}.$$

C Componentwise rate of convergence for Exact λ Policy Iteration

We here derive the convergence rate bounds for Exact λ Policy Iteration (as expressed in Proposition 4.5 page 21). We rely on the loss bound analysis of Appendix B with $\epsilon_k = 0$. In this specific case, we know that the loss $l_k \leq \overline{d_k} + \overline{s_k}$ where

$$\begin{aligned}
-\underline{b_k} &= \beta^{k-k_0} A_k A_{k-1} ... A_{k_0+1}(-b_{k_0}) \\
\overline{d_k} &= \frac{\lambda\gamma}{1-\lambda\gamma} \sum_{j=k_0}^{k-1} \gamma^{k-1-j} (P_*)^{k-1-j} A_{j+1}(-\underline{b_j}) + \gamma^{k-k_0}(P_*)^{k-k_0} d_{k_0} \\
\overline{s_k} &= \beta(I-\gamma P_k)^{-1} A_k(-\underline{b_{k-1}})
\end{aligned}$$

We therefore have:

$$\overline{d_k} = \frac{\lambda\gamma}{1-\lambda\gamma} \sum_{j=k_0}^{k-1} \gamma^{k-1-j} \beta^{j-k_0} (P_*)^{k-1-j} A_{j+1} A_j ... A_{k_0+1}(-b_{k_0}) + \gamma^{k-k_0}(P_*)^{k-k_0} d_{k_0}$$

and

$$\overline{s_k} = \beta^{k-k_0}(I-\gamma P_k)^{-1} A_k A_{k-1} ... A_{k_0+1}(-b_{k_0})$$

Therefore:

$$l_k \leq \left(\frac{\gamma^{k-k_0}}{1-\gamma}\right) E'_{kk_0}(-b_{k_0}) + \gamma^{k-k_0}(P_*)^{k-k_0} d_{k_0} \tag{21}$$

with

$$E'_{kk_0} := \left(\frac{1-\gamma}{\gamma^{k-k_0}}\right)\left(\frac{\lambda\gamma}{1-\lambda\gamma}\sum_{j=k_0}^{k-1} \gamma^{k-1-j}\beta^{j-k_0} X_{j,k_0,k} + \frac{\beta^{k-k_0}}{1-\gamma} Y_{k_0,k}\right).$$

Lemma C.1. *E'_{kk_0} is a stochastic matrix*

Proof.

$$\begin{aligned}
\|E'_{kk_0}\| &= \frac{1-\gamma}{\gamma^{k-k_0}}\left(\frac{\lambda\gamma}{1-\lambda\gamma}\sum_{j=k_0}^{k-1}\gamma^{k-1-j}\beta^{j-k_0} + \frac{\beta^{k-k_0}}{1-\gamma}\right) \\
&= \frac{1-\gamma}{\gamma^{k-k_0}}\left(\frac{\lambda\gamma}{1-\lambda\gamma}\frac{\gamma^{k-k_0}-\beta^{k-k_0}}{\gamma-\beta} + \frac{\beta^{k-k_0}}{1-\gamma}\right) \\
&= \frac{1-\gamma}{\gamma^{k-k_0}}\left(\frac{\gamma^{k-k_0}-\beta^{k-k_0}}{1-\gamma} + \frac{\beta^{k-k_0}}{1-\gamma}\right) \\
&= 1
\end{aligned}$$

where we used the facts that $\frac{\lambda\gamma}{\gamma-\beta} = \frac{1}{1-\beta}$ and $(1-\beta)(1-\lambda\gamma) = 1-\gamma$. □

C.1 A bound with respect to the Bellman residual

We first need the following lemma:

Lemma C.2. *The bias and the distance are related as follows:*

$$b_k \geq (I-\gamma P_*)d_k.$$

Proof. Since π_{k+1} is greedy with respect to v_k, $\mathcal{T}_{k+1}v_k \geq \mathcal{T}_* v_k$ and

$$\begin{aligned} b_k &= \mathcal{T}_{k+1}v_k - v_k \\ &= \mathcal{T}_{k+1}v_k - \mathcal{T}_* v_k + \mathcal{T}_* v_k - \mathcal{T}_* v_* + v_* - v_k \\ &\geq \gamma P_*(v_k - v_*) + v_* - v_k \\ &= (I - \gamma P_*)d_k. \quad \square \end{aligned}$$

We thus have:

$$d_{k_0} \leq (I - \gamma P_*)^{-1} b_{k_0}$$

Then equation 21 becomes

$$\begin{aligned} l_k &\leq \left[\gamma^{k-k_0}(P_*)^{k-k_0}(I - \gamma P_*)^{-1} - \left(\frac{\gamma^{k-k_0}}{1-\gamma}\right) E'_{kk_0}\right] b_{k_0} \\ &= \frac{\gamma^{k-k_0}}{1-\gamma}\left[E_{kk_0} - E'_{kk_0}\right] b_{k_0} \end{aligned}$$

where:

$$E_{kk_0} := (1-\gamma)(P_*)^{k-k_0}(I - \gamma P_*)^{-1}$$

is a stochastic matrix.

C.2 A bound with respect to the distance

From Lemma C.2, we know that

$$-b_{k_0} \leq (I - \gamma P_*)(-d_{k_0})$$

Then equation 21 becomes

$$\begin{aligned} l_k &\leq \left[\gamma^{k-k_0}(P_*)^{k-k_0} - \left(\frac{\gamma^{k-k_0}}{1-\gamma}\right) E'_{kk_0}(I - \gamma P_*)\right] d_{k_0} \\ &= \frac{\gamma^{k-k_0}}{1-\gamma}\left[F_{kk_0} - E'_{kk_0}\right] d_{k_0} \end{aligned}$$

where

$$F_{kk_0} := (1-\gamma)P_*^{k-k_0} + \gamma E'_{kk_0} P_*$$

is a stochastic matrix.

C.3 A bound with respect to the distance and the loss of the greedy policy

Let K be a constant such that $\hat{v}_{k_0} := v_{k_0} - Ke$. The following statements are equivalent:

$$\begin{aligned} \hat{b}_{k_0} &\geq 0 \\ \mathcal{T}_{k_0+1}\hat{v}_{k_0} &\geq \hat{v}_{k_0} \\ r_{k_0+1} + \gamma P_{k_0+1}(v_{k_0} - Ke) &\geq v_{k_0} - Ke \\ (I - \gamma P_{k_0+1})Ke &\geq -r_{k_0+1} + (I - \gamma P_{k_0+1})v_{k_0} \\ Ke &\geq (I - \gamma P_{k_0+1})^{-1}(-r_{k_0+1}) + v_{k_0} \\ Ke &\geq v_{k_0} - v^{\pi_{k_0+1}} \end{aligned}$$

The minimal K for which $\hat{b}_{k_0} \geq 0$ is thus $K := \max_s[v_{k_0}(s) - v^{\pi_{k_0+1}}(s)]$. As $\hat{v}_{k_0}$ and v_{k_0} only differ by a constant vector, they will generate the same sequence of policies $\pi_{k_0+1}, \pi_{k_0+2}...$ Then, as $\hat{b}_{k_0} \geq 0$, equation 21 tells us that

$$\begin{aligned} v_* - v^{\pi_k} &\leq \gamma^{k-k_0}(P_*)^{k-k_0}(v_* - \hat{v}_{k_0}) \\ &= \gamma^{k-k_0}(P_*)^{k-k_0}(v_* - v_{k_0} + Ke) \end{aligned}$$

Now notice that

$$\begin{aligned} K &= \max_s[v_{k_0}(s) - v_*(s) + v_*(s) - v^{\pi_{k_0+1}}(s)] \\ &\leq \max_s[v_{k_0}(s) - v_*(s)] + \max_s[v_*(s) - v^{\pi_{k_0+1}}(s)] \\ &= -\min_s[v_*(s) - v_{k_0}(s)] + \|v_*(s) - v^{\pi_{k_0+1}}\|_\infty \end{aligned}$$

Then, using the fact that $(P_*)^{k-k_0}e = e$, we get:

$$v_* - v^{\pi_k} \leq \gamma^{k-k_0}(P_*)^{k-k_0}\left[(v_* - v_{k_0}) - \min_s[v_*(s) - v_{k_0}(s)]e\right] + \|v_*(s) - v^{\pi_{k_0+1}}\|_\infty e.$$

D Asymptotic componentwise loss bounds with respect to the approximation error

We here use the loss bound analysis of Appendix B to derive an asymptotic analysis of approximate λ Policy Iteration with respect to the approximation error.

D.1 General analysis

Since

$$l_k = d_k + s_k \leq \overline{d_k} + \overline{s_k}, \tag{22}$$

an upper bound of the loss can be derived from the upper bound of the distance and the shift.

Let us first concentrate on the bound $\overline{d_k}$ of the distance. So far we have proved that:

$$\begin{aligned} \overline{d_k} &= \sum_{i=k_0}^{k-1} \gamma^{k-1-i}(P_*)^{k-1-i}y_i + \mathcal{O}(\gamma^{k-k_0}) \\ y_i &= \frac{\lambda\gamma}{1-\lambda\gamma}A_{i+1}(-\underline{b_i}) - \gamma P_*\epsilon_i \\ -\underline{b_i} &= \sum_{j=k_0}^{i} \beta^{i-j}\left(A_iA_{i-1}...A_{j+1}\right)(-x_j) + \mathcal{O}(\gamma^{i-k_0}) \\ -x_j &= (I - \gamma P_j)\epsilon_j. \end{aligned}$$

Writing

$$X_{i,j,k} := (P_*)^{k-1-i}A_{i+1}A_i...A_{j+1}$$

and putting all things together, we see that:

$$\begin{aligned}
\overline{d_k} &= \frac{\lambda\gamma}{1-\lambda\gamma}\sum_{i=k_0}^{k-1}\gamma^{k-1-i}\left(\sum_{j=k_0}^{i}\beta^{i-j}X_{i,j,k}(I-\gamma P_j)\epsilon_j+\mathcal{O}(\gamma^{i-k_0})\right)-\sum_{i=k_0}^{k-1}\gamma^{k-i}(P_*)^{k-i}\epsilon_i+\mathcal{O}(\gamma^{k-k_0})\\
&= \frac{\lambda\gamma}{1-\lambda\gamma}\sum_{i=k_0}^{k-1}\sum_{j=k_0}^{i}\gamma^{k-1-i}\beta^{i-j}X_{i,j,k}(I-\gamma P_j)\epsilon_j-\sum_{i=k_0}^{k-1}\gamma^{k-i}(P_*)^{k-i}\epsilon_i+\mathcal{O}(\gamma^{k-k_0})\\
&= \frac{\lambda\gamma}{1-\lambda\gamma}\sum_{j=k_0}^{k-1}\sum_{i=j}^{k-1}\gamma^{k-1-i}\beta^{i-j}X_{i,j,k}(I-\gamma P_j)\epsilon_j-\sum_{j=k_0}^{k-1}\gamma^{k-j}(P_*)^{k-j}\epsilon_j+\mathcal{O}(\gamma^{k-k_0})\\
&= \sum_{j=k_0}^{k-1}\left[\left(\frac{\lambda\gamma}{1-\lambda\gamma}\sum_{i=j}^{k-1}\gamma^{k-1-i}\beta^{i-j}X_{i,j,k}(I-\gamma P_j)\right)-\gamma^{k-j}(P_*)^{k-j}\right]\epsilon_j+\mathcal{O}(\gamma^{k-k_0}) \qquad (23)
\end{aligned}$$

Let us now consider the bound $\overline{s_k}$ of the shift:

$$\begin{aligned}
\overline{s_k} &= \beta(I-\gamma P_k)^{-1}A_k(-\underline{b_k})\\
&= \beta(I-\gamma P_k)^{-1}A_k\left[\left(\sum_{j=k_0}^{k-1}\beta^{k-1-j}\left(A_{k-1}A_{k-2}...A_{j+1}\right)(-x_j)\right)+\mathcal{O}(\gamma^{k-k_0})\right]\\
&= \sum_{j=k_0}^{k-1}\frac{\beta^{k-j}}{1-\gamma}Y_{j,k}(I-\gamma P_j)\epsilon_j+\mathcal{O}(\gamma^{k-k_0}) \qquad (24)
\end{aligned}$$

with

$$Y_{j,k} := (1-\gamma)(I-\gamma P_k)^{-1}A_kA_{k-1}...A_{j+1}.$$

Eventually, from equations 22, 23 and 24 we get:

$$l_k \le \sum_{j=k_0}^{k-1}\left[\left(\frac{\lambda\gamma}{1-\lambda\gamma}\sum_{i=j}^{k-1}\gamma^{k-1-i}\beta^{i-j}X_{i,j,k}+\frac{\beta^{k-j}}{1-\gamma}Y_{j,k}\right)(I-\gamma P_j)-\gamma^{k-j}(P_*)^{k-j}\right]\epsilon_j+\mathcal{O}(\gamma^{k-k_0}) \qquad (25)$$

Introduce the following matrices:

$$\begin{aligned}
B_{jk} &:= \frac{1-\gamma}{\gamma^{k-j}}\left[\frac{\lambda\gamma}{1-\lambda\gamma}\sum_{i=j}^{k-1}\gamma^{k-1-i}\beta^{i-j}X_{i,j,k}+\frac{\beta^{k-j}}{1-\gamma}Y_{j,k}\right]\\
B'_{jk} &:= \gamma B_{jk}P_j+(1-\gamma)(P_*)^{k-j}
\end{aligned}$$

Lemma D.1. *B_{jk} and B'_{jk} are stochastic matrices.*

Proof. It is clear from the definition of $X_{i,j,k}$ and $Y_{j,k}$ that normalizing B_{jk} and B'_{jk} will give stochastic matrices. So we just need to check that their norm is 1.

$$\begin{aligned}
\|B_{jk}\| &= \frac{(1-\gamma)}{\gamma^{k-j}}\left[\frac{\lambda\gamma}{1-\lambda\gamma}\sum_{i=j}^{k-1}\gamma^{k-1-i}\beta^{i-j}+\frac{\beta^{k-j}}{1-\gamma}\right] \\
&= \frac{(1-\gamma)}{\gamma^{k-j}}\left[\frac{\lambda\gamma}{1-\lambda\gamma}\frac{\gamma^{k-j}-\beta^{k-j}}{\gamma-\beta}+\frac{\beta^{k-j}}{1-\gamma}\right] \\
&= \frac{(1-\gamma)}{\gamma^{k-j}}\left[\frac{\gamma^{k-j}-\beta^{k-j}}{(1-\lambda\gamma)(1-\beta)}+\frac{\beta^{k-j}}{1-\gamma}\right] \\
&= \frac{(1-\gamma)}{\gamma^{k-j}}\left[\frac{\gamma^{k-j}-\beta^{k-j}}{1-\gamma}+\frac{\beta^{k-j}}{1-\gamma}\right] \\
&= 1.
\end{aligned}$$

where we used the identities: $\lambda\gamma = \frac{\gamma-\beta}{1-\beta}$ and $(1-\beta)(1-\gamma\lambda) = 1-\gamma$. Then it is also clear that $\|B'_{jk}\| = 1$. □

Equation 25 can be rewritten as follows:

$$\begin{aligned}
l_k &\leq \sum_{j=k_0}^{k-1}\left[\frac{\gamma^{k-j}}{1-\gamma}B_{jk}(I-\gamma P_j)-\gamma^{k-j}(P_*)^{k-j}\right]\epsilon_j+\mathcal{O}(\gamma^{k-k_0}) \\
&= \frac{1}{1-\gamma}\sum_{j=k_0}^{k-1}\gamma^{k-j}\left[B_{jk}-B'_{jk}\right]\epsilon_j+\mathcal{O}(\gamma^{k-k_0})
\end{aligned}$$

Taking the supremum limit, we see that for all k_0,

$$\limsup_{k\to\infty} l_k \leq \frac{1}{1-\gamma}\limsup_{k\to\infty}\sum_{j=k_0}^{k-1}\gamma^{k-j}\left[B_{jk}-B'_{jk}\right]\epsilon_j \tag{26}$$

D.2 When the value converges

Suppose λ Policy Iteration converges to some value v. Let policy π be the corresponding greedy policy, with stochastic matrix P. Let b be the Bellman residual of v. It is also clear that the approximation error also converges to some ϵ. Indeed from Algorithm 3 and equation 8, we get:

$$b = \mathcal{T}v - v = (I-\lambda\gamma P)(-\epsilon)$$

From the bound with respect to the Bellman residual (equation 19 page 27), we can see that:

$$\begin{aligned}
v_* - v^\pi &\leq \left[(I-\gamma P_*)^{-1}-(I-\gamma P)^{-1}\right]b \\
&= \left[(I-\gamma P)^{-1}-(I-\gamma P_*)^{-1}\right](I-\lambda\gamma P)\epsilon \\
&= \left[(I-\gamma P)^{-1}(I-\lambda\gamma P)-(I-\gamma P_*)^{-1}(I-\lambda\gamma P)\right]\epsilon \\
&= \left[(I-\gamma P)^{-1}(I-\gamma P+\gamma P-\lambda\gamma P)-(I-\gamma P_*)^{-1}(I-\lambda\gamma P)\right]\epsilon \\
&= \left[\left(I+(1-\lambda)(I-\gamma P)^{-1}\gamma P+\lambda(I-\gamma P_*)^{-1}\gamma P\right)-(I-\gamma P_*)^{-1}\right]\epsilon \\
&= \left[\left((1-\lambda)(I-\gamma P)^{-1}\gamma P+\lambda(I-\gamma P_*)^{-1}\gamma P\right)-(I-\gamma P_*)^{-1}\gamma P_*\right]\epsilon \\
&= \frac{\gamma}{1-\gamma}\left[B_v-D\right]\epsilon.
\end{aligned}$$

where

$$\begin{aligned} B_v &:= (1-\gamma)\left((1-\lambda)(I-\gamma P)^{-1}P + \lambda(I-\gamma P_*)^{-1}P\right) \\ D &:= (1-\gamma)P_*(I-\gamma P_*)^{-1}. \end{aligned}$$

Lemma D.2. *B_v and D are stochastic matrices.*

Proof. It is clear that $\|D\| = 1$. Also:

$$\begin{aligned} \|B_v\| &= (1-\gamma)\left(1 + \frac{(1-\lambda)\gamma}{1-\gamma} + \frac{\lambda\gamma}{1-\gamma}\right) \\ &= (1-\gamma)\left(1 + \frac{\gamma}{1-\gamma}\right) \\ &= 1. \quad \square \end{aligned}$$

D.3 When the policy converges

Suppose λ Policy Iteration converges to some policy π. Write P the corresponding stochastic matrix and

$$A^\pi := (1-\lambda\gamma)P(I-\lambda\gamma P)^{-1}.$$

Then for some big enough k_0, we have:

$$l_k \leq \sum_{j=k_0}^{k-1} \left[\frac{\gamma^{k-j}}{1-\gamma} {A^\pi}_{jk} A^\pi (I-\gamma P) - \gamma^{k-j}(P_*)^{k-j}\right] \epsilon_j + \mathcal{O}(\gamma^{k-k_0})$$

where

$${A^\pi}_{jk} := \frac{1-\gamma}{\gamma^{k-j}} \left[\frac{\lambda\gamma}{1-\lambda\gamma} \sum_{i=j}^{k-1} \gamma^{k-1-i}\beta^{i-j}(P_*)^{k-1-i}(A^\pi)^{i-j} + \beta^{k-j}(I-\gamma P)^{-1}(A^\pi)^{k-1-j}\right]$$

is a stochastic matrix (for the same reasons why B_{jk} is a stochastic matrix in Lemma D.1). Noticing that

$$\begin{aligned} A^\pi(I-\gamma P) &= (1-\lambda\gamma)P(I-\lambda\gamma P)^{-1}(I-\gamma P) \\ &= (1-\lambda\gamma)P(I-\lambda\gamma P)^{-1}(I-\lambda\gamma P + \lambda\gamma P - \gamma P) \\ &= (1-\lambda\gamma)P(I-(1-\lambda)(I-\lambda\gamma P)^{-1}\gamma P) \\ &= (1-\lambda\gamma)P - \gamma(1-\lambda)A^\pi P \end{aligned}$$

we can deduce that

$$\begin{aligned} l_k &\leq \sum_{j=k_0}^{k-1} \left[\frac{\gamma^{k-j}}{1-\gamma} {A^\pi}_{jk} \left[(1-\lambda\gamma)P - \gamma(1-\lambda)A^\pi P\right] - \gamma^{k-j}(P_*)^{k-j}\right] \epsilon_j + \mathcal{O}(\gamma^{k-k_0}) \\ &= \sum_{j=k_0}^{k-1} \gamma^{k-j} \left[\frac{1-\lambda\gamma}{1-\gamma} {A^\pi}_{jk} P - \left[\frac{\gamma(1-\lambda)}{1-\gamma} {A^\pi}_{jk} A^\pi P + (P_*)^{k-j}\right]\right] \epsilon_j + \mathcal{O}(\gamma^{k-k_0}) \\ &= \frac{1-\lambda\gamma}{1-\gamma} \sum_{j=k_0}^{k-1} \gamma^{k-j} \left[B^\pi_{jk} - B'^\pi_{jk}\right] \epsilon_j + \mathcal{O}(\gamma^{k-k_0}) \end{aligned} \tag{27}$$

where

$$\begin{aligned} B^{\pi}_{jk} &:= {A^{\pi}}_{jk} P \\ B'^{\pi}_{jk} &:= \frac{1-\gamma}{1-\lambda\gamma}\left[\frac{\gamma(1-\lambda)}{1-\gamma}{A^{\pi}}_{jk} A^{\pi} P + (P_*)^{k-j}\right]. \end{aligned}$$

Lemma D.3. *B^{π}_{jk} and B'^{π}_{jk} are stochastic matrices.*

Proof. It is clear that $\|B^{\pi}_{jk}\| = 1$. Also:

$$\begin{aligned} \|B'^{\pi}_{jk}\| &= \frac{1-\gamma}{1-\lambda\gamma}\left(1 + \frac{\gamma(1-\lambda)}{1-\gamma}\right) \\ &= \frac{1-\gamma}{1-\lambda\gamma}\frac{1-\gamma+\gamma-\lambda\gamma}{1-\gamma} \\ &= 1. \quad \square \end{aligned}$$

E From componentwise bounds to span seminorm bounds

This Appendix contains the proofs of the Lemmas 4.1,4.2, 4.3 and 4.4 that enable us to derive span seminorm performance bounds from the componentwise analysis developped in the previous Appendices.

E.1 Proof of lemma 4.1

Write $a_k := \arg\min_a \|y_k - ae\|_{p,\mu_k}$. As X_k and X'_k are stochastic matrices, $X_k e = X'_k e = e$, and we can write that:

$$\limsup_{k\to\infty} |x_k| \leq K \limsup_{k\to\infty} (X_k - X'_k)(y_k - a_k e).$$

By taking the absolute value componentwise we get

$$\limsup_{k\to\infty} |x_k| \leq K \limsup_{k\to\infty} (X_k + X'_k)|y_k - a_k e|.$$

It can then be seen that

$$\begin{aligned} \limsup_{k\to\infty} \left(\|x_k\|_{p,\mu}\right)^p &= K^p \limsup_{k\to\infty} \mu\left(|x_k|\right)^p \\ &\leq K^p \limsup_{k\to\infty} \mu\left[\left(\frac{1}{2}(X_k + X'_k)\right) 2|y_k - a_k e|\right]^p \\ &\leq K^p \limsup_{k\to\infty} \frac{1}{2}\mu(X_k + X'_k)(2|y_k - a_k e|)^p \\ &= K^p \limsup_{k\to\infty} \mu_k(2|y_k - a_k e|)^p \\ &= K^p \limsup_{k\to\infty} \left(2\|y_k - a_k e\|_{p,\mu_k}\right)^p \\ &= K^p \limsup_{k\to\infty} \left(\mathrm{span}_{p,\mu_k}[y_k]\right)^p \end{aligned}$$

where we used Jensen's inequality (using the convexity of $x \mapsto x^p$) and where the last inequality results from the definition of a_k.

E.2 Proof of lemma 4.2

Write $a_{kj} := \arg\min_a \|y_j - ae\|_{p,\mu_{kj}}$. As X_{jk} and X'_{jk} are stochastic matrices, $X_k e = X'_k e = e$ and we can write that:

$$\limsup_{k\to\infty} |x_k| \leq K \limsup_{k\to\infty} \sum_{j=k_0}^{k-1} \gamma^{k-j}(X_{kj} - X'_{kj})(y_j - a_{kj}e).$$

By taking the absolute value we get

$$\limsup_{k\to\infty} |x_k| \leq K \limsup_{k\to\infty} \sum_{j=k_0}^{k-1} \gamma^{k-j}(X_{kj} + X'_{kj})|y_j - a_{kj}e|.$$

It can then be seen that

$$\begin{aligned}
\limsup_{k\to\infty} \left(\|x_k\|_{p,\mu}\right)^p &= K^p \limsup_{k\to\infty} \mu\left(|x_k|\right)^p \\
&\leq K^p \limsup_{k\to\infty} \mu\left[\sum_{j=k_0}^{k-1} \gamma^{k-j}(X_{kj} + X'_{kj})\left(|y_j - a_{kj}e|\right)\right]^p \\
&= K^p \limsup_{k\to\infty} \mu\left[\frac{\left(\sum_{j=k_0}^{k-1} \gamma^{k-j}\frac{1}{2}(X_{kj} + X'_{kj})2\left(|y_j - a_{kj}e|\right)\right)\left(\sum_{j=k_0}^{k-1}\gamma^{k-j}\right)}{\sum_{j=k_0}^{k-1}\gamma^{k-j}}\right]^p \\
&\leq K^p \limsup_{k\to\infty} \mu\frac{\sum_{j=k_0}^{k-1} \gamma^{k-j}\frac{1}{2}(X_{kj} + X'_{kj})\left[2|y_j - a_{kj}e|\left(\sum_{j=k_0}^{k-1}\gamma^{k-j}\right)\right]^p}{\sum_{j=k_0}^{k-1}\gamma^{k-j}} \\
&= K^p \limsup_{k\to\infty} \sum_{j=k_0}^{k-1} \gamma^{k-j}\mu_{kj}\left[2|y_j - a_{kj}e|\right]^p \left(\sum_{j=k_0}^{k-1}\gamma^{k-j}\right)^{p-1} \\
&\leq K^p \limsup_{k\to\infty} \sum_{j=k_0}^{k-1} \gamma^{k-j}\left[2\|y_j - a_{kj}e\|_{p,\mu_{kj}}\right]^p \left(\frac{\gamma}{1-\gamma}\right)^{p-1} \\
&= K^p \left(\frac{\gamma}{1-\gamma}\right)^{p-1} \limsup_{k\to\infty} \sum_{j=k_0}^{k-1} \gamma^{k-j}\left[\mathrm{span}_{p,\mu_{kj}}[y_j]\right]^p \\
&\leq K^p \left(\frac{\gamma}{1-\gamma}\right)^{p-1} \limsup_{k\to\infty} \sum_{j=k_0}^{k-1} \gamma^{k-j}\left[\sup_{k'\geq j'\geq k_0} \mathrm{span}_{p,\mu_{k'j'}}[y_{j'}]\right]^p \\
&= K^p \left(\frac{\gamma}{1-\gamma}\right)^{p-1} \frac{\gamma}{1-\gamma}\left[\sup_{k'\geq j'\geq k_0} \mathrm{span}_{p,\mu_{k'j'}}[y_{j'}]\right]^p \\
&= K^p \left(\frac{\gamma}{1-\gamma}\right)^{p} \left[\sup_{k'\geq j'\geq k_0} \mathrm{span}_{p,\mu_{k'j'}}[y_{j'}]\right]^p.
\end{aligned}$$

where x^p means the componentwise power of vector x, and where we used Jensen's inequality (with the convex function $x \mapsto x^p$) and the fact that $\sum_{j=k_0}^{k-1}\gamma^{k-j} \leq \frac{\gamma}{1-\gamma}$. As this is true for all k_0, and as $k_0 \mapsto \sup_{k'\geq\geq j'\geq k_0} \mathrm{span}_{p,\mu_{k'j'}}[y_{j'}]$ is non-increasing, the result follows.

References

[1] D. Bertsekas and S. Ioffe. Temporal differences-based policy iteration and applications in neuro-dynamic programming. Technical Report LIDS-P-2349, MIT, 1996.

[2] D.P. Bertsekas and J.N. Tsitsiklis. *Neurodynamic Programming*. Athena Scientific, 1996.

[3] G.J. Gordon. Stable function approximation in dynamic programming. In Armand Prieditis and Stuart Russell, editors, *Proceedings of the Twelfth International Conference on Machine Learning*, pages 261–268, San Francisco, CA, 1995. Morgan Kaufmann.

[4] R. Munos. Error bounds for approximate policy iteration. In *Proceedings of the International Conference on Machine Learning*, pages 560–567, 2003.

[5] R. Munos. Performance bounds in Lp norm for Approximate Value Iteration. *SIAM Journal on Control and Optimization*, 2007. To appear.

[6] R. Munos and C. Szepesvari. Finite time bounds for sampling based fitted value iteration. *Journal of Machine Learning Research*, To appear.

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

[8] S. Singh and P. Dayan. Analytical mean squared error curves for temporal difference learning. *Machine Learning Journal*, 32(1):5–40, 1998.

[9] R.S. Sutton and A.G. Barto. *Reinforcement Learning, An introduction*. BradFord Book. The MIT Press, 1998.

[10] R. Williams and L. Baird. Tight performance bounds on greedy policies based on imperfect value functions, 1993.

Unité de recherche INRIA Lorraine
LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399