Error Bounds for Approximate Value Iteration

Rémi Munos
Centre de Mathématiques Appliquées,
Ecole Polytechnique, 91128 Palaiseau Cedex, France.
remi.munos@polytechnique.fr

Abstract

Approximate Value Iteration (AVI) is an method for solving a Markov Decision Problem by making successive calls to a supervised learning (SL) algorithm. Sequence of value representations V_n are processed iteratively by $V_{n+1} = \mathcal{A}\mathcal{T}V_n$ where $\mathcal{T}$ is the *Bellman operator* and $\mathcal{A}$ an *approximation operator*. Bounds on the error between the performance of the policies induced by the algorithm and the optimal policy are given as a function of weighted L_p-norms ($p \geq 1$) of the approximation errors. The results extend usual analysis in L_∞-norm, and allow to relate the performance of AVI to the approximation power (usually expressed in L_p-norm, for $p = 1$ or 2) of the SL algorithm. We illustrate the tightness of these bounds on an optimal replacement problem.

Introduction

We study the resolution of *Markov Decision Processes* (MDPs) (Puterman 1994) using approximate value function representations V_n. The **Approximate Value Iteration** (AVI) algorithm is defined by the iteration

$$V_{n+1} = \mathcal{A}\mathcal{T}V_n, \tag{1}$$

where $\mathcal{T}$ is the *Bellman operator* and $\mathcal{A}$ an *approximation operator* or a *supervised learning* (SL) algorithm. AVI is very popular and has been successfully implemented in many different settings in Dynamic Programming (DP) (Bertsekas & Tsitsiklis 1996) and Reinforcement Learning (RL) (Sutton & Barto 1998).

A simple version is: at stage n, select a sample of states $(x_k)_{k=1\ldots K}$ from some distribution μ, compute the backed-up values $v_k := \mathcal{T}V_n(x_k)$, then make a call to a SL algorithm. This returns a function V_{n+1} minimizing some average empirical loss $V_{n+1} = \arg\min_f \frac{1}{K}\sum_k l(f(x_k) - v_k)$.

Most SL algorithms use squared (L_2) or absolute (L_1) loss functions (or variants) thus perform a minimization problem in weighted L_1 or L_2, in which the weights are defined by μ. It is therefore crucial to estimate the performance of AVI as a function of the weighted L_p- norms ($p \geq 1$) used by the SL algorithm. The goal of this paper is to extend usual results in L_∞-norm to similar results in weighted L_p-norms. The performance achieved by such a resolution of the MDP may then be directly related to the approximation power of the SL algorithm.

Alternative results in approximate DP with weighted norms include Linear Programming (de Farias & Roy 2003) and Policy Iteration (Munos 2003).

Let X be the state space assumed to be finite with N states (although the results given in this paper extend easily to continuous spaces) and A a finite action space. Let $p(x, a, y)$ be the probability that the next state is y given that the current state is x and the action is a. Let $r(x, a, y)$ be the reward received when a transition $(x, a) \to y$ occurs.

A *policy* π is a mapping from X to A. We write P^π the $N \times N-$matrix with elements $P^\pi(x, y) := p(x, \pi(x), y)$ and r^π the vector with components $r^\pi(x) := \sum_y p(x, \pi(x), y) r(x, \pi(x), y)$.

For a policy π, we define the *value function* V^π which, in the discounted and infinite horizon setting studied here, is the expected discounted sum of future rewards

$$V^\pi(x) := \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t r(x_t, a_t, x_{t+1}) | x_0 = x, a_t = \pi(x_t)\Big],$$

where $\gamma \in [0, 1)$ is a *discount factor*. V^π is the fixed point of the operator $\mathcal{T}^\pi : I\!R^N \to I\!R^N$ defined, for any vector $W \in I\!R^N$, by $\mathcal{T}^\pi W := r^\pi + \gamma P^\pi W$.

The *optimal value function* $V^* := \sup_\pi V^\pi$ is the fixed-point of the Bellman operator $\mathcal{T}$ defined, for any $W \in I\!R^N$, by

$$\mathcal{T}W(x) = \max_{a\in A} \sum_y p(x, a, y)[r(x, a, y) + \gamma W(y)].$$

We say that a policy π is *greedy with respect to* $W \in I\!R^N$, if for all $x \in X$,

$$\pi(x) \in \arg\max_{a\in A} \sum_y p(x, a, y)[r(x, a, y) + \gamma W(y)].$$

An *optimal policy* π^* is a policy greedy w.r.t. V^*.

An exact resolution method for computing V^* is the *Value Iteration* algorithm defined by the iteration $V_{n+1} = \mathcal{T}V_n$. Due to the contraction property in $L_\infty-$norm of the operator $\mathcal{T}$, the iterates V_n converge to V^* as $n \to \infty$. However, problems with a large number of states prevent us from using such exact resolution methods; we need to represent the functions with a moderate number of coefficients and perform approximate iterations such as (1).

The paper is organized as follows. We first remind some approximation results in L_∞-norm, then give componentwise bounds and use them to derive error-bounds in L_p-norms. Finally we detail some practical implementations and provide a numerical experiment for an optimal replacement problem. The main result of this paper is Theorem 1. All proofs are detailed in the Appendix.

We recall the definition of the norms: let $u \in I\!R^N$. Its L_∞-norm is $||u||_\infty := \sup_x |u(x)|$. Let μ be a distribution on X. The weighted L_p-(semi) norms (denoted by $L_{p,\mu}$) is $||u||_{p,\mu} := \left[\sum_x \mu(x)|u(x)|^p\right]^{1/p}$. Let us denote by $||\cdot||_p$ the unweighted L_p-norms (i.e. when μ is uniform).

Approximation results in L_∞-norm

Consider the **AVI algorithm** defined by (1) and write $\varepsilon_n = \mathcal{T}V_n - V_{n+1}$ the **approximation error** at stage n. In general, this algorithm does not converge, but its asymptotic behavior may be analyzed. If the approximation errors are uniformly bounded $||\varepsilon_n||_\infty \le \varepsilon$ then, a bound on the error between the asymptotic performance of the policies π_n greedy w.r.t. V_n and the optimal policy is (Bertsekas & Tsitsiklis 1996):

$$\limsup_{n\to\infty} ||V^* - V^{\pi_n}||_\infty \le \frac{2\gamma}{(1-\gamma)^2}\varepsilon. \qquad (2)$$

This L_∞-bound requires a uniformly low approximation error over all states, which is difficult to guarantee in practice, especially for large-scale problems. Most function approximation (exceptions include (Gordon 1995; Guestrin, Koller, & Parr 2001)) such as those described in the next section perform a minimization problem using weighted L_1 and L_2 norms.

Approximation operators

A **supervised learning algorithm** $\mathcal{A}$ returns a good fit g (within given classes of functions $\mathcal{F}$) of the data $(x_k, v_k) \in X_N \times I\!R$, $k = 1 \dots K$, with the x_k being sampled from some distribution μ and the values v_k being unbiased estimates of some function $f(x_k)$, by minimizing an average empirical loss $\frac{1}{K}\sum_{k=1}^K l(v_k - g(x_k))$ using mainly L_1 or L_2 loss functions (or variants). If the values are not perturbated (i.e. $v_k = f(x_k)$), $\mathcal{A}$ may be considered as an **approximation operator** that returns a compact representation $g \in \mathcal{F}$ of a general function f by minimizing some $L_{1,\mu}$ or $L_{2,\mu}$-norm. Approximation theory studies the approximation error as a function of the smoothness of f (DeVore 1997).

The projection onto the span of a fixed family of functions (called *features*) is called *linear approximation* and include *Splines*, *Radial Basis, Fourier* or *Wavelet decomposition*. A better approximation is reached when choosing the features according to f (i.e. *feature selection*). This *non-linear approximation* is particularly efficient when f has piecewise regularities (e.g. in adaptive wavelet basis (Mallat 1997) such functions are compactly represented with few non-zero coefficients). Greedy algorithms for selecting the best features among a given dictionary of functions include the *Matching Pursuit* and variants (Davies, Mallat, & Avellaneda 1997).

In Statistical Learning (Hastie, Tibshirani, & Friedman 2001), other SL algorithms include *Neural Network*, *Locally Weighted Learning* and *Kernel Regression* (Atkeson, Schaal, & Moore 1997), *Support-Vectors* and *Reproducing Kernels* (Vapnik, Golowich, & Smola 1997).

We call $\mathcal{A}$ an $\varepsilon-$**approximation operator** if $\mathcal{A}$ returns an $\varepsilon-$approximation g of f: $||f - g|| \le \varepsilon$.

Componentwise error bounds

Here we provide componentwise bounds that will be used in the next section. We consider the AVI algorithm defined by (1) and write $\varepsilon_n = \mathcal{T}V_n - V_{n+1}$ the approximation error at stage n. A componentwise bound on the asymptotic performance of the policies π_n greedy w.r.t. V_n and the optimal policy is given now (and proved in the Appendix)

Lemma 1. *We have*

$$\limsup_{n\to\infty} V^* - V^{\pi_n} \le \limsup_{n\to\infty} (I - \gamma P^{\pi_n})^{-1} \sum_{k=0}^{n-1} \gamma^{n-k} \left[(P^{\pi^*})^{n-k} + P^{\pi_n} P^{\pi_{n-1}} \dots P^{\pi_{k+1}}\right] |\varepsilon_k|, \qquad (3)$$

where $|\varepsilon_k|$ is the vector whose components are the absolute values of ε_k.

Approximation results in L_p-norms

We use the componentwise result of the previous section to extend the error bound (2) in L_p-norm, under one of the two following assumption.

Let μ be a distribution over X.

Assumption A1 [Smooth transition probabilities]. There exists a constant $C > 0$ such that, for all states $x, y \in X$, all policy π,

$$P^\pi(x, y) \le C\mu(y).$$

Assumption A2 [Smooth future state distribution]. There exits a distribution ρ and coefficients $c(m)$ such that for all $m \ge 1$ policies $\pi_1, \pi_2, \dots, \pi_m$,

$$\rho P^{\pi_1} P^{\pi_2} \dots P^{\pi_m} \le c(m)\mu. \qquad (4)$$

Then we define the *smoothness constant C of the discounted future state distribution*

$$C := (1-\gamma)^2 \sum_{m\ge 1} m\gamma^{m-1} c(m).$$

Assumption A1 was introduced in (Munos 2003) for deriving performance bounds in policy iteration algorithms. We notice that A1 is stronger than A2 since when A1 holds, A2 also holds for any distribution ρ (with the same constant C). A1 concerns the immediate transition probabilities (an example for which A1 holds is the optimal replacement problem described below) whereas A2 expresses some smoothness property of the future state distribution w.r.t. μ when initial state is drawn from some distribution ρ. Indeed, in terms of Markov chain, assumption A2 implies that for any sequence of policies $\pi_1, \dots, \pi_m$, the discounted future state distribution starting from ρ is bounded by a constant C times μ: for all x_0, y in X,

$$(1-\gamma)^2 \sum_{m=1}^{\infty} m\gamma^{m-1} \Pr\big(x_m = y \big| x_0 \sim \rho,$$
$$x_i \sim p(x_{i-1}, \pi_i(x_{i-1}), \cdot), 1 \le i \le m\big) \le C\mu(y).$$

L_p error bounds

At each iteration of the AVI algorithm the new function V_{n+1} is obtained by approximating $\mathcal{T}V_n$ via a call to a SL algorithm $\mathcal{A}$, which solves a minimization problem in $L_{p,\mu}$-norm. Our main result relates the performance of AVI as a function of the approximation errors using the same norm as that used by the SL algorithm.

Theorem 1. *Let μ be a distribution on X. Let $\mathcal{A}$ be an $\varepsilon-$approximation operator in $L_{p,\mu}$-norm ($p \ge 1$) (i.e. for all $n \ge 0$, $||\varepsilon_n||_{p,\mu} \le \varepsilon$). Then:*

- *Given assumption A1, we have*
$$\limsup_{n\to\infty} ||V^* - V^{\pi_n}||_\infty \le \frac{2\gamma}{(1-\gamma)^2} C\varepsilon. \quad (5)$$
- *Given assumption A2, we have*
$$\limsup_{n\to\infty} ||V^* - V^{\pi_n}||_{p,\rho} \le \frac{2\gamma}{(1-\gamma)^2} C^{1/p}\varepsilon. \quad (6)$$

Let us give some insight about the constant C for a uniform distribution $\mu = (\frac{1}{N} \dots \frac{1}{N})$. Here assumption A1 may always be satisfied by choosing $C = N$ (but then, (5) is not better than (2), since $||\varepsilon_n||_\infty \le N^{1/p}||\varepsilon_n||_p$). We are thus interested in finding a constant $C \ll N$. An interesting case for which this happens is when the state space is continuous and the transition densities are upper bounded (this will be illustrated in the numerical experiments below).

Let us give some insight about the constant C in the case of assumption A2 when ρ and μ are uniform.

- The largest possible value of C is obtained in a MDP where for a specific policy π all states jump to a given state -say state 1- with probability 1. Thus $\rho(P^\pi)^m = (1\,0 \dots 0) \le N\mu$. Thus for all m, $c(m) = N$ and
$$C = (1-\gamma^2)\sum_{i\ge0}\gamma^i \sum_{j\ge0}\gamma^j h(i+j+1) = N.$$
This is the worst case. In that case, (5) may be derived from the L_∞ bound (2) since $\limsup_{n\to\infty} ||V^* - V^{\pi_n}||_{p,\rho} \le \limsup_{n\to\infty} ||V^* - V^{\pi_n}||_\infty \le \frac{2\gamma}{(1-\gamma)^2} N^{1/p}\varepsilon$.
- The lowest possible value of C is obtained in a MDP with uniform transition probabilities $P^\pi(x,y) = 1/N$ (or more generally if $\rho = \mu$ is the steady-state distribution of P^π). Here $c(m) = 1$ and $C = 1$.

Thus the smoothness constant expresses how picky the discounted future state distribution is, compared to μ. A low C means that the mass of the future state distribution starting from ρ does not accumulate on few specific states for which the distribution μ is low. For that purpose, it is desirable that μ be somehow uniformly distributed (this condition was already mentioned in (Koller & Parr 2000; Kakade & Langford 2002; Munos 2003) to secure Policy Improvement steps in Approximate Policy Iteration).

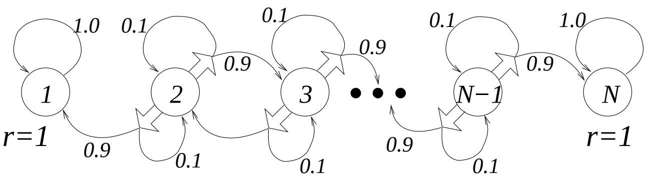

Figure 1: The chain walk MDP.

Illustration on the *chain walk MDP*

We illustrate the fact that the L_p-norm (for $p = 1$ and 2) bounds given in Theorem 1 under assumption A2 may be much tighter than the $L_\infty-$norm (2) on the *chain walk* MDP defined in (Lagoudakis & Parr 2003) (see Figure 1).

This is a linear chain with N states with two dead-end states: states 1 and N. On each of the interior states $2 \le x \le N-1$ there are two possible actions: right or left, which changes the state in the intended direction with probability 0.9, and fails with probability 0.1 changing the state in the opposite direction. The reward simply depends on the current state and is 1 at boundary states and 0 elsewhere: $r = (1\,0 \dots 0\,1)'$.

We consider an approximation of the value function of the form $V_n(x) = \alpha_n + \beta_n x$ where $x \in \{1, \dots, N\}$ is the state number. Assume that the initial approximation is zero: $V_0 = (0 \dots 0)'$. Then $\mathcal{T}V_0 = (1\,0 \dots 0\,1)'$. The best fit in L_∞-norm is a constant function $V_1 = (\frac{1}{2} \dots \frac{1}{2})'$ which produces an error $||V_1 - \mathcal{T}V_0||_\infty = \frac{1}{2}$.

Let us choose uniform distributions $\mu = (\frac{1}{N} \dots \frac{1}{N})$. In L_1-norm we find that the best fit is $V_1 = (0 \dots 0)'$ (for $N > 4$) and the resulting error is $||V_1 - \mathcal{T}V_0||_1 = \frac{2}{N}$. In L_2-norm the best fit is also constant $V_1 = (\frac{2}{N} \dots \frac{2}{N})'$ and the error is $||V_1 - \mathcal{T}V_0||_2 = \frac{\sqrt{2N-4}}{N}$.

In the three cases, by induction, we observe that the successive approximations V_n are constant, thus $\mathcal{T}V_n = r + \gamma V_n$ and the approximation errors remain the same as in the first iteration: $||V_{n+1} - \mathcal{T}V_n||_\infty = \frac{1}{2}$, $||V_{n+1} - \mathcal{T}V_n||_1 = \frac{2}{N}$, and $||V_{n+1} - \mathcal{T}V_n||_2 = \frac{\sqrt{2N-4}}{N}$.

Since V_n is constant, any policy π_n is greedy w.r.t. V_n. Hence for $\pi_n = \pi^*$ the l.h.s. of (2) and (6) are equal to zero. Now, in order to compare the r.h.s. of these inequalities, let us calculate the constant C under Assumptions A1 and A2. Since state 1 jumps to itself with probability 1, under A1, we have no better constant than $C = N$.

Under A2, the worst case in (4) is obtained when the mass of the future state distribution is mostly concentrated on one boundary state -say state 1- which corresponds to a policy π_{Left} that chooses everywhere action left. We see that for $\rho = \mu$,

$$\rho(P^{\pi_{\text{Left}}})^m(x) \le \rho(P^{\pi_{\text{Left}}})^m(1) \le (1 + 0.9m)\mu(x),$$

for all $x \ge 0$, thus Assumption A2 is satisfied with $c(m) = 1 + 0.9m$. We deduce that the constant $C = (1-\gamma)^2 \sum_{m\ge1} m\gamma^{m-1}(1 + 0.9m)$ *is independent from the number of states* N.

Thus, when the number of states N is large, the L_1-norm bound provides an approximation of order $O(N^{-1})$, the

L_2-norm bound is of order $O(N^{-1/2})$, whereas the $L_\infty-$norm bound (2) is only of order $O(1)$.

Notice that here, we used the same norms for the minimization problem (function fitting) as those used for the bounds. If, say, a L_2-norm were used for minimization, then this would provide even worst L_∞ error bounds.

Practical algorithms

Model-based AVI

Let μ be a distribution over X. Given $\varepsilon > 0$ and an $\varepsilon-$approximation operator $\mathcal{A}$ in $L_{p,\mu}$-norm (for $p \geq 1$). A model-based version of AVI would consider, at each iteration, the following steps:

1. Select set of states $x_k \in X_N$, $k = 1 \dots K$, sampled from the distribution μ,
2. Compute the backed-up values $v_k = \mathcal{T}V_n(x_k)$,
3. Make a call to the supervised learning algorithm $\mathcal{A}$ with the data $\{x_k; v_k\}$, which returns an $\varepsilon-$approximation V_{n+1}.

Reinforcement Learning

Step 2 in the preceding algorithm requires the knowledge of a model of the transition probabilities (as well as a way to compute the expectations in the operator $\mathcal{T}$). If this is not the case, one may consider using a Reinforcement Learning (RL) algorithm (Sutton & Barto 1998). Let us introduce the Q-values and the operator $\mathcal{R}$ defined on $X_N \times A$,

$$\mathcal{R}Q(x,a) := \sum_{y \in X} p(x,a,y)\big[r(x,a,y) + \gamma \max_{b \in A} Q(y,b)\big].$$

The AVI algorithm is equivalent to defining successive approximations Q_n according to

$$Q_{n+1} = \mathcal{A}\mathcal{R}Q_n,$$

where $\mathcal{A}$ is an approximation operator on $X \times A$. Thus, a model-free RL algorithm would be defined by the steps:

1. Observe a set of transitions: $(x_k, a_k) \overset{r_k}{\to} y_k$, $k = 1 \dots K$, where for current state x_k and action a_k, $y_k \sim p(x_k, a_k, \cdot)$ is the next observed state and r_k the received reward,
2. Compute the values $v_k = r_k + \gamma \max_b Q_n(y_k, b)$,
3. Make a call to the supervised learning algorithm $\mathcal{A}$ with the data $\{(x_k, a_k); v_k\}$, which returns an $\varepsilon-$approximation estimate $\widehat{Q_{n+1}}$.

An interesting case is when $\mathcal{A}$ is a linear operator *in the values* $\{v_k\}$ (which implies that the operators $\mathcal{A}$ and $\mathbb{E}$ commute) such as in Least Squares Regression, k-Nearest Neighbors, Locally Weighted Learning. Then the approximation $\widehat{Q_{n+1}}$ returned by $\mathcal{A}$ is an unbiased estimate of $\mathcal{A}\mathcal{R}Q_n$ (since the values $\{v_k\}$ are unbiased estimates of $\mathcal{R}Q_n(x_k, a_k)$). Thus when K is large, such an iteration acts like a (model-based) AVI iteration, and bounds similar to those of Theorem 1 may be derived.

Notice that the policy derived from the approximate Q-values: $\pi'_n(x) \in \arg\max_a Q_n(x,a)$ is different from the policy π_n greedy w.r.t. V_n, defined by $V_n(x) := \max_a Q_n(x,a)$. However, bounds similar to (2), (5), and (6) on the performance of these policies π'_n may be derived analogously. For example, one may prove that the L_∞ bound is

$$\limsup_{n \to \infty} ||V^* - V^{\pi'_n}||_\infty \leq \frac{2}{(1-\gamma)^2}\varepsilon.$$

(there is an additional error of $2\varepsilon/(1-\gamma)$ compared to (2)).

Experiment: an optimal replacement problem

This experiment illustrate the respective tightness of the L_∞, L_1, and L_2 norm bounds on a discretization (for several resolutions) of a continuous space control problem derived from (Rust 1996).

A one-dimensional continuous variable $x_t \in I\!\!R_+$ measures the accumulated utilization (such as the odometer reading on a car) of a durable. $x_t = 0$ denotes a brand new durable. At each discrete time t, there are two possible decisions: either keep ($a_t =$ K) or replace ($a_t =$ R), in which case an additional cost $C_{replace}$ (of selling the existing durable and replacing it for a new one) occurs.

The transition density functions are exponential with parameter β:

$$\begin{array}{rcl} p(x, a = \mathrm{K}, y) & = & \left\{ \begin{array}{ll} \beta e^{-\beta(y-x)} & \text{if } y \geq x \\ 0 & \text{if } y < x, \end{array} \right. \\ p(x, a = \mathrm{R}, y) & = & \left\{ \begin{array}{ll} \beta e^{-\beta y} & \text{if } y \geq 0 \\ 0 & \text{if } y < 0. \end{array} \right. \end{array} \tag{7}$$

Moreover, if the next state y is greater than a maximal value $x_{\max}$ (e.g. the car breaks down because it is too damaged) then a new state is immediately redrawn according to $p(x, a = \mathrm{R}, \cdot)$ and a penalty $C_{dead} > C_{replace}$ occurs. The current cost (opposite of a reward) $c(x)$ is the sum of a slowly increasing function (maintenance cost) and a short-term discontinuous and periodic additional cost (e.g. which may represent car insurance fees).

The current cost function and the value function (computed by a discretization on a high resolution grid) are shown on Figure 1a.

We choose the numerical values $\gamma = 0.6$, $\beta = 0.6$, $C_{replace} = 50$, $C_{dead} = 70$, and $x_{\max} = 10$. Two finite state space MDPs (with $N = 200$ or 2000 points) are generated by discretizing the continuous space problem over the domain $[0, x_{\max}]$ using uniform grids $\{x_k := kx_{\max}/N\}_{0 \leq k < N}$ with N points.

We consider a linear approximation on the space spanned by the truncated cosine basis

$$\mathcal{F} := \Big\{ f(x) = \textstyle\sum_{m=1}^{M} \alpha_m \cos(m\pi \frac{x}{x_{\max}}) \Big\},$$

with $M = 20$ coefficients. We start with initial values $V_0 = 0$. At each iteration, the fitted function $V_{n+1} \in \mathcal{F}$ is such as to minimise the L_2 error:

$$V_{n+1} = \arg\min_{f \in \mathcal{F}} \frac{1}{N} \textstyle\sum_{k=1}^{N} [f(x_k) - \mathcal{T}V_n(x_k)]^2.$$

In Figure 1.b we show the first iteration (for the grid with 200 points): the backed-up values $\mathcal{T}V_0$ (indicated with

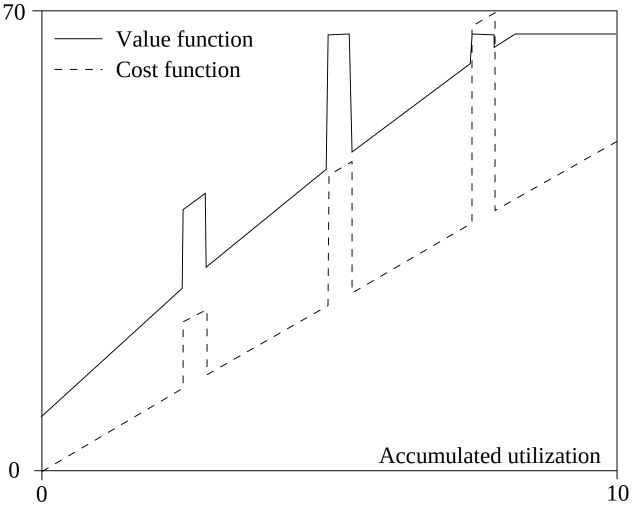

Figure 2: Cost and value functions.

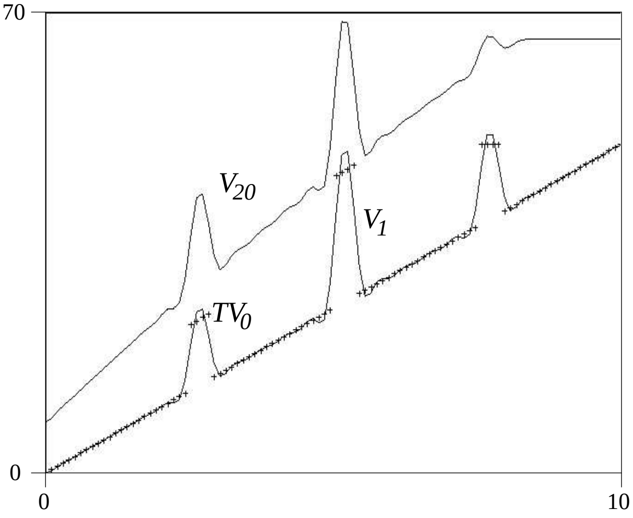

Figure 3: $\mathcal{T}V_0$ (crosses), V_1 and V_{20}.

crosses), the corresponding approximation V_1 (best fit of $\mathcal{T}V_0$ in the cosine approximation space). The approximate value function computed after 20 iterations (when there are no more improvement in the approximations) is also plotted.

Here, the highest pick in the future state distribution occurs at $x = 0$ for a policy that would always chose action R. By our choice of $x_{\max}$, $\int_{x_{\max}}^{\infty} p(0, R, y)dy$ is negligible. Thus, Assumption A1 is satisfied (as well as A2) with $C = \beta x_{\max} = 6$.

	$\lvert\lvert\varepsilon_n\rvert\rvert_\infty$	$C\lvert\lvert\varepsilon_n\rvert\rvert_1$	$C^{1/2}\lvert\lvert\varepsilon_n\rvert\rvert_2$
$N = 200$	12.4	0.367	1.16
$N = 2000$	12.4	0.0552	0.897

Table 1: Comparison of the L_∞, L_1 and L_2 bounds.

Table 1 compares the right hand side of equations (2), (5), and (6) (up to the constant $2\gamma/(1-\gamma)^2$). We notice that the L_1 and L_2 bounds are much tighter than the L_∞ one, and decrease when the number of grid points tends to infinity (asymptotic behavior similar to the chain walk MDP), whereas the L_∞ bound does not. Indeed, when the number of states N goes to infinity, the discrete MDP gets closer to the continuous problem, and since the cost function is discontinuous, the L_∞ approximation error (using continuous function approximation such as the cosine basis) will never be lower than half the value of the largest discontinuity.

Conclusion

Theorem 1 provides a useful tool to relate the performance of AVI to the approximation power of the SL algorithm. Expressing the performance of AVI in the same norm as that used by the supervised learner to minimize the approximation error guarantees the tightness of this bound.

Extension to other loss functions l, such as *ϵ-insensitive* (used in Support Vectors) or *Huber loss function* (for robust regression) is straightforward (as long as l is an increasing and convex function over $\mathbb{R}^+$).

Other possible extensions include Markov games and on-line reinforcement learning.

Appendix: proof of the results

Proof of Lemma 1

Since $\mathcal{T}V_k \geq \mathcal{T}^{\pi^*}V_k$ and $\mathcal{T}V^* \geq \mathcal{T}^{\pi_k}V^*$, we have

$$\begin{aligned}
V^* - V_{k+1} &= \mathcal{T}^{\pi^*}V^* - \mathcal{T}^{\pi^*}V_k + \mathcal{T}^{\pi^*}V_k - \mathcal{T}V_k + \varepsilon_k \\
&\leq \gamma P^{\pi^*}(V^* - V_k) + \varepsilon_k \\
V^* - V_{k+1} &= \mathcal{T}V^* - \mathcal{T}^{\pi_k}V^* + \mathcal{T}^{\pi_k}V^* - \mathcal{T}V_k + \varepsilon_k \\
&\geq \gamma P^{\pi_k}(V^* - V_k) + \varepsilon_k,
\end{aligned}$$

from which we deduce by induction

$$\begin{aligned}
V^* - V_n \leq &\sum_{k=0}^{n-1} \gamma^{n-k-1}(P^{\pi^*})^{n-k-1}\varepsilon_k \\
&+\gamma^n(P^{\pi^*})^n(V^* - V_0) \qquad (8) \\
V^* - V_n \geq &\sum_{k=0}^{n-1} \gamma^{n-k-1}(P^{\pi_{n-1}}P^{\pi_n}\dots P^{\pi_{k+1}})\varepsilon_k \\
&+\gamma^n(P^{\pi_n}P^{\pi_{n-1}}\dots P^{\pi_1})(V^* - V_0). \qquad (9)
\end{aligned}$$

Now, from the definition of π_k and since $\mathcal{T}V_n \geq \mathcal{T}^{\pi^*}V_n$, we have:

$$\begin{aligned}
&V^* - V^{\pi_n} \\
&= \mathcal{T}^{\pi^*}V^* - \mathcal{T}^{\pi^*}V_n + \mathcal{T}^{\pi^*}V_n - \mathcal{T}V_n + \mathcal{T}V_n - \mathcal{T}^{\pi_n}V^{\pi_n} \\
&\leq \gamma P^{\pi^*}(V^* - V_n) + \gamma P^{\pi_n}(V_n - V^* + V^* - V^{\pi_n}),
\end{aligned}$$

thus $(I - \gamma P^{\pi_n})(V^* - V^{\pi_n}) \leq \gamma(P^{\pi^*} - P^{\pi_n})(V^* - V_n)$. Now, since $(I - \gamma P^{\pi_n})$ is invertible and its inverse $\sum_{k\geq 0}(\gamma P^{\pi_n})^k$ has positive elements, we deduce

$$V^* - V^{\pi_n} \leq \gamma(I - \gamma P^{\pi_n})^{-1}(P^{\pi^*} - P^{\pi_n})(V^* - V_n).$$

This, combined with (8) and (9), and after taking the absolute value, yields

$$\begin{aligned}
&V^* - V^{\pi_n} \le (I - \gamma P^{\pi_n})^{-1} \\
&\Big\{ \sum_{k=0}^{n-1} \gamma^{n-k} \big[(P^{\pi^*})^{n-k} + (P^{\pi_n} P^{\pi_{n-1}} \dots P^{\pi_{k+1}}) \big] |\varepsilon_k| \qquad (10) \\
&+\gamma^{n+1} \big[(P^{\pi^*})^{n+1} + (P^{\pi_n} P^{\pi_n} P^{\pi_{n-1}} \dots P^{\pi_1}) \big] |V^* - V_0| \Big\}.
\end{aligned}$$

We deduce (3) by taking the upper limit. □

Proof of Theorem 1

We have seen that if assumption A1 holds, then A2 also holds for any distribution ρ. Now, for $p = 1$, if the bound (6) holds for any ρ, then (5) also holds. Thus, we only need to prove (6) in the case of assumption A2.

We may rewrite (10) as

$$V^* - V^{\pi_n} \le \frac{2\gamma(1-\gamma^{n+1})}{(1-\gamma)^2} \Big[\sum_{k=0}^{n-1} \alpha_k A_k |\varepsilon_k| + \alpha_n A_n |V^* - V_0| \Big],$$

with the positive coefficients ($0 \le k < n$)

$$\alpha_k := \frac{(1-\gamma)\gamma^{n-k-1}}{1-\gamma^{n+1}} \text{ and } \alpha_n := \frac{(1-\gamma)\gamma^n}{1-\gamma^{n+1}},$$

(defined such that they sum to 1) and the stochastic matrices

$$A_k := \tfrac{1-\gamma}{2} (I - \gamma P^{\pi_n})^{-1} [(P^{\pi^*})^{n-k} + (P^{\pi_n} \dots P^{\pi_{k+1}})]$$
$$A_n := \tfrac{1-\gamma}{2} (I - \gamma P^{\pi_n})^{-1} [(P^{\pi^*})^{n+1} + (P^{\pi_n} P^{\pi_n} \dots P^{\pi_1})].$$

We have

$$\begin{aligned}
||V^* - V^{\pi_n}||_{p,\rho}^p &\le \Big[\frac{2\gamma(1-\gamma^{n+1})}{(1-\gamma)^2}\Big]^p \sum_{x \in X} \rho(x) \\
&\qquad \Big[\sum_{k=0}^{n-1} \alpha_k A_k |\varepsilon_k| + \alpha_n A_n |V^* - V_0| \Big]^p (x) \\
&\le \Big[\frac{2\gamma(1-\gamma^{n+1})}{(1-\gamma)^2}\Big]^p \sum_{x \in X} \rho(x) \qquad (11) \\
&\qquad \Big[\sum_{k=0}^{n-1} \alpha_k A_k |\varepsilon_k|^p + \alpha_n A_n |V^* - V_0|^p \Big] (x),
\end{aligned}$$

by using two times Jensen's inequality (since the coefficients $\{\alpha_k\}_{0 \le k \le n}$ sum to 1 and the matrix A_k are stochastic) (i.e. convexity of $x \to |x|^p$). The second term in the brackets will disappear when taking the upper limit. Now, from assumption A2, $\rho A_k \le (1-\gamma) \sum_{m \ge 0} \gamma^m c(m+n-k)\mu$, thus the first term $\sum_x \rho(x) \sum_{k=0}^{n-1} \alpha_k A_k |\varepsilon_k|^p(x)$ in (11) is bounded by

$$\begin{aligned}
&\sum_{k=0}^{n-1} \alpha_k (1-\gamma) \sum_{m \ge 0} \gamma^m c(m+n-k) ||\varepsilon_k||_{p,\mu}^p \\
&\le \frac{(1-\gamma)^2}{1-\gamma^{n+1}} \sum_{m \ge 0} \sum_{k=0}^{n-1} \gamma^{m+n-k-1} c(m+n-k) \varepsilon^p \\
&\le \frac{1}{1-\gamma^{n+1}} C \varepsilon^p,
\end{aligned}$$

where we replaced α_k by their values, and used the fact that $||\varepsilon_k||_{p,\mu} \le \varepsilon$. By taking the upper limit in (11), we deduce (6). □

References

Atkeson, C. G.; Schaal, S. A.; and Moore, A. W. 1997. Locally weighted learning. *AI Review* 11.

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

Davies, G.; Mallat, S.; and Avellaneda, M. 1997. Adaptive greedy approximations. *J. of Constr. Approx.* 13:57–98.

de Farias, D., and Roy, B. V. 2003. The linear programming approach to approximate dynamic programming. *Operations Research* 51(6).

DeVore, R. 1997. *Nonlinear Approximation*. Acta Numerica.

Gordon, G. 1995. Stable function approximation in dynamic programming. *Proceedings of the International Conference on Machine Learning*.

Guestrin, C.; Koller, D.; and Parr, R. 2001. Max-norm projections for factored mdps. *Proceedings of the International Joint Conference on Artificial Intelligence*.

Hastie, T.; Tibshirani, R.; and Friedman, J. 2001. *The Elements of Statistical Learning*. Springer Series in Statistics.

Kakade, S., and Langford, J. 2002. Approximately optimal approximate reinforcement learning. *Proceedings of the 19th International Conference on Machine Learning*.

Koller, D., and Parr, R. 2000. Policy iteration for factored mdps. *Proceedings of the 16th conference on Uncertainty in Artificial Intelligence*.

Lagoudakis, M., and Parr, R. 2003. Least-squares policy iteration. *Journal of Machine Learning Research* 4:1107–1149.

Mallat, S. 1997. *A Wavelet Tour of Signal Processing*. Academic Press.

Munos, R. 2003. Error bounds for approximate policy iteration. *19th International Conference on Machine Learning*.

Puterman, M. L. 1994. *Markov Decision Processes, Discrete Stochastic Dynamic Programming*. A Wiley-Interscience Publication.

Rust, J. 1996. *Numerical Dynamic Programming in Economics*. In Handbook of Computational Economics. Elsevier, North Holland.

Sutton, R. S., and Barto, A. G. 1998. Reinforcement learning: An introduction. *Bradford Book*.

Vapnik, V.; Golowich, S. E.; and Smola, A. 1997. Support vector method for function approximation, regression estimation and signal processing. *In Advances in Neural Information Processing Systems* 281–287.