Temporal difference learning with kernels for pricing American-Style option

Kengy Barty[1], Jean-Sébastien Roy[2], Cyrille Strugarek[3]

May 12, 2005

Abstract

We propose in this paper to study the problem of estimating the cost-to-go function for an infinite-horizon discounted Markov chain with possibly continuous state space. For implementation purposes, the state space is typically discretized. As soon as the dimension of the state space becomes large, the computation is no more practicable, a phenomenon referred to as the curse of dimensionality. The approximation of dynamic programming problems is therefore of major importance.

A powerful method for dynamic programming, often referred to as neuro-dynamic programming, consists in representing the Bellman function as a linear combination of a priori defined functions, called neurons. The choice of the neurons represents a delicate operation since it requires to have an idea of the optimal solution. Furthermore, in a classical learning algorithm once the choice of these neurons is made it is no longer modified although the amount of the available information concerning the solution increases along the iterations. In other words, such algorithms are "locked" in the vector subspace generated by these neurons. Consequently, the algorithm is not able to reach the optimal solution if it does not belong to the neurons vector subspace.

In this article, we propose an alternative approach very similar to temporal differences, based on functional gradient descent and using an infinite kernel basis. Our algorithm is a result of the combination of stochastic approximation ideas, and nonparametric estimation concepts. Furthermore, our algorithm, though aimed at infinite dimensional problems, is implementable in practice. We prove the convergence of this algorithm under a few conditions, which are classical in stochastic approximation schemes. We conclude by showing on examples how this algorithm can be used to solve both infinite-horizon discounted Markov chain problems, and bermudan option pricing.

Keywords: TD Learning, Robbins-Monro Algorithm, Kernel Approximation, Approximate Dynamic Programming

1. Introduction

Dynamic programming is a powerful methodology for dealing with problems of sequential decision-making under uncertainty. In the case of a continuous system state, the usual approach to apply dynamic programming is to perform a discretization of the state and recursively apply the Bellman operator. This discretization usually leads to very large state spaces, a problem known as the curse of dimensionality. An additionnal complexity arises in the stochastic case, since the conditional expectation appearing in the Bellman equation must also be approximated through a discretization of the dynamic.

Temporal difference learning introduced by Sutton[Sutton, 1988] provides a way to carry out the Bellman operator fixed point iterations while approximating the expectation through random sampling. While solving the second problem, this approach still requires a discretization of the state space which, in the large scale case, might not be practicable. To overcome the

[1]**École Nationale des Ponts et Chaussées** (ENPC),
kengy.barty@cermics.enpc.fr
[2]**EDF R&D**
1, avenue du Général de Gaulle
F-92141 Clamart Cedex
jean-sebastien.roy@edf.fr
[3]**EDF R&D**
cyrille.strugarek@edf.fr
also with the École Nationale Supérieure de Techniques Avancées (ENSTA) and the École Nationale des Ponts et Chaussées (ENPC)

curse of dimensionality most approaches so far have proposed to approximate the value function as a linear combination of basis functions. This approach, called approximate dynamic programming, and first described in [Bellman and Dreyfus, 1959], has been thoroughly studied. See [Sutton and Barto, 1998] and [Bertsekas and Tsitsiklis, 1996] for detailed introductions to temporal difference and approximate dynamic programming methods. Recent and promising approaches to this problem include a formulation of dynamic programming through a linear program which can ensure performance guarantees [de Farias and Van Roy, 2004]. Nevertheless, all these approaches require the use of a predefined finite functional basis and therefore give up optimality, even asymptotically. Moreover, while the quality of the approximation might increase with the number of functions used in the basis, the complexity of each iteration (usually a least-square regression or linear program), renders the use of large basis impracticable.

We introduce an alternative approach, based on functional gradient descent and using an infinite kernel basis, that preserves optimality under very light conditions while being implementable in practice. In contrast to finite functional basis methods, where the a priori basis is used to abitrarily generalize the local gradient information provided by each sample, we aim at generalizing using only regularity assumptions of the value fonction and therefore better exploiting the information provided.

Similar ideas dates back to recursive nonparametric density estimation [Wolverton and Wagner, 1969], and have been proposed in the context of econometry in [Chen and White, 1998]. Our approach aims at providing more sensible assumptions in the context of optimization and simpler proofs, based on a completely different theory.

Section 2 describes our new algorithm to approximate the Bellman equation and shows its convergence. Section 3 establishes a link between the Robbins-Monro stochastic approximation [Robbins and Monro, 1951] and our algorithm. As an application, this analysis shows how our algorithm is a generalization of classical temporal difference schemes in infinite dimensional framework. Finally two numerical examples are presented in section 4, one of them being Bermudan option pricing.

2. Learning with kernels

We consider the problem of approximating the cost-to-go function for an infinite-horizon discounted Markov chain with possibly continuous state space. Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space, $(S, \mathcal{B})$ be a topological space endowed with its Borel σ-field and $(X_t)_{t\in\mathbb{N}}$ be a Markov chain with values on the state space S. Under these assumptions there exist transition kernels describing the dynamics of the Markov chain. We also suppose that the Markov chain is stationary, i.e., its transition kernels are time-independent. We can hence define $\Pi : S \times \mathcal{B} \to [0, 1]$ to be the transition kernel of the Markov chain $(X_t)_{t\in\mathbb{N}}$, by:

$$\forall t \in \mathbb{N}, \quad \forall x \in S, \quad \forall A \in \mathcal{B}, \quad \Pi(x, A) = \mathbb{P}(X_t \in A \mid X_0 = x).$$

Assumption 2.1. There exists a measure denoted $\pi : \mathcal{B} \to [0, 1]$ such that:

$$\forall A \in \mathcal{B}, \quad \pi(A) = \int_S \Pi(x, A)\pi(dx).$$

The previous equality implies that π is an invariant probability measure for the Markov chain considered. Such a probability measure is often referred to as the steady-state probability.

We endow the space of square π-integrable random variables denoted by $L^2(S, \mathcal{B}, \pi)$ with the inner product $\langle\cdot,\cdot\rangle_\pi$:

$$\forall u, v \in L^2(S, \mathcal{B}, \pi), \quad \langle u, v\rangle_\pi = \int_S u(x)v(x)\pi(dx),$$

and with the following norm $\|\cdot\|_\pi$ as well:

$$\forall v \in L^2(S, \mathcal{B}, \pi), \quad \|v\|_\pi = \sqrt{\langle v, v\rangle_\pi}.$$

To simplify notations let us denote:

$$\forall f \in L^2(S,\mathcal{B},\pi), \forall x \in S, \quad \Pi(f)(x) = \int_S f(y)\Pi(x,dy) \quad \text{and,} \quad \pi(f) = \int_S f(y)\pi(dy).$$

We will also write $\mathbb{E}[v] = \int_\Omega v(\omega)\mathbb{P}(d\omega)$.

Let $g : S \to \mathbb{R}$ be a bounded function. For a given $\alpha \in [0,1[$, we define the cost-to-go function J^* as follows:

$$J^*(x) = \mathbb{E}\left[\sum_{t=0}^{n} \alpha^t g(X_t) \mid X_0 = x\right].$$

J^* is the unique solution to Bellman's equation:

$$J = TJ, \tag{2.1}$$

where $T : L^2(S,\mathcal{B},\pi) \to L^2(S,\mathcal{B},\pi)$ is given by:

$$\forall J \in L^2(S,\mathcal{B},\pi),\ TJ = g + \alpha\Pi(J), \tag{2.2}$$

which also reads, for all $J \in L^2(S,\mathcal{B},\pi)$:

$$\forall x \in S, \quad (TJ)(x) = \int_S (g(x) + \alpha J(y))\Pi(x,dy).$$

One can remark that Bellman's operator T is α-Lipschitz continuous for the previously defined norm:

$$\begin{aligned}
\forall J, \bar{J} \in L^2(S,\mathcal{B},\pi), \quad \left\|TJ - T\bar{J}\right\|_\pi^2 &= \int_S \left(g(x) + \alpha\Pi(J)(x) - g(x) - \alpha\Pi(\bar{J})(x)\right)^2 \pi(dx), \\
&= \alpha^2 \int_S \left(\Pi(J-\bar{J})(x)\right)^2 \pi(dx), \\
&\leq \alpha^2 \int_S \Pi(\left(J-\bar{J}\right)^2)(x)\pi(dx), \quad \text{by Jensen's inequality,} \\
&\leq \alpha^2 \pi\left(\Pi((J-\bar{J})^2)\right), \\
&\leq \alpha^2 \pi (J-\bar{J})^2, \\
&\leq \alpha^2 \left\|J-\bar{J}\right\|_\pi^2.
\end{aligned}$$

This contraction property of the operator T ensures that the solution J^* of (2.1) is well-defined.

In a classical approach a discrete formulation of the problem is provided by introducing a linear combination of prescribed basis functions [Tsitsiklis and Van Roy, 1997] to represent the Bellman function. Its main drawback is the loss of any optimality guarantee: such approaches are known to converge to the optimal linear combination of the prescribed basis, but the evaluation of the deviation from the optimal solution is still open.

In order to avoid such an optimality loss, we present a new algorithm to approximate the solution of (2.1) and show its convergence. The main advantage of this algorithm is that it provides a method to incrementally increase the number of neurons while it improves its accuracy as well. As long as the number of iterations grows, we build a sum of applications where each new element contributes to reduce the distance to the optimal solution.

A description of our infinite dimensional TD(0) algorithm can be given by:

Algorithm 2.2 (Infinite dimensional TD(0))**.** Step -1 : initialize $J_0(\cdot) = 0$,
Step $k \geq 0$:

- Draw ξ_{k+1} independently from the past draws with respect to the distribution π and w_{k+1} with respect to the distribution $\Pi(\xi_{k+1},\cdot)$;
- Update :

$$d_k(\xi,w) := g(\xi) + \alpha J_k(w) - J_k(\xi),$$

$$J_{k+1}(\cdot) := J_k(\cdot) + \gamma_k d_k(\xi_{k+1},w_{k+1})K_k(\xi_{k+1},\cdot). \tag{2.3}$$

- If a maximal iteration number is reached, stop, else increment k and loop.

Where $\forall k \in \mathbb{N}, \quad K_k : S \times S \to \mathbb{R}$ is a predefined sequence of mappings. For example, consider a nonnegative sequence (ε_k) decreasing to 0, let $S = \mathbb{R}^n$, and $V \in \mathbb{R}^n \times \mathbb{R}^n$ an inversible matrix then an adequate mapping K_k is the Gaussian kernel:

$$K_k : (x, y) \to \left(\frac{1}{\sqrt{2\pi}}\right)^n e^{-\frac{1}{2\varepsilon_k}(x-y)'V^{-1}(x-y)}$$

Remark 2.3 (Sample space)**.** The sequence $(J_k)_{k\in\mathbb{N}}$ is a stochastic process defined on the sample space $(\Omega^{\otimes\mathbb{N}}, \mathcal{F}^{\otimes\mathbb{N}}, \mathbb{P}^{\otimes\mathbb{N}})$ with values in the Hilbert space $(L^2(\Xi, \mathcal{B}, \pi), \|\cdot\|_\pi)$. We denote $\mathcal{F}_k$ the complete σ-field generated by the random variable $(\xi_1, \ldots, \xi_k)$ and by $\mathbb{E}^k[\cdot]$, the conditional expectation according to $\mathcal{F}_k$. In the one hand, one can observe that J_k is $\mathcal{F}_k$-measurable, in the other hand:

$$\mathbb{E}^k\left[J_k(\xi_{k+1})^2\right] = \|J_k\|_\pi^2 \tag{2.4}$$

The core of the algorithm is provided by the kernels K_k allowing to obtain a functional update of J_k. Algorithm 2.2 can be viewed as a variant of the algorithm TD(0) since we have here a functional temporal difference D_k:

$$D_k(\xi, w)(\cdot) = \frac{1}{\varepsilon_k} d_k(\xi, w) K_k(\xi, \cdot),$$

$$J_{k+1} = J_k + \gamma_k \varepsilon_k D_k(\xi_{k+1}, w_{k+1}).$$

In the classical point of view, temporal differences are the realizations of random variables (d_k), in the previous point of view the temporal differences are realizations of random functions (D_k). We call (D_k) functional temporal differences.

Remark 2.4 (Convolution and Stochastic Gradient)**.** Let p be the density of the random variable ξ w.r.t. the Lebesgue measure, and $K_k(x, y) = \frac{1}{p(x)} K(\frac{x-y}{k})$. Then the algorithm (2.3) can be re written as:

$$J_{k+1}(\cdot) = J_k(\cdot) + \gamma_k \varepsilon_k d_k(\xi_{k+1}, w_{k+1}) \frac{K((\xi_{k+1} - \cdot)/k)}{p(\cdot)\varepsilon_k}. \tag{2.5}$$

We can observe that our algorithm combines ideas concerning stochastic gradient and convolution approximations. In fact, the application of a classical Robbins-Monro algorithm for equation (2.1) gives us:

$$J_{k+1} = J_k + \gamma_k(g + \alpha\Pi(J_k) - J_k).$$

A possible problem arises when J_k is of infinite dimension. In such a case it is not possible to perform this algorithm. Hence to overcome this hurdle, we can approximate the previous equation using a mollifier sequence. Rearranging terms, we see that the last relation can be written as follows:

$$\begin{aligned} J_{k+1}(\cdot) =& J_k(\cdot) + \gamma_k \mathbb{E}\left[\left(g(\xi) + \alpha J_k(w) - J_k(\xi)\right) K_k((\xi, \cdot)\right] \\ =& J_k(\cdot) + \gamma_k \varepsilon_k \int \left(\underbrace{g(x) + \alpha J_k(y) - J_k(x)}_{\text{temporal difference sample}} \right) \underbrace{\frac{K((x-\cdot)/k)}{\varepsilon_k}}_{\text{mollifier}} \Pi(x, dy)dx \end{aligned} \tag{2.6}$$

In numerical analysis, the use of mollifier sequences is a useful method, provided that

$$\lim_{k\to\infty} \int f(x, y) \frac{K((x-\xi)/k)}{\varepsilon_k} \Pi(x, dy)dx = \int f(\xi, y)\Pi(\xi, dy),$$

for a sufficiently large class of mappings f, including the successive temporal differences. The final step consists of combining the convolution (or mollifier) ideas introduces in (2.6) with stochastic approximation. Indeed, using a Monte-Carlo method, we replace the integral by successive samples (ξ_{k+1}, w_{k+1}), hoping that the mappings J_k do not change too much along the iterations:

$$\int (g(x) + \alpha J_k(y) - J_k(x)) \frac{K((x-\xi)/k)}{\varepsilon_k} \Pi(x, dy)dx \sim \sum_{l\le k} \left(g(\xi_{l+1}) + \alpha J_l(w_{l+1}) - J_l(\xi_{l+1})\right) \frac{K((\xi_{l+1} - \xi)/l)}{p(\xi_{l+1})\varepsilon_l}$$

We are now going to give a proof of Algorithm 2.2. First of all, let us prove the following useful lemma :

Lemma 2.5. *Let $f \in L^2(S, \mathcal{B}, \pi)$:*

$$\langle f, \Pi(f) \rangle_\pi \leq \|f\|_\pi^2 .$$

Proof : The application of the Cauchy-Schwarz inequality and the Jensen inequality imply:

$$\begin{aligned} \langle f, \Pi(f) \rangle_\pi &= \int_S f(x)\Pi(f)(x)\pi(dx), \\ &\leq \left(\int_S f(x)^2 \pi(dx) \right)^{1/2} \left(\int_S \Pi(f)(x)^2 \pi(dx) \right)^{1/2}, \\ &\leq \|f\|_\pi \left(\int_{S\times S} f(y)^2 \Pi(x, dy)\pi(dx) \right)^{1/2}. \end{aligned}$$

Since π is an invariant distribution for the kernel Π,

$$\begin{aligned} \langle f, \Pi(f) \rangle_\pi &\leq \|f\|_\pi \left(\int_S f(y)^2 \pi(dy) \right)^{1/2}, \\ &\leq \|f\|_\pi^2 , \end{aligned}$$

which completes the proof. □

In order to simplify the formulas, we adopt the following notation:

$$\Pi(d_k)(x) = \int_S d_k(x, y)\Pi(x, dy).$$

Let us state the main result of this section:

Theorem 2.6. *Under the following assumptions:*

(i) $(\xi_k, w_k)_{k\in\mathbb{N}}$ *is an i.i.d. sample of the random variable* (ξ, w),

(ii) the functional temporal differences (D_k) *are such that there exists a nonnegative sequence* $(\varepsilon_k)_{k\in\mathbb{N}}$ *and* $b_1 \geq 0$ *such that for all* $k \in \mathbb{N}$,

$$\left\| \mathbb{E}^k \left[D_k(\xi_{k+1}, w_{k+1}) \right] - \Pi(d_k) \right\|_\pi \leq b_1 \varepsilon_k \left(1 + \|\Pi(d_k)\|_\pi \right), \tag{2.7a}$$

$$\int_S K_k(\xi_{k+1}, y)^2 \pi(dy) \leq \varepsilon_k, \tag{2.7b}$$

(iii) the sequences (γ_k) *and* (ε_k) *satisfy the following properties:*

$$\sum_{k\in\mathbb{N}} \gamma_k \varepsilon_k = \infty, \quad \sum_{k\in\mathbb{N}} \gamma_k^2 \varepsilon_k < \infty, \quad \sum_{k\in\mathbb{N}} b_1 \gamma_k \varepsilon_k^2 < \infty, \tag{2.8}$$

the sequence $(J_k)_{k\in\mathbb{N}}$ *generated by Algorithm 2.2 strongly converges to the unique optimal solution of (2.1).*

Proof : We shall first study the evolution of the sequence $(\|J_k - J^*\|_\pi)_{k\in\mathbb{N}}$. The conclusion will be obtained as a consequence of the Robbins-Siegmund Lemma, (see [Robbins and Siegmund, 1971]).

$$\begin{aligned} \|J_{k+1} - J^*\|_\pi^2 &= \|J_k - J^* + \gamma_k \left(g(\xi_{k+1}) + \alpha J_k(w_{k+1}) - J_k(\xi_{k+1}) \right) K_k(\xi_{k+1}, \cdot)\|_\pi^2 , \\ &= \|J_k - J^*\|_\pi^2 + 2\gamma_k \left\langle J_k - J^*, \left(g(\xi_{k+1}) + \alpha J_k(w_{k+1}) - J_k(\xi_{k+1}) \right) K_k(\xi_{k+1}, \cdot) \right\rangle_\pi , \\ &\quad + \gamma_k^2 \|\left(g(\xi_{k+1}) + \alpha J_k(w_{k+1}) - J_k(\xi_{k+1}) \right) K_k(\xi_{k+1}, \cdot)\|_\pi^2 . \end{aligned}$$

By considering the conditional expectation with respect to $\mathcal{F}_k$:

$$\begin{aligned} \mathbb{E}^k \left[\|J_{k+1} - J^*\|_\pi^2 \right] &= \|J_k - J^*\|_\pi^2 + 2\gamma_k \underbrace{\mathbb{E}^k \left[\left\langle J_k - J^*, \left(g(\xi_{k+1}) + \alpha J_k(w_{k+1}) - J_k(\xi_{k+1}) \right) K_k(\xi_{k+1}, \cdot) \right\rangle_\pi \right]}_{A} \\ &\quad + \gamma_k^2 \underbrace{\mathbb{E}^k \left[\|\left(g(\xi_{k+1}) + \alpha J_k(w_{k+1}) - J_k(\xi_{k+1}) \right) K_k(\xi_{k+1}, \cdot)\|_\pi^2 \right]}_{B} . \end{aligned} \tag{2.9}$$

We shall now provide upper bounds for $\frac{1}{\varepsilon_k}A$ and B as well:

$$\begin{aligned}
\frac{1}{\varepsilon_k}A &\leq \left\langle J_k - J^*, \mathbb{E}^k\left[(g(\xi_{k+1}) + \alpha J_k(w_{k+1}) - J_k(\xi_{k+1}))\,\frac{K_k(\xi_{k+1},\cdot)}{\varepsilon_k}\right]\right\rangle_\pi, \\
&\leq \left\langle J_k - J^*, \mathbb{E}^k\left[d_k(\xi_{k+1},w_{k+1})\frac{K_k(\xi_{k+1},\cdot)}{\varepsilon_k}\right] - (g + \alpha\Pi(J_k) - J_k)\right\rangle_\pi \\
&\quad + \langle J_k - J^*, T(J_k) - J_k\rangle_\pi, \\
&\leq \|J_k - J^*\|_\pi \left\|\mathbb{E}^k\left[D_k(\xi_{k+1},w_{k+1})\right] - \Pi(d_k)\right\|_\pi, \\
&\quad + \langle J_k - J^*, T(J_k) - J^*\rangle_\pi + \langle J_k - J^*, J^* - J_k\rangle_\pi.
\end{aligned}$$

Assumption (2.7a) implies:

$$\frac{1}{\varepsilon_k}A \leq b_1\varepsilon_k \|J_k - J^*\|_\pi \left(1 + \|\Pi(d_k)\|_\pi\right) + \|J_k - J^*\|_\pi \|T(J_k) - J^*\|_\pi - \|J_k - J^*\|_\pi^2. \tag{2.10}$$

One can remark that:

$$\begin{aligned}
\|\Pi(d_k)\|_\pi &= \|T(J_k) - J_k\|_\pi, \\
&\leq \|T(J_k) - J^*\|_\pi + \|J^* - J_k\|_\pi, \\
&\leq (1+\alpha)\|J_k - J^*\|_\pi. \qquad (2.11)
\end{aligned}$$

Equation (2.10) then becomes:

$$\frac{1}{\varepsilon_k}A \leq b_1\varepsilon_k \|J_k - J^*\|_\pi + (1+\alpha)b_1\varepsilon_k \|J_k - J^*\|_\pi^2 + (\alpha - 1)\|J_k - J^*\|_\pi^2.$$

By use of the inequality $x \leq 1 + x^2$ and the Lemma 2.5 one can have:

$$\frac{1}{\varepsilon_k}A \leq (b_1\varepsilon_k + (1+\alpha)b_1\varepsilon_k + \alpha - 1)\|J_k - J^*\|^2 + b_1\varepsilon_k.$$

The application of Cauchy-Schwarz inequality gives:

$$B \leq \mathbb{E}^k\left[|(g(\xi_{k+1}) + \alpha J_k(w_{k+1}) - J_k(\xi_{k+1}))|^2 \|K_k(\xi_{k+1},\cdot)\|_\pi^2\right].$$

Assumption (2.7b) yields:

$$\begin{aligned}
B \leq& \varepsilon_k\mathbb{E}^k\left[|(g(\xi_{k+1}) + \alpha J_k(w_{k+1}) - J_k(\xi_{k+1}))|^2\right], \\
\leq& \varepsilon_k\mathbb{E}^k\left[(\alpha(J_k(w_{k+1}) - J^*(w_{k+1})) - (J_k(\xi_{k+1}) - J^*(\xi_{k+1})) + g(\xi_{k+1}) + \alpha J^*(w_{k+1}) - J^*(\xi_{k+1}))^2\right].
\end{aligned}$$

Under Jensen's inequality $(x+y+z)^2 \leq 3(x^2+y^2+z^2)$ the previous relation becomes:

$$\begin{aligned}
B \leq 3\varepsilon_k \Big(&\alpha^2\mathbb{E}^k\left[(J_k(w_{k+1}) - J^*(w_{k+1}))^2\right] + \mathbb{E}^k\left[(J_k(\xi_{k+1}) - J^*(\xi_{k+1}))^2\right] \\
&+\mathbb{E}^k\left[(g(\xi_{k+1}) + \alpha J^*(w_{k+1}) - J^*(\xi_{k+1}))^2\right]\Big).
\end{aligned}$$

As a consequence of (2.4) it holds,

$$B \leq 3\varepsilon_k(\alpha^2+1)\|J_k - J^*\|_\pi^2 + 3\varepsilon_k\mathbb{E}^k\left[(g(\xi_{k+1}) + \alpha J^*(w_{k+1}) - J^*(\xi_{k+1}))^2\right]$$

We use again the Jensen inequality $(x+y+z)^2 \leq 3(x^2+y^2+z^2)$:

$$B \leq 3\varepsilon_k(\alpha^2+1)\|J_k - J^*\|_\pi^2 + 9\varepsilon_k\left(\mathbb{E}^k\left[g(\xi_{k+1})^2\right] + \alpha^2\mathbb{E}^k\left[J^*(w_{k+1})^2\right] + \mathbb{E}^k\left[J^*(\xi_{k+1})^2\right]\right).$$

Thanks to (2.4):

$$B \leq 3\varepsilon_k(\alpha^2+1)\|J_k - J^*\|_\pi^2 + 9\varepsilon_k \underbrace{\left(\|g\|_\pi^2 + \alpha^2\|J^*\|_\pi^2 + \|J^*\|_\pi^2\right)}_{\delta}.$$

Therefore the inequality (2.9) can be rewritten as:

$$\begin{aligned}
\mathbb{E}^k\left[\|J_{k+1} - J^*\|_\pi^2\right] \leq& \left[1 + 2\gamma_k\varepsilon_k(b_1\varepsilon_k + (1+\alpha)b_1\varepsilon_k + \alpha - 1 + \frac{3}{2}\gamma_k(\alpha^2+1))\right]\|J_k - J^*\|_\pi^2 \\
&+ 2b_1\gamma_k\varepsilon_k^2 + 9\gamma_k^2\varepsilon_k\delta.
\end{aligned}$$

Hence we can apply the Robbins-Siegmund's Lemma [Robbins and Siegmund, 1971]:

$$\|J_k - J^*\|_\pi^2 \text{ converges as when } k \to \infty \text{ and,}$$

$$\sum_{\in\mathbb{N}} \gamma_k \varepsilon_k \left\| J_k - J^* \right\|_\pi^2 < \infty.$$

The previous relations prove that $(\|J_k - J^*\|_\pi)_{k\in\mathbb{N}}$ converges to 0. □

Remark 2.7. We shall stress here the importance of the following two remarks:

- The idea of the assumption (2.7a) is that the functional temporal difference constitutes in expectation an approximation of the conditional expectation of the classical temporal difference. It is hence a convolution assumption.
- Assumption (2.8) is useful since it provides the joint stepsize decrease speed. Furthermore it is worth noting the symetry of these relations since it implies that the sequences $(\gamma_k)_{k\in\mathbb{N}}$ and $(\varepsilon_k)_{k\in\mathbb{N}}$ may exchange their decrease speed.

3. Perturbed Gradient analysis

We are going to provide another convergence proof using recent results about perturbed gradient methods with biased estimators (see [Barty et al., 2005]). This other setting will lead to a more general result than Theorem 2.6. We will use the same notations as before, and add a few ones. First of all, consider $\mathcal{U}$ to be a finite dimensional Hilbert space, endowed with the inner product $\langle\cdot,\cdot\rangle_{\mathcal{U}}$. We consider now the bilinear real valued application denoted by $\langle\cdot,\cdot\rangle_{\pi,\mathcal{U}}$, and defined by :

$$\forall u, v : S \to \mathcal{U},\ \langle u, v\rangle_{\pi,\mathcal{U}} = \int_S \langle u(x), v(x)\rangle_{\mathcal{U}}\, \pi(dx).$$

It will turn out that this application is a scalar product, and we will denote the associated norm by $\|\cdot\|_{\pi,\mathcal{U}}$. We also define H to be

$$H : L^2_{\mathcal{U}}(S,\mathcal{B},\pi) \to L^2_{\mathcal{U}}(S,\mathcal{B},\pi) \quad H(u)(x) = \int_S h(x,u(y))\Pi(x,dy),$$

where $L^2_{\mathcal{U}}(S,\mathcal{B},\pi)$ denote the set of all the $\mathcal{B}$-measurable mappings $u : S \to \mathcal{U}$ such that $\|u\|_{\pi,\mathcal{U}} < \infty$. It is of course an Hilbert space, endowed with the inner product $\langle\cdot,\cdot\rangle_{\pi,\mathcal{U}}$.
The aim is to approximate numerically the solution of the following fixed point equation:

$$u = H(u). \tag{3.1}$$

We propose the following algorithm:

Algorithm 3.1. Step -1 : initialize $u_0(\cdot)$,
Step $k \geq 0$:

- Draw ξ_{k+1} independently from the past draws with respect to a distribution π and then draw w_{k+1} with respect to the distribution $\Pi(\xi_{k+1},\cdot)$;
- Update:

$$\begin{aligned} s_k &= H(u_k) - u_k, \\ \Delta_k &= h(\xi_{k+1}, u_k(w_{k+1})) - u_k(\xi_{k+1}), \\ z_k &= \Delta_k \frac{K_k(\xi_{k+1},\cdot)}{\varepsilon_k} - (H(u_k) - u_k), \\ u_{k+1} &= u_k + \gamma_k\varepsilon_k(s_k + z_k). \end{aligned} \tag{3.2}$$

We have already presented an original algorithm in various points:

(1) We are working directly in the infinite dimension space to which the solution belongs. In spite of the infinite dimension, this method remains numerically tractable since in order to compute u_{k+1} one only needs to keep in memory $\{u_k, \Delta_k, \xi_{k+1}\}$. Using the previous notation of Δ_k it holds that:

$$u_{k+1}(\cdot) = \sum_{i=0}^{k} \gamma_i \Delta_i K_i(\xi_{i+1},\cdot) + u_0(\cdot).$$

Since $\Delta_i \in \mathcal{U}$ and $\xi_i \in S$ we need $(k+1)(\dim \mathcal{U}+\dim S)$ scalar values to compute completely the function u_{k+1}. One can also observe that in the worst case, the computational time to perform u_k grows linearly with k, but in most cases, the expensive part of the computation will be the evaluation of Δ_k.

(2) A second worthwhile point is that we are solving the original problem, without any a priori knowledge on the solution.

Theorem 3.2. *If the following assumptions are verified*

(i) $(\xi_k, w_k)_{k\in\mathbb{N}}$ *is an i.i.d. sample of the random variable* (ξ, w),

(ii) the mapping H *is a contraction mapping with* $\|\cdot\|_{\pi,\mathcal{U}}$*:*

$$\exists \beta \in [0,1[,\ \forall u, \bar{u} \in L^2_{\mathcal{U}}(S,\mathcal{B},\pi), \quad \|H(u)-H(\bar{u})\|_{\pi,\mathcal{U}} \leq \beta \|u-\bar{u}\|_{\pi,\mathcal{U}}\,. \tag{3.3}$$

(iii) it holds for the sequence defined by (3.2):

$$\exists b \geq 0, \quad \forall k \in \mathbb{N}, \quad \|\mathbb{E}\left[z_k \mid \mathcal{F}_k\right]\|_{\pi,\mathcal{U}} \leq b\varepsilon_k(1+\|H(u_k)-u_k\|_{\pi,\mathcal{U}}), \tag{3.4a}$$

$$\exists A \geq 0, \quad \forall k \in \mathbb{N}, \quad \mathbb{E}\left[\|z_k\|^2_{\pi,\mathcal{U}} \mid \mathcal{F}_k\right] \leq A(1+\frac{1}{\varepsilon_k}\|H(u_k)-u_k\|^2_{\pi,\mathcal{U}}), \tag{3.4b}$$

(iv) the sequences (γ_k) *and* (ε_k) *are such that:*

$$\sum_{k\in\mathbb{N}} \gamma_k\varepsilon_k = \infty, \quad \sum_{k\in\mathbb{N}} \gamma_k^2\varepsilon_k < \infty, \quad \sum_{k\in\mathbb{N}} b\gamma_k\varepsilon_k^2 < \infty, \tag{3.5}$$

then there exist a unique $u^* \in L^2_{\mathcal{U}}(S,\mathcal{B},\pi)$*, such that* $H(u^*)=u^*$*, and the sequence* $(u_k)_{k\in\mathbb{N}}$ *strongly converges to* u^**.*

Proof : The proof will be obtained by means of [Barty et al., 2005, Theorem 2.4].

Let us define a Lyapunov function $f : \mathcal{U} \to \mathbb{R}$ as follow:

$$\forall u \in L^2_{\mathcal{U}}(S,\mathcal{B},\pi), \quad f(u) = \frac{1}{2}\|u-u^*\|^2_{\pi,\mathcal{U}}\,.$$

The gradient of f denoted by ∇f is given by:

$$\forall u \in L^2_{\mathcal{U}}(S,\mathcal{B},\pi), \quad \nabla f(u) = u-u^*.$$

Clearly f is a strongly convex function and its Gâteaux derivative ∇f is Lipschitz continuous so the first and the third assumptions of [Barty et al., 2005, Theorem 2.4] are fulfilled as well.

Moreover it holds true that:

$$\begin{aligned}
\langle s_k, u_k-u^*\rangle_{\pi,\mathcal{U}} &= \langle H(u_k)-u_k, u_k-u^*\rangle_{\pi,\mathcal{U}}\,,\\
&= \langle H(u_k)-u^*, u_k-u^*\rangle_{\pi,\mathcal{U}} + \langle u^*-u_k, u_k-u^*\rangle_{\pi,\mathcal{U}}\,,\\
&\leq \|H(u_k)-u^*\|_{\pi,\mathcal{U}}\|u_k-u^*\|_{\pi,\mathcal{U}} - \|u_k-u^*\|^2_{\pi,\mathcal{U}}\,,\\
&\leq (\beta-1)f(u_k),\\
&\leq (1-\beta)(f(u^*)-f(u_k)).
\end{aligned}$$

Therefore s_k is a descent direction for the Lyapunov function f.

Furthermore:

$$\begin{aligned}
\|s_k\|_{\pi,\mathcal{U}} &= \|H(u_k)-u_k\|_{\pi,\mathcal{U}}\,,\\
&\leq \|H(u_k)-u^*\|_{\pi,\mathcal{U}} + \|u^*-u_k\|_{\pi,\mathcal{U}}\,,\\
&\leq (1+\beta)\|u_k-u^*\|_{\pi,\mathcal{U}}\,,\\
&\leq (1+\beta)(1+\|\nabla f(u_k)\|_{\pi,\mathcal{U}}).
\end{aligned}$$

We have already satisfied the fourth assumption of [Barty et al., 2005, Theorem 2.4]. Since all assumptions of [Barty et al., 2005, Theorem 2.4] are satisfied we deduce that (u_k) strongly converge to u^*. □

Remark 3.3 (Variance assumption)**.** Clearly one can easily see the main advantage of assumption (3.4b). The key point here is that a priori it is much easier to bound the variance of z_k by a non-constant amount.

Remark 3.4 (Contraction of H and invariant distribution)**.** Theorem 3.2 shows that it is possible to obtain the convergence result as soon as the operator H is a contraction with respect to the underlying L^2 norm. The fact that H is a contraction mapping is often linked with the invariance property of the probability measure π, when the underlying problem is a stochastic dynamic programming problem. Very often, the invariant probability of a Markov chain is not easy to compute.
We can notice that if we have a probability measure on the same space, denoted by π', such that the associated Hilbert spaces $L^2_{\mathcal{U}}(S, \mathcal{B}, \pi)$ and $L^2_{\mathcal{U}}(S, \mathcal{B}, \pi')$ coincides, and such that they are equivalent, with essentially bounded Radon-Nykodym derivatives, then the two norms $\|\cdot\|_{\pi,\mathcal{U}}$ and $\|\cdot\|_{\pi',\mathcal{U}}$ are topologically equivalent. Hence, a mapping which is a contraction mapping with parameter β with the norm $\|\cdot\|_{\pi,\mathcal{U}}$ is Lipschitz continuous for the norm $\|\cdot\|_{\pi',\mathcal{U}}$, with Lipschitz constant β' given by:

$$\beta' = \beta \sqrt{\left\| \frac{d\pi}{d\pi'} \right\|_\infty \left\| \frac{d\pi'}{d\pi} \right\|_\infty}.$$

Therefore, a condition on the Radon-Nykodym derivatives may ensure that a mapping remains a contraction mapping under norms induced by different equivalent probability measures.
Practically, it means that it is possible to use another probability measure as soon as it is not far (in the sense of the Radon-Nykodym derivatives) from the invariant one for which the mapping is a contraction.

Remark 3.5 (Convergence of the TD(0) Algorithm)**.** The convergence of Algorithm 2.2 can be obtained by the use of Theorem 3.2 and an appropriate mapping H. The Algorithm 2.2 is obtained as an application of Algorithm 3.1 with $\mathcal{U} = \mathbb{R}$ and the mapping h defined by:

$$h(x, J) = g(x) + \alpha J, \ \forall x \in S, \ J \in \mathbb{R},$$

and $H(J)(x) = \int_S (g(y) + \alpha J(y))\, \Pi(x, dy)$. That is:

$$J_{k+1}(\cdot) = J_k(\cdot) + \gamma_k \underbrace{(g(\xi_{k+1}) + \alpha J_k(w_{k+1}) - J_k(w_{k+1}))}_{d_k(\xi_{k+1}, w_{k+1})} K_k(\xi_{k+1}, \cdot).$$

If $d_k(\xi_{k+1}, w_{k+1})$ denotes the classical temporal difference then an implementation of Algorithm 3.1 is given by:

$$J_{k+1}(\cdot) = J_k(\cdot) + \gamma_k d_k(\xi_{k+1}, w_{k+1}) K_k(\xi_{k+1}, \cdot).$$

Hence, the functional temporal difference learning algorithm is a particular case of the general stochastic approximation Algorithm 3.1, and if one seeks to verify the assumptions of Theorem 3.2, one will get it under the assumptions of Theorem 2.6.

4. Applications

To amplify and to enhance our understanding, let us present in more details two applications of the previously defined Algorithms 2.2 and 3.1. The first one provides the computation of the Bellman function of a not-controled infinite horizon problem. The second one adresses the pricing of a Bermudan put option.

4.1. **Infinite Horizon problem.** Let α be a discount factor and $(X_t)_{t\in\mathbb{N}}$ be an autoregressive process in $\mathbb{R}$:

$$\forall t \in \mathbb{N}, \ X_{t+1} = \gamma X_t + \eta_t,$$

with (η_t) i.i.d. with distribution $\mathcal{N}\left(0, \sigma^2\right)$ and γ the autocorrelation factor.

We are interested in computing

$$J^*(x) = \mathbb{E}\left[\sum_{t\geq 0} \alpha^t X_t^2 \mid X_0 = x \right].$$

This example is chosen so that the calculation can be carried out by hand. It yields:

$$J^*(x) = \frac{x^2 - \sigma^2 \frac{\alpha}{(\alpha-1)}}{1 - \alpha\gamma^2}.$$

For the numerical application, we implement the use of the temporal difference learning method TD(0) adapted to use kernels (Algorithm 2.2). We progressively draw a realization of the (X_t) process and incrementally update an estimation J_k of the expected income J^*, starting with $J_0(\cdot) = 0$. A straightforward application of Algorithm 2.2 yields:

$$J_k(\cdot) = J_{k-1}(\cdot) + \gamma_k \left(X_k^2 + \alpha J_{k-1}(X_{k+1}) - J_{k-1}(X_k) \right) K_k(X_k, \cdot)$$

With K_k a given Gaussian kernel of chosen variance ϵ_k^2, centered in X_k and γ_k an appropriately chosen stepsize.
We obtain the Figure 4.1 showing the evolution of the L^2 error between J_k and J^* along the iterations.

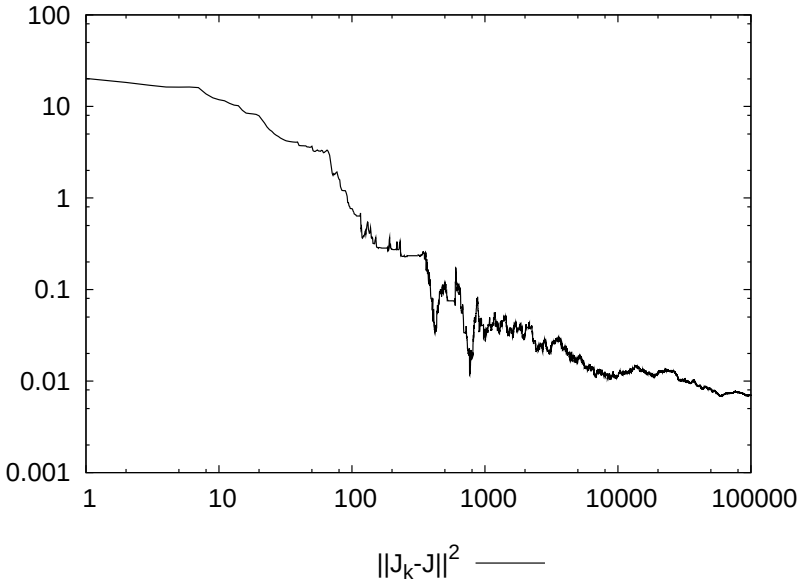


FIGURE 4.1. Convergence speed

Figure 4.2 shows the iterates J_k and the optimal solution J^* after 100, 1000, 10000 and 100000 iterations, and illustrates the convergence.

After this first academic example, we go to a more important example, namely the pricing of bermudan put options.

4.2. **Option pricing.** We apply our algorithm to the pricing of a Bermudan put option. A Bermudan put option is an option giving the right to sell the underlying stock at prescribed exercising dates, during a given period, at prescribed prices. It is hence a kind of intermediate between european and american options. In our case, the exercise dates are restricted to equispaced dates t in $0, \ldots, T$, and the stock price X_t follows a discretized risk-neutral Black-Scholes dynamics, given by:

$$\forall t \in \mathbb{N},\ \ln \frac{X_{t+1}}{X_t} = r - \frac{1}{2}\sigma^2 + \sigma \eta_t$$

where (η_t) is a Gaussian white noise of variance unity, and r is the risk-free interest rate. The strike price is assumed to be s, therefore the intrinsic value of the option when the price is x is $g(x) = \max(0, s - x)$. Let us define the discount factor $\alpha = e^{-r}$. Given the price x_0 at $t = 0$, our objective is to calculate the value of the option:

$$\max_{\tau} \mathbb{E}\left[\alpha^\tau g(X_\tau) \mid X_0 = x_0 \right],$$

where τ is taken among the stopping times with respect to the filtration generated by the discretized price process (X_t). In our case, $\tau \in \{0, \ldots, T\}$.

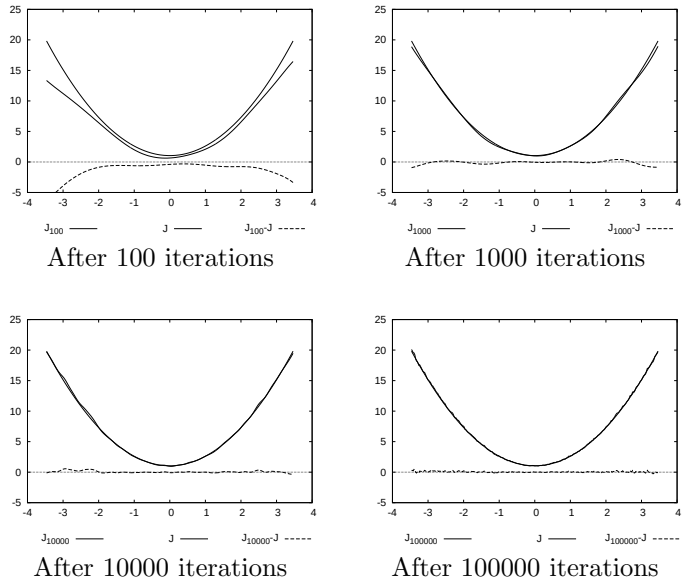


FIGURE 4.2. Estimation and error at 100, 1000, 10000, 100000 iterations

Among the multiple methods that have been proposed for option pricing, two share similarities with our approach. [Van Roy and Tsitsiklis, 2001] describes an approximate dynamic programming approach but neither presents numerical results nor suggests good choices for the basis. Our work directly extends the methodolgy presented by guaranteing asymptotic convergence and eliminating the need to choose a basis. [Longstaff and Schwartz, 2001] describes a regression approach to estimate the conditional expected payoff of the option. Our scheme can be very roughly seen as an incremental, non parametric implementation of this regression.

Let $J_t(x)$ be the value of the option at time t if the price X_t is equal to x. Since the option must be exercised before $T+1$, we have $J_{T+1}(x) = 0$. Therefore, for all $t \leq T$:

$$J_t(x) = \max\left(g(x), \alpha\mathbb{E}\left[J_{t+1}(X_{t+1}) \mid X_t = x\right]\right). \tag{4.1}$$

$(J_t(X_t))$ is often referred to as the Snell enveloppe of the stochastic process $(g(X_t))$.

In order to get a formula analogous to (3.1), we introduce the Q-functions (Q_t) defined by:

$$Q_t(x) = \alpha\mathbb{E}\left[J_{t+1}(X_{t+1}) \mid X_t = x\right]$$

i.e. the expected payoff at time t if we do not exercise the option. At each time t the value of the option is hence given by $J_t(x) = \max\left(g(x), Q_t(x)\right)$. Since $J_{T+1}(x) = 0$, we have $Q_T(x) = 0$. Equation (4.1) now reads:

$$Q_t(x) = \alpha\mathbb{E}\left[\max\left(g(X_{t+1}), Q_{t+1}(X_{t+1})\right) \mid X_t = x\right]$$

We perform the resolution using Algorithm 3.1, with the mapping $H : L^2_{\mathbb{R}^{T+1}}(\mathbb{R}^{T+1}, \mathcal{B}) \to L^2_{\mathbb{R}^{T+1}}(\mathbb{R}^{T+1}, \mathcal{B})$ defined by:

$$\forall t \in \{0, \ldots, T\},\ H(Q)_t(y) := \mathbb{E}\left[\alpha \max(g(X_t), Q_{t+1}(X_{t+1})) \mid X_t = y\right].$$

We are now able to implement Algorithm 3.1. For the numerical experiment, we take $\mu = 1$, $\sigma = 1$, $s = 1$, $x_0 = 1$ and $r = 0.01$ (and therefore $\alpha = 0.99$).

Lacking an analytic solution, our results (referred to as Q^k for the k^{th} iterate in the following graphs) are compared to a reference implementation of dynamic programming where the price process is finely discretized. we abusively denote this approximation of the optimal solution by Q^*. The graph of Q^* is provided in Figure 4.4.

Figure 4.3 shows the L^2 error along the iterations, while Figure 4.5 show the Q-functions $(Q_{t,k})$ along the iterations.

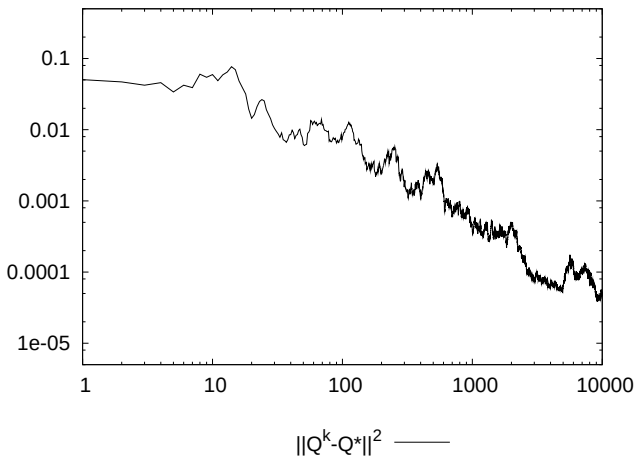


FIGURE 4.3. Convergence speed

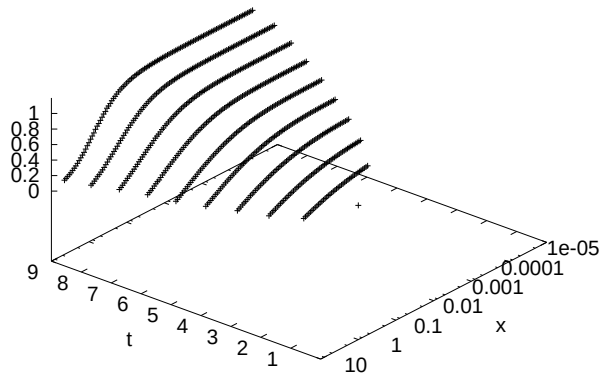


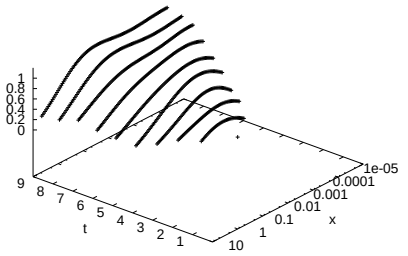
FIGURE 4.4. Optimum Q^* function

5. Conclusion

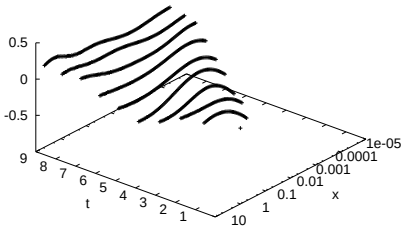
For Stochastic Dynamic Programming Problems, an usual and fruitfull way was up to now to use neural networks to avoid the drawbacks of any discretization of the underlying state space. Such approaches have but no guarantee of optimality.

In this paper, we present a new approach, based on nonparametric estimation and stochastic approximation techniques. Our approach generalizes e.g. the TD(0) Learning Algorithm in a continuous state space setting. Its main strength is to build iteratively a solution whose optimality is proven, by using only draws of the underlying stochastic processes, and without any a priori knowledge of the optimal solution. By using successive kernels, the iterations are performed directly in the infinite dimensional space, without any loss of optimality.

Two convergence proofs are given for the Algorithm. The first one is centered on the estimation of the cost-to-go function for an infinite horizon discounted Markov chain with continuous state space, whereas the second one allows to consider stopping time problems, i.e. finite horizon markovian control problems. The assumptions of the convergence theorems are classical in the

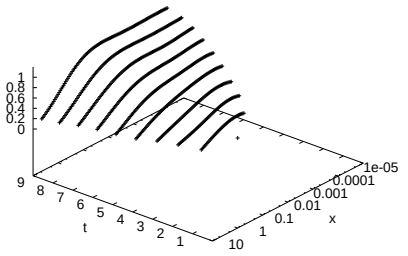


Estimation Q^{100}

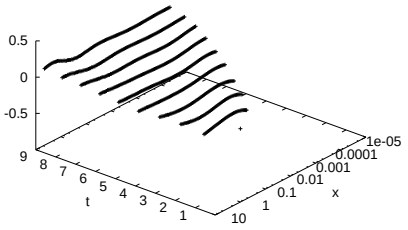


Error $Q^{100} - Q^*$

After 100 iterations

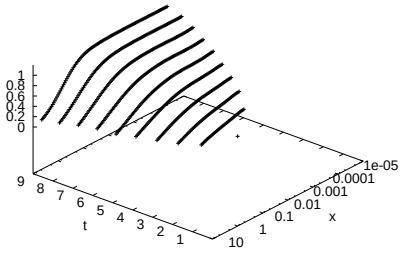


Estimation Q^{1000}

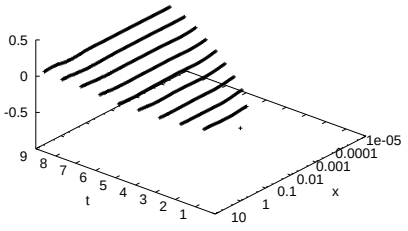


Error $Q^{1000} - Q^*$

After 1000 iterations



Estimation Q^{10000}



Error $Q^{10000} - Q^*$

After 10000 iterations

FIGURE 4.5. Estimation and error at 100, 1000, 10000 iterations.

framework of stochastic approximation, and allow a lot of applications.
In a straightforward application of our approach, the invariant distribution of the underlying Markov chain is not necessary to perform Algorithm 2.2. After a careful inspection of this paper the only requirement concerning the distribution, is that the Bellman operator must be a contraction mapping for the associated norm. Essentially, as shown in the Remark 3.4 such measures exist.

As an illustration, we show how our approach can be used for the pricing of Bermudan put options in the Black-Scholes framework.

A forthcoming work focuses on the extension of this approach to general Q-Learning algorithms. This would enable us to solve general optimal control problems with possibly high dimensional state space.

References

[Barty et al., 2005] Barty, K., Roy, J.-S., and Strugarek, C. (2005). A perturbed gradient algorithm in Hilbert spaces. *Optimization Online*. `http://www.optimization-online.org/DB_HTML/2005/03/1095.html`.

[Bellman and Dreyfus, 1959] Bellman, R. and Dreyfus, S. (1959). Functional approximations and dynamic programming. *Math tables and other aides to computation*, 13:247–251.

[Bertsekas and Tsitsiklis, 1996] Bertsekas, D. and Tsitsiklis, J. (1996). *Neuro-Dynamic Programming*. Athena Scientific.

[Chen and White, 1998] Chen, X. and White, H. (1998). Nonparametric learning with feedback. *Journal of Economic Theory*, 82:190–222.

[de Farias and Van Roy, 2004] de Farias, D. and Van Roy, B. (2004). A linear program for bellman error minimization with performance guarantees. submitted to Mathematics of Operations Research.

[Longstaff and Schwartz, 2001] Longstaff, F. A. and Schwartz, E. S. (2001). Valuing american options by simulation: A simple least squares approach. *Rev. Financial Studies*, 14(1):113–147.

[Robbins and Monro, 1951] Robbins, H. and Monro, S. (1951). A stochastic approximation method. *Annals of Mathematical Statistics*, 22:400–407.

[Robbins and Siegmund, 1971] Robbins, H. and Siegmund, D. (1971). A convergence theorem for nonnegative almost supermartingales and some applications. In Rustagi, J., editor, *Optimizing Methods in Statistics*, pages 233–257. Academic Press, New York.

[Sutton, 1988] Sutton, R. (1988). *Learning to predict by the method of temporal difference*, volume 37. IEEE Transaction on Automatic Control.

[Sutton and Barto, 1998] Sutton, R. and Barto, A. (1998). *Reinforcement Learning, an Introduction*. MIT press Cambridge.

[Tsitsiklis and Van Roy, 1997] Tsitsiklis, J. and Van Roy, B. (1997). An analysis of temporal-difference learning with function approximation. *IEEE Transaction on automatic control*, 42(5):674–690.

[Van Roy and Tsitsiklis, 2001] Van Roy, B. and Tsitsiklis, J. (2001). Regression methods for pricing complex american-style options. *IEEE Trans. on Neural Networks*, 12(4):694–703.

[Wolverton and Wagner, 1969] Wolverton, C. and Wagner, T. (1969). Recursive estimates of probability densities. *IEEE Transactions on Systems, Science and Cybernetics*, 5:307.