Machine Learning, 40, 265–299, 2000

A Study of Reinforcement Learning in the Continuous Case by the Means of Viscosity Solutions

RÉMI MUNOS* munos@cs.cmu.edu
Carnegie Mellon University, Robotics Institute, Pittsburgh, PA 15213, USA

Editor: Sridhar Mahadevan

Abstract. This paper proposes a study of *Reinforcement Learning* (RL) for continuous state-space and time control problems, based on the theoretical framework of *viscosity solutions* (VSs). We use the method of *dynamic programming* (DP) which introduces the *value function* (VF), expectation of the best future cumulative reinforcement. In the continuous case, the value function satisfies a non-linear first (or second) order (depending on the deterministic or stochastic aspect of the process) differential equation called the *Hamilton-Jacobi-Bellman* (HJB) equation. It is well known that there exists an infinity of generalized solutions (differentiable almost everywhere) to this equation, other than the VF. We show that gradient-descent methods may converge to one of these generalized solutions, thus failing to find the optimal control.

In order to solve the HJB equation, we use the powerful framework of viscosity solutions and state that there exists a unique viscosity solution to the HJB equation, which is the value function. Then, we use another main result of VSs (their stability when passing to the limit) to prove the convergence of numerical approximations schemes based on finite difference (FD) and finite element (FE) methods. These methods discretize, at some resolution, the HJB equation into a DP equation of a *Markov Decision Process* (MDP), which can be solved by DP methods (thanks to a "strong" contraction property) if all the initial data (the state dynamics and the reinforcement function) were perfectly known. However, in the RL approach, as we consider a system in interaction with some a priori (at least partially) unknown environment, which learns "from experience", the initial data are not perfectly known but have to be approximated during learning. The main contribution of this work is to derive a general convergence theorem for RL algorithms when one uses only "approximations" (in a sense of satisfying some "weak" contraction property) of the initial data. This result can be used for model-based or model-free RL algorithms, with off-line or on-line updating methods, for deterministic or stochastic state dynamics (though this latter case is not described here), and based on FE or FD discretization methods. It is illustrated with several RL algorithms and one numerical simulation for the "Car on the Hill" problem.

Keywords: reinforcement learning, dynamic programming, optimal control, viscosity solutions, finite difference and finite element methods, Hamilton-Jacobi-Bellman equation

1. Introduction

This paper is about Reinforcement Learning (RL) in the continuous state-space and time case. RL techniques (see Kaelbling, Littman, and Moore (1996) for a survey) are adaptive methods for solving optimal control problems for which only a partial amount of initial data are available to the system that learns. In this paper, we focus on the Dynamic Programming

*http://www.cs.cmu.edu/∼munos/.

(DP) method which introduces a function, called the *value function* (VF) (or cost function), that estimates the best future cumulative reinforcement (or cost) as a function of initial states.

RL in the continuous case is a difficult problem for at least two reasons. Since we consider a **continuous state-space**, the first reason is that the value function has to be approximated, either by using *discretization* (with grids or triangulations) or *general approximation* (such as neural networks, polynomial functions, fuzzy sets, etc.) methods. RL algorithms for continuous state-space have been implemented with neural networks (see for example Barto, Sutton, and Anderson (1983), Barto (1990), Gullapalli (1992), Williams (1992), Lin (1993), Sutton and Whitehead (1993), Harmon, Baird, and Klopf (1996), and Bertsekas and Tsitsiklis (1996)), fuzzy sets (see Nowé, 1995; Glorennec & Jouffe, 1997), approximators based on state aggregation (see Singh, Jaakkola, & Jordan, 1994), clustering (see Mahadevan & Connell, 1992), sparse-coarse-coded functions (see Sutton, 1996) and variable resolution grids (see Moore, 1991; Moore & Atkeson, 1995). However, as it has been pointed out by several authors, the combination of DP methods with function approximators may produce unstable or divergent results even when applied to very simple problems (see Boyan & Moore, 1995; Baird, 1995; Gordon, 1995). Some results using clever algorithms (like *Residual algorithms* of Baird (1995)) or particular classes of approximation functions (like the *Averagers* of Gordon (1995)) can lead to the convergence to a local or global solution within the class of functions considered.

Anyway, it is difficult to define the class of functions (for a neural network, the suitable architecture) within which the optimal value function could be approximated, knowing that we have little prior knowledge of its smoothness properties.

The second reason is because we consider a **continuous-time** variable. Indeed, the value function derived from the DP equation (see Bellman, 1957), relates the value at some state as a function of the values at successor states. In the continuous-time limit, as the successor states get infinitely closer, the value at some point becomes a function of its differential, defining a non linear differential equation, called the *Hamilton-Jacobi-Bellman* (HJB) equation.

In the discrete-time case, the resolution of the Markov Decision Process (MDP) is equivalent to the resolution, on the whole state-space, of the DP equation; this property provides us with DP or RL algorithms that locally solve the DP equation and lead to the optimal solution. With continuous time, it is no longer the case since the HJB equation holds only if the value function is differentiable. And in general, the value function is not differentiable everywhere (even for smooth initial data), thus this equation cannot be solved in the usual sense, because this leads to either no solution (if we look for classical solutions, i.e. differentiable everywhere) or an infinity of solutions (if we look for generalized solutions, i.e. differentiable almost everywhere).

This fact, which will be illustrated with a very simple 1-dimensional example, explains why there could be many "bad" solutions to gradient-descent methods for RL. Indeed, such methods intend to minimize the integral of some Hamiltonian. But the generalized solutions of the HJB equation are global optima of this problem, so the gradient-descent methods may lead to approximate any (among an infinity of) generalized solutions giving little chance to reach the desired value function.

In order to deal with the problem of integrating the HJB equation, we use the formalism of Viscosity Solutions (VSs), introduced by Crandall and Lions (in Crandall and Lions (1983); see the user's guide (Crandall, Ishii, & Lions, 1992)) in order to define an adequate class (which appears as a weak formulation) of solutions to non-linear first (and second) order differential equation such as HJB equations.

The main properties of VSs are their existence, their uniqueness and the fact that the value function is a VS. Thus, for a large class of optimal control problems, there exists a unique VS to the HJB equation, which is the value function. Furthermore, VSs have remarkable stability properties when passing to the limit, from which we can derive proofs of convergence for discretization methods.

Our approach here consists in defining a class of convergent numerical schemes, among which are the finite element (FE) and finite difference (FD) approximation schemes introduced in Kushner (1990) and Kushner and Dupuis (1992) to discretize, at some resolution δ, the HJB equation into a DP equation for some discrete Markov Decision Process. We apply a result of convergence (from Barles & Souganidis, 1991) to prove the convergence of the value function V^δ of the discretized MDP to the value function V of the continuous problem as the discretization step δ tends to zero.

The DP equation of the discretized MDP could be solved by any DP method (because the DP equation satisfies a "strong" contraction property leading successive iterations to converge to the value function, the unique fixed point of this equation), but only if the data (the transition probabilities and the reinforcements) were perfectly known by the learner, which is not the case in the RL approach.

Thus, we propose a result of convergence for RL algorithms when we only use "approximations" of these data (in the sense that the approximated DP equation need to satisfy some "weak" contraction property). The convergence occurs as the number of iterations tends to infinity and the discretization step tends to zero. This result applies to model-based or model-free RL algorithms, for off-line or on-line methods, for deterministic or stochastic state dynamics, and for FE or FD based discretization methods. It is illustrated with several RL algorithms and one numerical simulation for the "Car on the Hill" problem.

In what follows, we consider the *discounted, infinite-time horizon case* (for a description of the finite-time horizon case, see Munos (1997a)) with *deterministic state dynamics* (for the stochastic case, see Munos and Bourgine (1997) or Munos (1997a)).

Section 2 introduces the formalism for RL in the continuous case, defines the value function, states the HJB equation and presents a result showing continuity of the VF.

Section 3 illustrates the problems of classical solutions to the HJB equation with a simple 1-dimensional example and introduces the notion of viscosity solutions.

Section 4 is concerned with numerical approximation of the value function using discretization schemes. The finite element and finite difference methods are presented and a general convergence theorem (whose proof is in Appendix A) is stated.

Section 5 states a convergence theorem (whose proof is in Appendix B) for a general class of RL algorithms and illustrates it with several algorithms.

Section 6 presents a simple numerical simulation for the "Car on the Hill" problem.

2. A formalism for reinforcement learning in the continuous case

The objective of reinforcement learning is to learn from experience how to influence the behavior of a dynamic system in order to maximize some payoff function called the *reinforcement* or *reward* function (or equivalently to minimize some cost function). This is a problem of optimal control in which the state dynamics and the reinforcement function are, a priori, at least partially unknown.

In this paper we are concerned with *deterministic problems* in which the dynamics of the system is governed by a controlled differential equation. For similar results in the stochastic case, see Munos and Bourgine (1997) and Munos (1997a), for a related work using multigrid methods, see Pareigis (1996).

The two possible approaches for optimal control are Pontryagin's *maximum principle* (for theoretical work, see Pontryagin et al. (1962) and more recently Fleming and Rishel (1975), for a study of Temporal Difference, see Doya (1996), and for an application to the control with neural networks, see Bersini and Gorrini (1997) and the Bellman's *Dynamic Programming* (DP) (introduced in Bellman (1957)) approach considered in this paper.

2.1. Deterministic optimal control for discounted infinite-time horizon problems

In what follows, we consider infinite-time horizon problems under the discounted framework. In that case, the state dynamics do not depend on the time. For a study of the finite time horizon case (for which there is a dependency in time), see Munos (1997a).

Let $x(t)$ be the *state of the system*, which belongs to the *state-space* $\bar{O}$, closure of an open subset $O \subset \mathbb{R}^d$. The evolution of the system depends on the current state $x(t)$ and *control* (or *action*) $u(t) \in U$, where U, closed subset, is the *control space*; it is defined by the controlled differential equation:

$$\frac{dx(t)}{dt} = f(x(t), u(t)) \tag{1}$$

where the control $u(t)$ is a bounded, Lebesgue measurable function with values in U. The function f is called the *state dynamics*. We assume that f is Lipschitzian with respect to the first variable: there exists some constant $L_f > 0$ such that:

$$\forall x, y \in \bar{O}, \quad |f(x,u) - f(y,u)| \leq L_f \|x - y\| \tag{2}$$

For initial state x_0 at time $t = 0$ the choice of a control $u(t)$ leads to a unique (because the state dynamics (1) is deterministic) *trajectory* $x(t)$ (see figure 1).

Definition 1. We define the discounted **reinforcement functional** J, which depends on initial data x_0 and control $u(t)$ for $0 \leq t \leq \tau$, with τ the exit time of $x(t)$ from $\bar{O}$ (with $\tau = \infty$ if the trajectory always stays inside $\bar{O}$):

$$J(x_0; u(t)) = \int_0^\tau \gamma^t \cdot r(x(t), u(t))\, dt + \gamma^t \cdot R(x(\tau)) \tag{3}$$

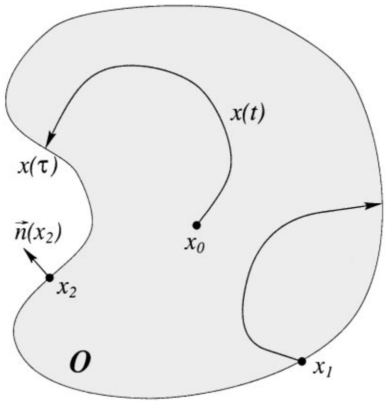

Figure 1. The state-space $\bar{O}$. From initial state x_0 at $t = 0$, the choice of control $u(t)$ leads to the trajectory $x(t)$ for $0 \le t \le \tau$, where τ is the exit time from the state-space.

with $r(x, u)$ the *current reinforcement* (defined on $\bar{O}$) and $R(x)$ the *boundary reinforcement* (defined on ∂O, the boundary of the state-space). $\gamma \in [0, 1)$ is the *discount factor* which weights short-term rewards more than long-term ones (and ensures the convergence of the integral).

The **objective of the control problem** is to find, for any initial state x_0, the control $u^*(t)$ that optimizes the reinforcement functional $J(x_0; u(t))$.

Remark. Unlike the discrete case, in the continuous case, we need to consider two different reinforcement functions: r is obtained and accumulated during the running of the trajectory, whereas R occurs whenever the trajectory exits from the state-space (if it does). This formalism enables us to consider many optimal control problem, such as reaching a target while avoiding obstacles, viability problems, and many other optimization problems.

Definition 2. We define the **value function**, the maximum value of the reinforcement functional as a function of initial state at time $t = 0$:

$$V(x) = \sup_{u(t)} J(x; u(t)) \tag{4}$$

Before giving some properties of the value function (HJB equation, continuity and differentiability properties), let us first describe the reinforcement learning framework considered here and the constraints it implies.

2.2. *The reinforcement learning approach*

RL techniques are adaptive methods for solving optimal control problems whose data are a priori (at least partially) unknown. Learning occurs iteratively, based on the experience of the interactions between the system and the environment, through the (current and boundary) reinforcement signals.

The objective of RL is to find the optimal control, and the techniques used are those of DP. However, in the RL approach, the state dynamics $f(x, u)$, and the reinforcement functions $r(x, u)$, $R(x)$ are partially unknown to the system. Thus RL is a constructive and iterative process that estimates the value function by successive approximations.

The learning process includes both a mechanism for the choice of the control, which has to deal with the *exploration* versus *exploitation* dilemma (exploration provides the system with new information about the unknown data, whereas exploitation consists in optimizing the estimated values based on the current knowledge) (see Meuleau, 1996), and a mechanism for integrating new information for refining the approximation of the value function. The latter topic is the object of this paper.

The study and the numerical approximations of the value function is of great importance in RL and DP because from this function we can deduce an optimal feed-back controller. The next section shows that the value function satisfies a local property, called the Hamilton-Jacobi-Bellman equation, and points out its relation to the optimal control.

2.3. The Hamilton-Jacobi-Bellman equation

Using the dynamic programming principle (introduced in Bellman (1957)), we can prove that the value function satisfies a local condition, called the Hamilton-Jacobi-Bellman (HJB) equation (see Fleming and Soner (1993) for a complete survey). In the deterministic case studied here, it is a first-order non-linear partial differential equation (in the stochastic case, we can prove that a similar equation of order two holds). Here we assume that U is a compact set.

Theorem 1 (*Hamilton-Jacobi-Bellman***).** *If the value function V is differentiable at x, let $DV(x)$ be the gradient of V at x, then the Hamilton-Jacobi-Bellman equation*:

$$V(x)\ln\gamma + \sup_{u\in U}[DV(x)\cdot f(x,u) + r(x,u)] = 0 \tag{5}$$

holds at $x \in O$. Additionally, V satisfies the following boundary condition:

$$V(x) \geq R(x) \quad \textit{for } x \in \partial O \tag{6}$$

Remark. The boundary condition is an inequality because at some boundary point (for example at $x_1 \in \partial O$ on figure 1) there may exist a control $u(t)$ such that the corresponding trajectory stays inside $\bar{O}$ and whose reinforcement functional is strictly superior to the immediate boundary reinforcement $R(x_1)$. In such cases, (6) holds with a strict inequality.

Remark. Using an equivalent definition, the HJB Eq. (5) means that V is the solution of the equation:

$$H(x, W, DW) = 0 \tag{7}$$

with the *Hamiltonian* H defined, for any differentiable function W, by:

$$H(x, W, DW) = -W(x) \ln \gamma - \sup_{u \in U} [DW(x) \cdot f(x, u) + r(x, u)].$$

Dynamic programming computes the value function in order to find the optimal control with a feed-back control policy, that is a function $\pi(x) : \bar{O} \to U$ such that the optimal control $u^*(t)$ at time t depends on current state $x(t) : u^*(t) = \pi(x(t))$. Indeed, from the value function, we deduce the following optimal feed-back control policy:

$$\pi^*(x) \in \arg \sup_{u \in U} [DV(x) \cdot f(x, u) + r(x, u)] \tag{8}$$

Now that we have pointed out the importance of computing the value function V for defining the optimal control, we show some properties of V (continuity, differentiability) and how to integrate (and in what sense) the HJB equation for approximating V.

2.4. *Continuity of the value function*

The property of continuity of the value function may be obtained under the following assumption concerning the state dynamics f around the boundary ∂O (which is assumed smooth, i.e. $\partial O \in \mathcal{C}^2$):

For all $x \in \partial O$, let $\vec{n}(x)$ be the outward normal of O at x (for example, see $\vec{n}(x_2)$ in figure 1), we assume that:

$$\begin{aligned}&\text{If } \exists u \in U \text{ with } f(x, u) \cdot \vec{n}(x) \leq 0, \text{ then } \exists v \in U \text{ with } f(x, v) \cdot \vec{n}(x) < 0\\ &\text{If } \exists u \in U \text{ with } f(x, u) \cdot \vec{n}(x) \geq 0, \text{ then } \exists v \in U \text{ with } f(x, v) \cdot \vec{n}(x) > 0\end{aligned} \tag{9}$$

These hypotheses mean that at any point of the boundary, there ought not be only trajectories tangential to the boundary of the state space.

Theorem 2 (*Continuity***).** *Suppose that* (2) *and* (9) *are satisfied, then the value function is continuous in* $\bar{O}$.

The proof of this theorem can be found in Barles and Perthame (1990).

3. Introduction to viscosity solutions

From Theorem 1, we know that if the value function is differentiable then it solves the HJB equation. However, in general, the value function is not differentiable everywhere even when the data of the problem are smooth. Thus, we cannot expect to find classical solutions (i.e. differentiable everywhere) to the HJB equation. Now, if we look for generalized solutions (i.e. differentiable *almost* everywhere), we find that there are many solutions other than the value function that solve the HJB equation.

Therefore, we need to define a weak class of solutions to this equation. Crandall and Lions introduced such a weak formulation by defining the notion of *Viscosity Solutions* (VSs) in Crandall and Lions (1983). For a complete survey, see Crandall et al. (1992), Barles (1994) or Fleming and Soner (1993). This notion has been developed for a very broad class of non-linear first and second order differential equations (including HJB equations for the stochastic case of controlled diffusion processes). Among the important properties of viscosity solutions are some uniqueness results, the stability of solutions to approximated equations when passing to the limit (this very important result will be used to prove the convergence of the approximation schemes in Section 4.4) and mainly the fact that the value function is the unique viscosity solution of the HJB Eq. (5) with the boundary condition (6).

First, let us illustrate with a simple example the problems raised here when one looks for classical or generalized solutions to the HJB equation.

3.1. 3 problems illustrated with a simple example

Let us study a very simple control problem in 1 dimension. Let the state $x(t) \in [0, 1]$, the control $u(t) \in \{-1, +1\}$ and the state dynamics be: $\frac{dx}{dt} = u$.

Consider a current reinforcement $r = 0$ everywhere and a boundary reinforcement defined by $R(0)$ and $R(1)$. In this example, we deduce that the value function is:

$$V(x) = \max\left\{R(0) \cdot \gamma^{x}, R(1) \cdot \gamma^{1-x}\right\} \tag{10}$$

and the HJB equation is:

$$V(x) \ln \gamma + \max\{V'(x), -V'(x)\} = 0 \tag{11}$$

with the boundary conditions $V(0) \geq R(0)$ and $V(1) \geq R(1)$.

1. **First problem:** there is no classical solution to the HJB equation. Let $R(0) = 1$, $R(1) = 2$, and $\gamma = 0.3$. The corresponding value function is plotted in figure 2. We observe that V is not differentiable everywhere, thus does not satisfy the HJB equation everywhere: there is no classical solution to the HJB equation.
2. **Second problem:** there are several generalized solutions. If one looks for generalized solutions that satisfy the HJB equation almost everywhere, we find many functions other than the value function. An example of a function satisfying (11) everywhere with the boundary conditions $R(0) = 1$ and $R(1) = 2$ is illustrated in figure 3.

   Remark. This problem is of great importance when one wants to use gradient-descent methods with some general function approximator, like neural networks, to approximate the value function. The use of gradient-descent methods may lead to approximate any of the generalized solutions of the HJB equation and thus fail to find the value

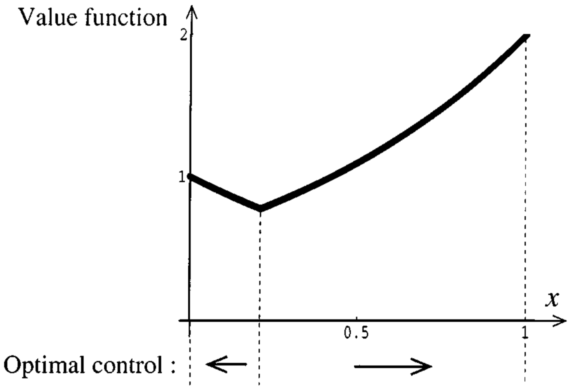

Figure 2. The value function is not differentiable everywhere.

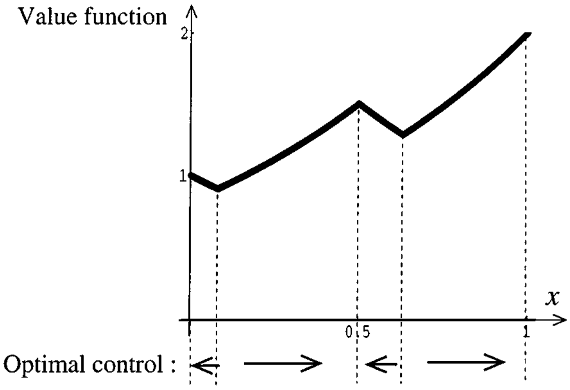

Figure 3. There are many generalized solutions other than the value function.

function. Indeed, suppose that we use a gradient-descent method for finding a function W minimizing the error:

$$E(W) = \int_{x \in O} H(x, W, DW)^2 \, dx \tag{12}$$

with H the Hamiltonian defined in Section 2.3. Then, the method will converge, in the best case, to any *generalized solution* V_g of (7) (because these functions are *global optima* of this minimization problem, since their error $E(V_g = 0)$ which are probably different from the value function V. Moreover the control induced by such functions (by the closed loop policy (8)) might be very different from the optimal control (defined by V). For example, the function plotted in figure 3 generates a control (given by the direction of the gradient) very different from the optimal control, defined by the value function plotted in figure 2.

In fact, in such continuous time and space problems, there exists an infinity of *global minima* for gradient descent methods, and these functions may be very different from the expected value function.

In the case of neural networks, we usually use smooth functions (differentiable everywhere), thus neither the value function V (figure 2), nor a generalized solution V_{g} (figure 3) can be exactly represented, but both can be approximated. Let us denote $\tilde{V}$ and $\tilde{V}_{\mathrm{g}}$, the best approximations of V and V_{g} in the network. Then $\tilde{V}$ and $\tilde{V}_{\mathrm{g}}$ are *local* minima of the gradient-descent method minimizing E, but nothing proves that $\tilde{V}$ is a *global* minimum. In this example, it could seem that V is "smoother" than the generalized solutions (because it has only one discontinuity instead of several ones) in the sense that $E(\tilde{V}) \leq E(\tilde{V}_{\mathrm{g}})$, but this is not true in general. In any case, in the continuous-time case, when we use a smooth function approximator, there exists an infinity of *local* solutions for the problem of minimizing the error E and nothing proves that the expected $\tilde{V}$ is a *global* solution. See Munos, Baird, and Moore (1999) for some numerical experiments on simple (one and two dimensional) problems.

When time is discretized, this problem disappears, but we still have to be careful when passing to the limit. In this paper, we describe discretization methods that converge to the value function when passing to the limit of the continuous case.

3. **Third problem:** the boundary condition is an inequality. Here we illustrate the problem of the inequality of the boundary condition. Let $R(0) = 1$ and $R(1) = 5$. The corresponding value function is plotted in figure 4. We observe that $V(0)$ is strictly superior to the boundary reinforcement $R(0)$. This strict inequality occurs at any boundary point $x \in \partial O$ for which there exists a control $u(t)$ such that the trajectory goes immediately inside $\bar{O}$ and generates a better reinforcement functional than the boundary reinforcement $R(x)$ obtained by exiting from $\bar{O}$ at x.

   We will give (in definition 4 that follows) a weak (viscosity) formulation of the boundary condition (6).

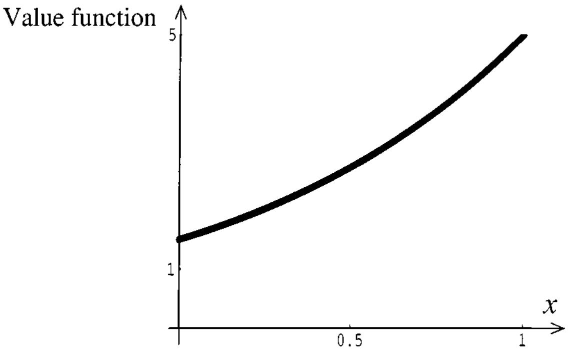

Figure 4. The boundary condition may hold with a strict inequality condition ($V(0) > R(0) = 1$).

3.2. *Definition of viscosity solutions*

In this section, we define the notion of viscosity solutions for continuous functions (a definition for discontinuous functions is given in Appendix A).

Definition 3 (Viscosity solution). Let W be a *continuous* real-valued function defined in O.

- W is a *viscosity sub-solution* of (7) in O if for all functions $\varphi \in C^1(O)$, for all $x \in O$ local maximum of $W - \varphi$ such that $W(x) = \varphi(x)$, we have:

$$H(x, \varphi(x), D\varphi(x)) \leq 0$$

- W is a *viscosity super-solution* of (7) in O if for all functions $\varphi \in C^1(O)$, for all $x \in O$ local minimum of $W - \varphi$ such that $W(x) = \varphi(x)$, we have:

$$H(x, \varphi(x), D\varphi(x)) \geq 0$$

- W is a *viscosity solution* of (7) in O if it is a viscosity sub-solution and a viscosity super-solution of (7) in O.

3.3. *Some properties of viscosity solutions*

The following theorem (whose proof can be found in Crandall et al. (1992)) states that the value function is a viscosity solution.

Theorem 3. *Suppose that the hypotheses of Theorem* 2 *hold. Then the value function V is a viscosity solution of* (7) *in* O.

In order to deal with the inequality of the boundary condition (6), we define a viscosity formulation in a differential type condition instead of a pure Dirichlet condition.

Definition 4 (Viscosity boundary condition). Let W be a continuous real-valued function defined in $\bar{O}$,

- W is a *viscosity sub-solution* of (7) in O *with the boundary condition* (6) if it is a viscosity sub-solution of (7) in O and for all functions $\varphi \in C^1(\bar{O})$, for all $x \in \partial O$ local maximum of $W - \varphi$ such that $W(x) = \varphi(x)$, we have:

$$\min\{H(x, W, DW), W(x) - \varphi(x)\} \leq 0$$

- W is a *viscosity super-solution* of (7) in O *with the boundary condition* (6) if it is a viscosity super-solution of (7) in O and for all functions $\varphi \in C^1(\bar{O})$, for all $x \in \partial O$ local minimum of $W - \varphi$ such that $W(x) = \varphi(x)$, we have:

$$\min\{H(x, W, DW), W(x) - \varphi(x)\} \geq 0 \tag{13}$$

- W is a *viscosity solution* of (7) in O *with the boundary condition* (6) if it is a viscosity sub- and super-solution of (7) in O with the boundary condition (6).

Remark. When the Hamiltonian H is related to an optimal control problem (which is the case here), the condition (13) is simply equivalent to the boundary inequality (6).

With this definition, Theorem 3 extends to viscosity solutions with boundary conditions. Moreover, from a result of uniqueness, we obtain the following theorem (whose proof is in Crandall et al. (1992) or Fleming and Soner (1993)):

Theorem 4. *Suppose that the hypotheses of Theorem* 2 *hold. Then the value function V is the unique viscosity solution of* (7) *in O with the boundary condition* (6).

Remark. This very important theorem shows the relevance of the viscosity solutions formalism for HJB equations. Moreover this provides us with a very useful framework (as will be illustrated in next few sections) for proving the convergence of numerical approximations to the value function.

Now we study numerical approximations of the value function. We define approximation schemes by discretizing the HJB equation with finite element or finite difference methods, and prove the convergence to the viscosity solution of the HJB equation, thus to the value function of the control problem.

4. Approximation with convergent numerical schemes

4.1. Introduction

The main idea is to discretize the HJB equation into a Dynamic Programming (DP) equation for some stochastic Markovian Decision Process (MDP). For any resolution δ, we can solve the MDP and find the discretized value function V^{δ} by using DP techniques, which are guaranteed to converge since the DP equation is a fixed-point equation satisfying some *strong contraction property* (see Puterman, 1994; Bertsekas, 1987). We are also interested in the convergence properties of the discretized V^{δ} to the value function V as δ decreases to 0.

From Kushner (1990) and Kushner and Dupuis (1992), we define two classes of approximation schemes based on finite difference (FD) (Section 4.2) and finite element (FE) methods (Section 4.3). Section 4.4 provides a very general theorem of convergence (deduced from the abstract formulation of Barles and Souganidis (1991) and using the stability properties of viscosity solutions), that covers both FE and FD methods (the only important required properties for the convergence are the *monotonicity* and the *consistency* of the scheme).

In the following, we assume that the control space U is approximated by finite control spaces U^{δ} such that: $\delta \leq \delta' \Rightarrow U^{\delta'} \subset U^{\delta}$ and: $\overline{\bigcup_{\delta} U^{\delta}} = U$.

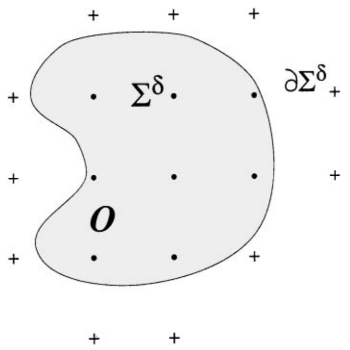

Figure 5. The discretized state-space Σ^δ (the dots) and its frontier $\partial\Sigma^\delta$ (the crosses).

4.2. Approximation with finite difference methods

Let $e_1, e_2, \ldots, e_d$ be a basis for $\mathbb{R}^d$. The dynamics are: $f = (f_1, \ldots, f_d)$. Let the positive and negative parts of f_i be: $f_i^+ = \max(f_i, 0)$, $f_i^- = \max(-f_i, 0)$. For any *discretization step* δ, let us consider the lattices: $\delta\mathbf{Z}^d = \{\delta \cdot \sum_{i=1}^d j_i e_i\}$ where $j_1, \ldots, j_d$ are any integers, and $\Sigma^\delta = \delta\mathbf{Z}^d \cap O$. Let $\partial\Sigma^\delta$, the *frontier* of Σ^δ, denote the set of points $\{\xi \in \delta\mathbf{Z}^d \backslash O$ such that at least one adjacent point $\xi \pm \delta e_i \in \Sigma^\delta\}$ (see figure 5). Let us denote by $\|y\|_1 = \sum_{i=1}^d |y_i|$ the 1-norm of any vector y.

The FD method consists of replacing the gradient $DV(\xi)$ by the forward and backward difference quotients of V at $\xi \in \Sigma^\delta$ in direction e_i:

$$\Delta_i^+ V(\xi) = \frac{1}{\delta}\left[V(\xi + \delta e_i) - V(\xi)\right]$$
$$\Delta_i^- V(\xi) = \frac{1}{\delta}\left[V(\xi - \delta e_i) - V(\xi)\right]$$

Thus the HJB equation can be approximated by the following equation:

$$V^\delta(\xi)\ln\gamma + \sup_{u \in U^\delta}\left\{\sum_{i=1}^d \left[f_i^+(\xi,u)\cdot\Delta_i^+ V^\delta(\xi) + f_i^-(\xi,u)\cdot\Delta_i^- V^\delta(\xi)\right] + r(\xi,u)\right\} = 0$$

Knowing that $(\Delta t \ln\gamma)$ is an approximation of $(\gamma^{\Delta t} - 1)$ as Δt tends to 0, we deduce the following equivalent approximation equation: for $\xi \in \Sigma^\delta$,

$$V^\delta(\xi) = \sup_{u \in U^\delta}\left\{\gamma^{\frac{\delta}{\|f(\xi,u)\|_1}} \sum_{\xi'} p(\xi' \mid \xi, u)\cdot V^\delta(\xi') + \frac{\delta}{\|f(\xi,u)\|_1} r(\xi,u)\right\} \tag{14}$$

$$\text{with } p(\xi' \mid \xi, u) = \begin{cases} \frac{f_i^\pm(\xi,u)}{\|f(\xi,u)\|_1} & \text{for } \xi' = \xi \pm \delta e_i \\ 0 & \text{otherwise} \end{cases}$$

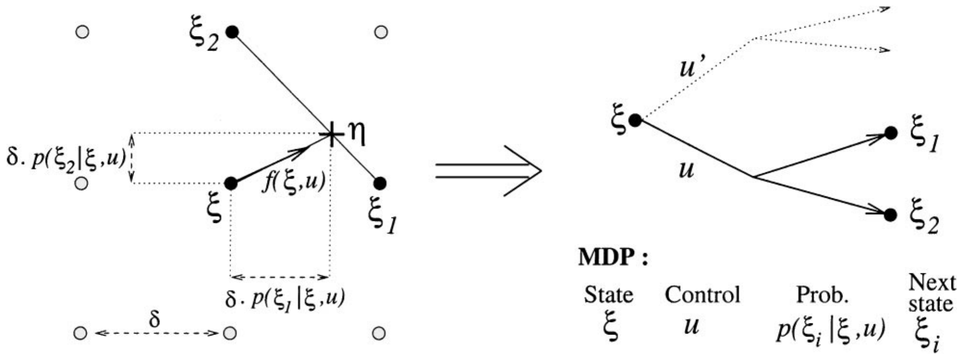

Figure 6. *A geometrical interpretation of the FD discretization.* The continuous process (on the left) is discretized at some resolution δ into an MDP (right). The transition probabilities $p(\xi_i \mid \xi, u)$ of the MDP are the coordinates of the vector $\frac{1}{\delta}(\eta - \xi)$ with η the projection of ξ onto the segment $(\xi + \delta \cdot e_1, \xi + \delta \cdot e_2)$ in a direction parallel to $f(\xi, u)$.

which is a DP equation for a finite Markovian Decision Process whose *state-space* is Σ^δ, *control space* is U^δ and *probabilities of transition* are $p(\xi' \mid \xi, u)$ (see figure 6 for a geometrical interpretation).

From the boundary condition, we define the absorbing terminal states:

$$\text{For } \xi \in \partial\Sigma^\delta, \quad V^\delta(\xi) = R(\xi) \tag{15}$$

By defining F^δ_{FD} the finite difference scheme:

$$F^\delta_{FD}[\varphi](\xi) = \sup_{u \in U^\delta} \left\{ \gamma^{\frac{\delta}{\|f(\xi,u)\|_1}} \sum_{\xi'} p(\xi' \mid \xi, u) \cdot \varphi(\xi') + \frac{\delta}{\|f(\xi, u)\|_1} \cdot r(\xi, u) \right\} \tag{16}$$

DP Eq. (14) becomes: for $\xi \in \Sigma^\delta$,

$$V^\delta(\xi) = F^\delta_{FD}[V^\delta](\xi) \tag{17}$$

This equation states that V^δ is a fixed point of F^δ_{FD}. Moreover, as f is bounded from above (with some value M_f), F^δ_{FD} satisfies the following *strong contraction property*:

$$\left\| F^\delta_{FD}[\varphi_1] - F^\delta_{FD}[\varphi_2] \right\| \leq \lambda \cdot \|\varphi_1 - \varphi_2\| \quad \text{with } \lambda = \gamma^{\frac{\delta}{M_f}} \tag{18}$$

and since $\lambda < 1$, there exists a fixed point which is the value function V^δ; it is unique and can be computed by DP iterative methods (see Puterman, 1994; Bertsekas, 1987).

Computation of V^δ and convergence. There are two standard methods for computing the value function V^δ of some MDP: *value iteration* (V^δ is the limit of a sequence of successive iterations $V^\delta_{n+1} = F^\delta_{FD}[V^\delta_n]$) and *policy iteration* (approximations in policy space by alternative policy evaluation steps and policy improvement steps). See Puterman (1994), Bertsekas (1987) or Bertsekas and Tsitsiklis (1996) for more information about DP theory. In Section 5, we describe RL methods for computing iteratively the approximated value functions V^δ.

In the following section, we study a similar method for discretizing the continuous process into an MDP by using finite element methods. The convergence of these two methods (i.e. the convergence of the discretized V^δ to the value function V as δ tends to 0) will be derived from a general theorem in Section 4.4.

4.3. Approximations with finite element methods

We use a finite element (FE) method (with linear simplexes) based on a triangulation Σ^δ covering the state-space (see figure 7).

The value function V is approximated by piecewise linear functions V^δ defined by their values at the vertices $\{\xi\}$ of the triangulation Σ^δ. The value of V^δ at any point x inside some simplex $(\xi_0, \ldots, \xi_d)$ is a linear combination of V^δ at the vertices $\xi_0, \ldots, \xi_d$ (see figure 7):

$$V^\delta(x) = \sum_{i=0}^{d} \lambda_{\xi_i}(x) V^\delta(\xi_i) \quad \text{for all } x \in \text{Simplex } (\xi_0, \ldots, \xi_d)$$

with $\lambda_{\xi_i}(x)$ being the *barycentric coordinates* of x inside the simplex $(\xi_0, \ldots, \xi_d) \ni x$. (We recall that the definition of the barycentric coordinates is that $\lambda_{\xi_i}(x)$ satisfy: $\sum_{i=0}^{d} \lambda_{\xi_i}(x) \cdot (\xi_i - x) = 0$, $\sum_{i=0}^{d} \lambda_{\xi_i}(x) = 1$ and $\lambda_{\xi_i}(x) \geq 0$).

By using a finite element approximation scheme derived from Kushner (1990), the continuous HJB equation is approximated by the following equation:

$$V^\delta(\xi) = \sup_{u \in U^\delta} \left[\gamma^{\tau(\xi,u)} \cdot V^\delta(\eta(\xi, u)) + \tau(\xi, u) r(\xi, u)\right]$$

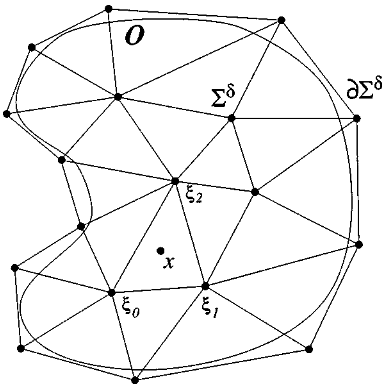

Figure 7. Triangulation Σ^δ of the state-space. $V^\delta(x)$ is a linear combination of the $V^\delta(\xi_i)$, for $x \in$ simplex (ξ_0, ξ_1, ξ_2), weighted by the barycentric coordinates $\lambda_{\xi_i}(x)$.

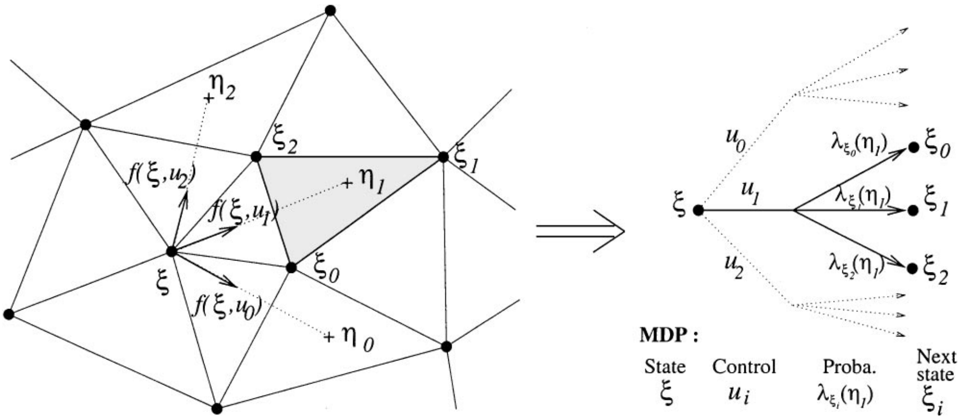

Figure 8. A finite element approximation. Consider a vertex ξ and $\eta_1 = \xi + \tau(\xi, u_1).f(\xi, u_1)$. $V^\delta(\eta_1)$ is a linear combination of $V^\delta(\xi_0)$, $V^\delta(\xi_1)$, $V^\delta(\xi_2)$ weighted by the barycentric coordinates $\lambda_{\xi_i}(\eta_1)$. Thus, the probabilities of transition of the MDP are these barycentric coordinates.

where $\eta(\xi, u)$ is a point inside Σ^δ such that $\eta(\xi, u) = \xi + \tau(\xi, u) \cdot f(\xi, u)$ (see figure 8) for some "time discretization" function $\tau : \Sigma^\delta \times U^\delta \to \mathbb{R}$. We require that τ satisfies the following condition, for some positive constants k_1 and k_2:

$$\forall \xi \in \Sigma^\delta, \quad \forall u \in U^\delta, \quad k_1 \cdot \delta \leq \tau(\xi, u) \leq k_2 \cdot \delta \tag{19}$$

Remark. It is interesting to notice that this time discretization function $\tau(\xi, u)$ does not need to be constant and can depend on the state ξ and the control u. This provides us with some freedom on the choice of these parameters, assuming that Eq. (19) still holds. For a discussion on the choice of a constant time discretization function τ according to the space discretization size δ in order to optimize the precision of the approximations, see Pareigis (1997).

Let us denote $(\xi_0, \ldots, \xi_d)$ the simplex containing $\eta(\xi, u)$. As V^δ is linear inside the simplex, this equation can be written:

$$V^\delta(\xi) = \sup_{u \in U^\delta} \left[\gamma^{\tau(\xi,u)} \cdot \sum_{i=1}^{d} \lambda_{\xi_i}(\eta(\xi, u)) V^\delta(\xi_i) + \tau(\xi, u) r(\xi, u) \right] \tag{20}$$

which is a DP equation for a Markov Decision Process whose state-space is the set of vertices $\{\xi\}$ and the probability of transition from (state ξ, control u) to next states $\xi' \in \{\xi_0, \ldots, \xi_d\}$ are the barycentric coordinates of $\eta(\xi, u)$ inside simplex $(\xi_0, \ldots, \xi_d)$ (and 0 for $\xi' \notin \{\xi_0, \ldots, \xi_d\}$) (see figure 8). The boundary states satisfy the terminal condition:

$$\text{For } \xi \in \partial\Sigma^\delta, \quad V^\delta(\xi) = R(\xi). \tag{21}$$

By defining F_{FE}^{δ} the finite element scheme,

$$F_{FE}^{\delta}[\varphi](\xi) = \sup_{u \in U^{\delta}} \left\{ \gamma^{\tau(\xi,u)} \cdot \sum_{i=1}^{d} \lambda_{\xi_i}(\eta(\xi,u))\varphi(\xi_i) + \tau(\xi,u) r(\xi,u) \right\} \tag{22}$$

the approximated value function V^{δ} satisfies the DP equation

$$V^{\delta}(\xi) = F_{FE}^{\delta}[V^{\delta}](\xi). \tag{23}$$

Similarly to the FD scheme, F_{FE}^{δ} satisfies the following “strong” contraction property:

$$\left\| F_{FE}^{\delta}[\varphi_1] - F_{FE}^{\delta}[\varphi_2] \right\| \leq \lambda \cdot \|\varphi_1 - \varphi_2\| \quad \text{with } \lambda = \gamma^{k_1 \delta}. \tag{24}$$

and since $\lambda < 1$, there is a unique solution, V^{δ} to (23) with (21) which can be computed by DP techniques.

4.4. *Convergence of the approximation schemes*

In this section, we present a convergence theorem for a general class of approximation schemes. We use the stability properties of viscosity solutions (described in Barles and Souganidis, 1991) to obtain the convergence. Another kind of convergence result, using probabilistic considerations, can be found in Kushner and Dupuis (1992), but such results do not treat the problem with the general boundary condition (9). In fact, the only important required properties for convergence is monotonicity (property (27)) and consistency (properties (30) and (31) below). As a corollary, we deduce that the FE and the FD schemes studied in the previous sections are convergent.

4.4.1. A general convergence theorem. Let Σ^{δ} and $\partial\Sigma^{\delta}$ be two discrete and finite subsets of $\mathbb{R}^d$. We assume that for all $x \in O$, $\lim_{\delta\downarrow 0} dist(x, \Sigma^{\delta}) = 0$ and for all $x \in \partial O$, $\lim_{\delta\downarrow 0} dist(x, \partial\Sigma^{\delta}) = 0$. Let F^{δ} be an operator on the space of bounded functions on Σ^{δ}. We are concerned with the convergence of the solution V^{δ} to the dynamic programming equation:

$$V^{\delta}(\xi) = F^{\delta}[V^{\delta}](\xi) \quad \text{for } \xi \in \Sigma^{\delta} \tag{25}$$

with the boundary condition:

$$V^{\delta}(\xi) = R(\xi) \quad \text{for } \xi \in \partial\Sigma^{\delta} \tag{26}$$

We make the following assumptions on F^{δ}:

- *Monotonicity*:

$$\text{if } \varphi_1 \leq \varphi_2 \text{ then } F^{\delta}[\varphi_1] \leq F^{\delta}[\varphi_2] \tag{27}$$

- For any constant c,

$$F^{\delta}[\varphi + c] = F^{\delta}[\varphi] + c(1 + O(\delta)) \tag{28}$$

- For any δ,

  there exists a solution V^{δ} to (25) and (26) which is
  bounded with a constant M_V independent of δ. (29)

- *Consistency*: there exists a constant $k > 0$ such that:
  - if $H(x, \varphi(x), D\varphi(x)) \geq 0$ then

$$\liminf_{\xi_\delta \xrightarrow{\delta\downarrow 0} x} \frac{1}{\delta}[\varphi - F^{\delta}[\varphi]](\xi_\delta) \geq k.H(x, \varphi(x), D\varphi(x)) \tag{30}$$

  - if $H(x, \varphi(x), D\varphi(x)) \leq 0$ then

$$\limsup_{\xi_\delta \xrightarrow{\delta\downarrow 0} x} \frac{1}{\delta}[\varphi - F^{\delta}[\varphi]](\xi_\delta) \leq k.H(x, \varphi(x), D\varphi(x)) \tag{31}$$

Remark. Conditions (30) and (31) are satisfied in the particular case of:

$$\lim_{\xi_\delta \xrightarrow{\delta\downarrow 0} (\xi_\delta} \frac{1}{\delta}[\varphi - F^{\delta}[\varphi]](\xi_\delta) = H(x, \varphi(x), D\varphi(x))$$

Theorem 5 (*Convergence of the scheme***).** *Assume that the hypotheses of Theorem* 2 *are satisfied. Assume that* (27), (28), (30) *and* (31) *hold, then F^{δ} is a convergent approximation scheme, i.e. the solutions V^{δ} of* (25) *and* (26) *satisfy*:

$$\lim_{\xi_\delta \xrightarrow{\delta\downarrow 0} x} V^{\delta}(\xi_\delta) = V(x) \text{ uniformly on any compact } \Omega \subset O \tag{32}$$

4.4.2. Outline of the proof. We use the procedure described in Barles and Perthame (1988). The idea is to define the largest limit function $V_{\sup} = \limsup V^{\delta}$ and the smallest limit function $V_{\inf} = \liminf V^{\delta}$ and prove that they are respectively discontinuous sub- and super viscosity solutions. This proof, based on the general convergence theorem of Barles and Souganidis (1991), is given in Appendix A. Then we use a comparison result which states that if (9) holds then viscosity sub-solutions are less than viscosity super-solutions, thus $V_{\sup} \leq V_{\inf}$. By definition $V_{\sup} \geq V_{\inf}$, thus $V_{\sup} = V_{\inf} = V$ and the limit function V is the viscosity solution of the HJB equation, thus (from Theorem 4) the value function of the problem.

4.4.3. FD and FE approximation schemes converge

Corollary 1. *The approximation schemes F^{δ}_{FD} and F^{δ}_{FE} are convergent.*

Indeed, for the finite difference scheme, it is obvious that since $p(\xi' \mid \xi, u)$ are considered as transition probabilities, the approximation scheme F^{δ}_{FD} satisfies (27) and (28). As (17) is a DP equation for some MDP, DP theory ensures that (29) is true. We can check that the scheme is also consistent: conditions (30) and (31) hold with $k = \frac{1}{M_f}$. Thus F^{δ}_{FD} satisfy the hypotheses of Theorem 5.

Similarly, for the finite element scheme, from the basic properties of the barycentric coordinates $\lambda_{\xi_i}(x)$, the approximation scheme F^{δ}_{FE} satisfies (27). From (19), condition (28) holds. DP theory ensures that (29) is true. The scheme is consistent and conditions (30) and (31) hold with $k = k_1$. Thus F^{δ}_{FE} satisfies the hypotheses of Theorem 5.

4.5. Summary of the previous results of convergence

For any given discretization step δ, from the "strong" contraction property (18) or (24), DP theory ensures that the values V^{δ}_n iterated by some DP algorithm converge to the value V^{δ} of the approximation scheme F^{δ} as n tends to infinity. From the convergence of the scheme (Theorem 5), the V^{δ} tend to the value function V of the continuous problem as δ tends to 0 (see figure 9).

Remark. Theorem 5 gives a result of convergence on any compact $\Omega \subset O$, provided that the hypothesis (9), for the continuity of V, is satisfied. However, if this hypothesis is not satisfied, but if the value function is continuous, the theorem still applies. Now, if (9) is not satisfied and the value function is discontinuous at some area, then we still have the convergence on any compact $\Omega \subset O$ where the value function is continuous.

5. Designing convergent reinforcement learning algorithms

In order to solve the DP Eqs. (14) or (20), one can use DP off-line methods—such as *value iteration*, *policy iteration*, *modified policy iteration* (see Puterman, 1994), with synchronous or asynchronous back-ups, or on-line methods—like *Real Time DP* (see Barto, Bradtke, & Singh, 1991; Bertsekas & Tsitsiklis, 1996). For example, by introducing the *Q-values* $Q^{\delta}_n(\xi, u)$, Eq. (20) can be solved by successive back-ups (indexed by n) of states ξ, control u (in any order provided that every state and control are updated regularly) by:

$$\begin{aligned} &Q^{\delta}_{n+1}(\xi, u) = \gamma^{\tau(\xi,u)} \cdot V^{\delta}_n(\xi + \tau(\xi, u) \cdot f(\xi, u)) + \tau(\xi, u) r(\xi, u) \\ &\text{with: } V^{\delta}_n(\xi) = \sup_{u \in U^{\delta}} Q^{\delta}_n(\xi, u) \end{aligned} \tag{33}$$

$$\begin{tikzcd}
\textbf{HJB equation} & \textbf{DP equation} \\
V & V^{\delta} \arrow[l, "\delta \to 0"'] \\
 & V^{\delta}_n \arrow[u, "n \to \infty", "\text{DP with the ``strong'' contraction property}"']
\end{tikzcd}$$

Figure 9. The HJB equation is discretized, for some resolution δ, into a DP equation whose solution is V^{δ}. The convergence of the scheme ensures that $V^{\delta} \to V$ as $\delta \to 0$. Thanks to the "strong" contraction property, the iterated values V^{δ}_n tend to V^{δ} as $n \to \infty$.

The values V_n^δ of this algorithm converges to the value function V^δ of the discretized MDP as $n \to \infty$.

However, in the RL approach, the state dynamics f and the reinforcement functions r, R are unknown to the learner. Thus, the right side of the iterative rule (33) is unknown and has to be approximated thanks to the available knowledge. In the RL terminology, there are two possible approaches for updating the values:

- The *model-based* approach consists of first, learning a model of the state dynamics and of the reinforcement functions, and then using DP algorithms based on such a rule (33) with the approximated model instead of the true values. The learning (the updating of the estimated Q-value Q_n^δ) may occur iteratively during the simulation of trajectories (*on-line learning*) or at the end (at the exit time) of one or several trajectories (*off-line* or *batch learning*).
- The *model-free* approach consists of updating incrementally the estimated values V_n^δ or Q-value Q_n^δ of the visited states without learning any model.

In what follows, we propose a convergence theorem that applies to a large class of RL algorithms (model-based or model-free, on-line or off-line, for deterministic or stochastic dynamics) provided that the updating rule satisfies some **"weak" contraction property** with respect to some convergent approximation scheme such as the FD and FE schemes studied previously.

5.1. Convergence of RL algorithms

The following theorem gives a general condition for which an RL algorithm converges to the optimal solution of the continuous problem. The idea is that the updated values (by any model-free or model-based method) must be close enough to those of a convergent approximation scheme so that their difference satisfies the "weak" contraction property (34) below.

Theorem 6 (*Convergence of RL algorithms***).** *For any δ, let us build finite subsets Σ^δ and $\partial\Sigma^\delta$ satisfying the properties of Section* 4.4. *We consider an algorithm that leads to update every state $\xi \in \Sigma^\delta$ regularly and every state $\xi \in \partial\Sigma^\delta$ at least once. Let F^δ be a convergent approximation scheme (for example* (22)*) and V^δ be the solution to* (25) *and* (26). *We assume that the values updated at the iteration n satisfy the following properties*:

- *for $\xi \in \Sigma^\delta, V_{n+1}^\delta(\xi)$ approximates $V^\delta(\xi)$ in the sense of the following "weak" contraction property*:

$$\left|V_{n+1}^\delta(\xi) - V^\delta(\xi)\right| \leq (1 - k.\delta) \sup_{\xi \in \Sigma^\delta \cup \partial\Sigma^\delta} \left|V_n^\delta(\xi) - V^\delta(\xi)\right| + e(\delta).\delta \tag{34}$$

  for some positive constant k and some function $e(\delta)$ that tends to 0 *as* $\delta \downarrow 0$,
- *for $\xi \in \partial\Sigma^\delta$, $V_{n+1}^\delta(\xi)$ approximates $V^\delta(\xi) = R(\xi)$, in the sense*:

$$\left|V_{n+1}^\delta(\xi) - R(\xi)\right| \leq k_R.\delta \tag{35}$$

for some positive constant k_R, then for any compact $\Omega \subset O$, for all $\varepsilon > 0$, there exists Δ such that for any $\delta \leq \Delta$, there exists N, for all $n \geq N$,

$$\sup_{\xi \in \Omega \cap (\Sigma^\delta \cup \partial \Sigma^\delta)} \left| V_n^\delta(\xi) - V(\xi) \right| \leq \varepsilon.$$

This result states that when the hypotheses of the theorem applies (mainly when we find some updating rule satisfying the weak contraction property (34)) then the values V_n^δ computed by the algorithm converge to the value function V of the continuous problem as the discretization step δ tends to zero and the number of iterations n tends to infinity.

5.1.1. Outline of the proof. The proof of this theorem is given in Appendix B. If condition (34) were a strong contraction property such as

$$\left| V_{n+1}^\delta(\xi) - V^\delta(\xi) \right| \leq \lambda \sup_{\xi \in \Sigma^\delta \cup \partial \Sigma^\delta} \left| V_n^\delta(\xi) - V^\delta(\xi) \right| \tag{36}$$

for some constant $\lambda < 1$, then the convergence would be obvious since from (25) and from the fact that all the states are updated regularly, for a fixed δ, V_n^δ would converge to V^δ as $n \to \infty$. From the fact (Theorem 5) that V^δ converges to V as $\delta \downarrow 0$, we could deduce that $V_n^\delta \to V$ as $\delta \downarrow 0$ and $n \to \infty$ (see figure 9).

If it is not the case, we can no longer expect that $V_n^\delta \to V^\delta$ as $n \to \infty$. However, if (34) holds, we can prove (this is the object of Section B.2 in Appendix B) that for any $\varepsilon > 0$, there exists small enough values of δ such that at some stage N, $|V_n^\delta - V^\delta| \leq \varepsilon$ for $n \geq N$. This result together with the convergence of the scheme leads to the convergence of the algorithm as $\delta \downarrow 0$ and $n \to \infty$ (see figure 10).

5.1.2. The challenge of designing convergent algorithms. In general the "strong" contraction property (36) is impossible to obtain unless we have perfect knowledge of the dynamics f and the reinforcement functions r and R. In the RL approach, these components are estimated and approximated during some learning phase. Thus the iterated values V_n^δ are imperfect, but may be "good enough" to satisfy the weak contraction property (34). **Defining such "good" approximations is the challenge for designing convergent RL algorithms**.

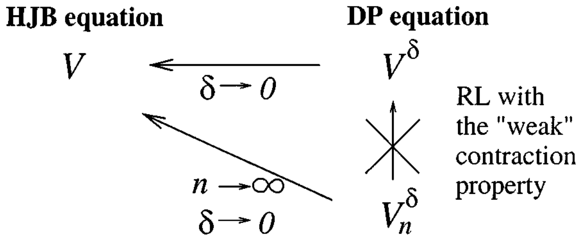

Figure 10. The values V_n^δ iterated by an RL algorithm do not converge to V^δ as $n \to \infty$. However, if the "weak" contraction property is satisfied, the V_n^δ tend to V as $\delta \to 0$ and $n \to \infty$.

In order to illustrate the method, we present in Section 5.2 a procedure for designing model-based algorithms, and in Section 5.3, we give a model-free algorithm based on a FE approximation scheme.

5.2. Model-based algorithms

The basic idea is to build a model of the state dynamics f and the reinforcement functions r and R at states ξ from the local knowledge obtained through the simulation of trajectories. So, if some trajectory $x_n(t)$ goes inside the neighborhood of ξ (by defining the neighborhood as an area whose diameter is bounded by $k_N.\delta$ for some positive constant k_N) at some time t_n and keep a constant control u for a period τ_n (from $x_n = x_n(t_n)$ to $y_n = x_n(t_n + \tau_n)$), we can build the model of $f(\xi, u)$ and $r(\xi, u)$:

$$\tilde{f}_n(\xi, u) = \frac{y_n - x_n}{\tau_n}$$
$$\tilde{r}(\xi, u) = r(x_n, u)$$

(see figure 11). Then we can approximate the scheme (22), by the following values using the previous model: the Q-values Q_n^δ are updated according to:

$$Q_{n+1}^\delta(\xi, u) = \gamma^{\tau(\xi,u)} \cdot V_n^\delta(\xi + \tau(\xi, u) \cdot \tilde{f}_n(\xi, u)) + \tau(\xi, u) \cdot \tilde{r}(\xi, u)$$
$$\text{and } V_n^\delta(\xi) = \sup_{u \in U^\delta} Q_n^\delta(\xi, u)$$

(for any function $\tau(\xi, u)$ satisfying (19)), which corresponds to the iterative rule (33) with the model $\tilde{f}_n$ and $\tilde{r}$ instead of f and r.

It is easy to prove (see Munos and Moore (1998) or Munos (1997a)) that assuming some smoothness assumptions (r, R Lipschitzian), the approximated V_n^δ satisfy the condition (34) and theorem 6 applies.

Remark. Using the same model, we can build a similar convergent RL algorithm based on the finite difference scheme (22) (see Munos, 1998). Thus, it appears quite easy to design model-based algorithms satisfying the condition (34).

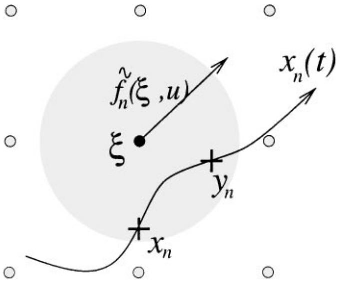

Figure 11. A trajectory goes through the neighborhood (the grey area) of ξ. The state dynamics is approximated by $\tilde{f}_n(\xi, u) = \frac{y_n - x_n}{\tau_n}$.

Remark. This method can also be used in the stochastic case, for which a model of the state dynamics is the *average*, for several trajectories, of such $\frac{y_n - x_n}{\tau_n}$, and a model of the noise is their *variance* (see Munos and Bourgine, 1997).

Furthermore, it is possible to design model-free algorithms satisfying the condition (34), which is the topic of the following section.

5.3. A model-free algorithm

The Finite Element RL algorithm. Consider a triangulation Σ^δ satisfying the properties of Section 4.3. The direct RL approach consists of updating on-line the Q-values of the vertices without learning any model of the dynamics.

We consider the FE scheme (22) with $\tau(\xi, u)$ being such that $\eta(\xi, u) = \xi + \tau(\xi, u)$. $f(\xi, u)$ is the projection of ξ onto the opposite side of the simplex, in a parallel direction to $f(\xi, u)$ (see figure 12). If we suppose that the simplexes are non degenerated (i.e. $\exists k_\rho$ such that the radius of the sphere inscribed in each simplex is superior to $k_\rho \delta$) then (19) holds.

Let us consider that a trajectory $x(t)$ goes through a simplex. Let $x = x(t_1)$ be the input point and $y = x(t_2)$ be the output point. The control u is assumed to be kept constant inside the simplex.

As the values $\tau(\xi, u)$ and $\eta(\xi, u)$ are unknown to the system, we make the following estimations (from Thales' theorem):

- $\tau(\xi, u)$ is approximated by $\frac{\tau}{\lambda_\xi(x)}$ (where $\lambda_\xi(x)$ is the ξ−barycentric coordinate of x inside the simplex)
- $\eta(\xi, u)$ is approximated by $\xi + \frac{y - x}{\lambda_\xi(x)}$

which only use the knowledge of the state at the input and output points (x and y), the running time τ of the trajectory inside the simplex and the barycentric coordinate $\lambda_\xi(x)$ (which can be computed as soon as the system knows the vertices of the input side of the simplex). Besides, $r(\xi, u)$ is approximated by the current reinforcement $r(x, u)$ at the input point.

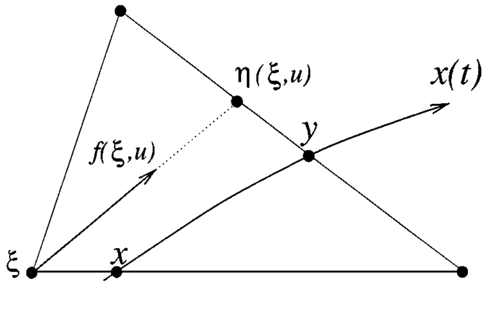

Figure 12. A trajectory going through a simplex. $\eta(\xi, u)$ is the projection of ξ onto the opposite side of the simplex. $\frac{y - x}{\lambda_\xi(x)}$ is a good approximation of $\eta(\xi, u) - \xi$.

Thanks to the linearity of V^δ inside the simplex, $V^\delta(\eta(\xi, u))$ is approximated by $V^\delta(\xi) + \frac{V^\delta(y) - V^\delta(x)}{\lambda_\xi(x)}$. Then the algorithm consists in updating the quality $Q_n^\delta(\xi, u)$ with the estimation:

$$Q_{n+1}^\delta(\xi, u) = \gamma^{\frac{\tau}{\lambda_\xi(x)}} \cdot \left[V_n^\delta(\xi) + \frac{V_n^\delta(y) - V_n^\delta(x)}{\lambda_\xi(x)} \right] + \frac{\tau}{\lambda_\xi(x)} \cdot r(x, u) \tag{37}$$

$$\text{and } V_n^\delta(\xi) = \sup_{u \in U^\delta} Q_n^\delta(\xi, u) \tag{38}$$

and if the system exits from the state-space inside the simplex (i.e. $y \in \partial O$), then update the closest vertex $\xi' \in \partial \Sigma^\delta$ of the simplex with:

$$V_{n+1}^\delta(\xi') = R(y).$$

By assuming some additional regularity assumptions (r and R Lipschitzian, f bounded from below), the values $V_n^\delta(\xi)$ satisfy (34) and (35) which proves the convergence of the model-free RL algorithm based on the FE scheme (see Munos (1996) for the proof).

In a similar way, we can design a direct RL algorithm based on the finite difference scheme F_{FD}^δ (16) and prove its convergence (see Munos, 1997b).

6. A numerical simulation for the "Car on the Hill" problem

For a description of the dynamics of this problem, see Moore and Atkeson (1995). This problem has a state-space of dimension 2: the position and the velocity of the car. In our experiments, we chose the reinforcement functions as follows: the current reinforcement $r(x, u)$ is zero everywhere. The terminal reinforcement $R(x)$ is -1 if the car exits from the left side of the state-space, and varies linearly between $+1$ and -1 depending on the velocity of the car when it exits from the right side of the state-space. The best reinforcement $+1$ occurs when the car reaches the right boundary with a null velocity (see figure 13). The control u has only 2 possible values: maximal positive or negative thrust.

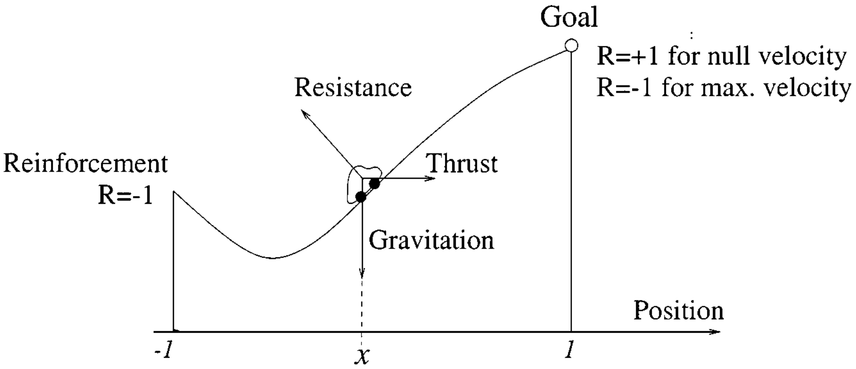

Figure 13. The "Car on the Hill" problem.

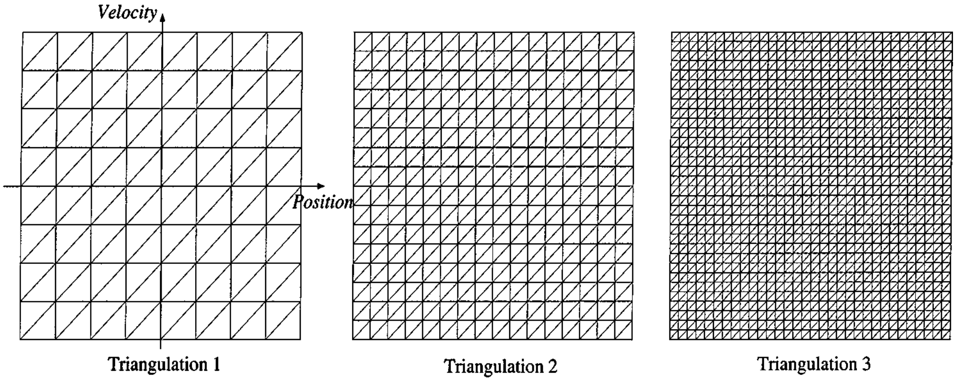

Figure 14. Three triangulations used for the simulations.

In order to approximate the value function, we used 3 different triangulations T_1, T_2 and T_3, composed respectively of 9 by 9, 17 by 17 and 33 by 33 states (see figure 14), and, for each of these, we ran the two algorithms that follows:

- An asynchronous Real Time DP (based on the updating rule (33)), assuming that we have a perfect model of the initial data (the state dynamics and the reinforcement functions).
- An asynchronous Finite Element RL algorithm, described in Section 5.3 (based on the updating rule (37)), for which the initial data are approximated by parts of trajectories selected at random.

In order to evaluate the quality of approximation of these methods, we also computed a very good approximation of the value function V (plotted in figure 15) by using DP

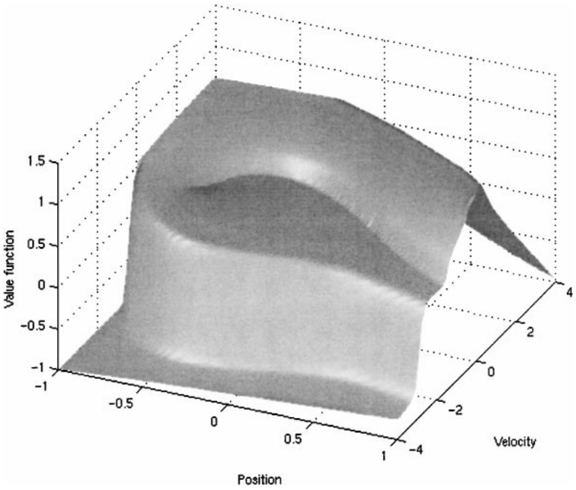

Figure 15. The value function of the “Car on the Hill”, computed with a triangulation composed of 257 by 257 states.

(with rule (33)) on a very dense triangulation (of 257 by 257 states) with a perfect model of the initial data.

We have computed the approximation error $E_n(T_k) = \sup_{\xi \in \Omega} |V_n^{\delta_k}(\xi) - V(\xi)|$ with δ_k being the discretization step of triangulation T_k. For this problem, we notice that hypothesis (9) does not hold (because all the trajectories are tangential to the boundary of the state-space at the boundary states of zero velocity), and the value function is discontinuous. A frontier of discontinuity happens because a point beginning just above this frontier can eventually get a positive reward whereas any point below is doomed to exit on the left side of the state-space. Thus, following the remark in Section 4.5, in order to compute $E_n(T_k)$, we chose Ω to be the whole state-space except some area around the discontinuity.

Figures 16 and 17 represent, respectively for the 2 algorithms, the approximation error $E_n(T_k)$ (for the 3 triangulations T_1, T_2 and T_3) as a function of the number of iterations n. We observe the following points:

- Whatever the resolution of the discretization δ is, the values V_n^δ computed by RTDP converge, as n increases. Their limit is V^δ, solution of the DP Eq. (20). Moreover, we observe the convergence of the V^δ to the value function V as the resolution δ tends to zero. These results illustrate the convergence properties showed in figure 9.
- For a given triangulation, the values V_n^δ computed by FERL do not converge. For T_1 (rough discretization), the error of approximation decreases rapidly, and then oscillates within a large range. For T_2, the error decreases more slowly (because there are more states to be updated) but then oscillates within a smaller range. And for T_3 (dense discretization), the error decreases still more slowly but eventually gets close to zero (while still oscillating). Thus, we observe that, as illustrated in figure 10, for any given discretization step δ, the values do not converge. However, they oscillate within a range depending on δ. Theorem 6 simply states that for any desired precision ($\forall\varepsilon$), there exists

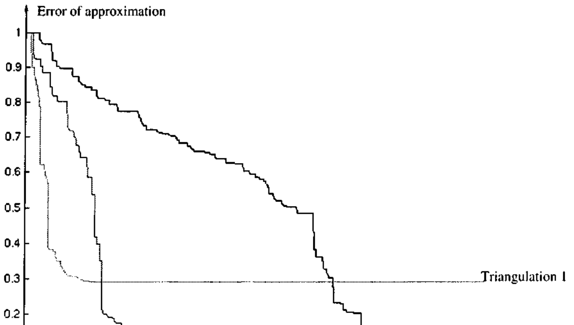

Figure 16. The approximation error $E_n(T_k)$ of the values computed by the asynchronous Real Time DP algorithm as a function of the number of iterations n for several triangulations.

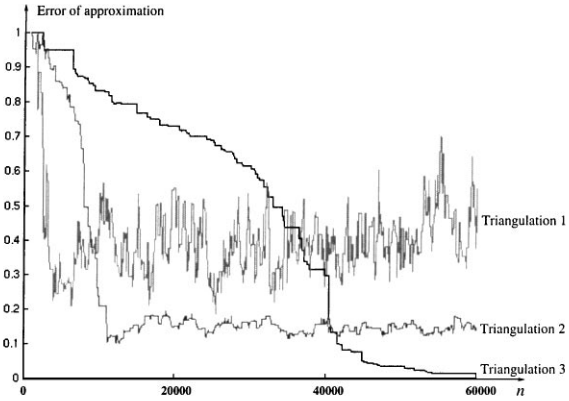

Figure 17. The approximation error $E_n(T_k)$ of the values computed by the asynchronous Finite Element RL algorithm.

a discretization step δ such that eventually $(\exists N, \forall n > N)$, the values will approximate the value function at that precision $(\sup |V_n^\delta - V| < \varepsilon)$.

7. Conclusion and future work

This paper proposes a formalism for the study of RL in the continuous state-space and time case. The Hamilton-Jacobi-Bellman equation is stated and several properties of its solutions are described. The notion of viscosity solution is introduced and used to integrate the HJB equation for finding the value function. We describe discretization methods (by using finite element and finite difference schemes) for approximating the value function, and use the stability properties of the viscosity solutions to prove their convergence.

Then, we propose a general method for designing convergent (model-based or model-free) RL algorithms and illustrate it with several examples. The convergence result is obtained by substituting the "strong" contraction property used to prove the convergence of DP method (which cannot hold any more when the initial data are not perfectly known) by some "weak" contraction property, that enables some approximations of these data. The main theorem states a convergence result for RL algorithms as the discretization step δ tends to 0 and the number of iterations n tends to infinity.

For practical applications of this method, we must combine to the *learning dynamics* ($n \to \infty$) some *structural dynamics* ($\delta \to 0$) which operates on the discretization process. For example, in Munos (1997c), an initial rough Delaunay triangulation (high δ) is progressively refined (by adding new vertices) according to a local criterion estimating the irregularities of the value function. In Munos and Moore (1999), a Kuhn triangulation embedded in a kd-tree is adaptively refined by a non-local splitting criterion that allows the cells to take into account their impact on other cells when deciding whether to split.

Future theoretical work should consider the study of approximation schemes (and the design of algorithms based on these scheme) for adaptive and variable resolution discretizations (like the adaptive discretizations of Munos and Moore (1999); Munos (1997c), the partigame algorithm of Moore and Atkeson (1995), the multi-grid methods of Akian (1990) and Pareigis (1996), or the sparse grids of Griebel (1998)), the study of the rates of convergence of these algorithms (which already exists in some cases, see Dupuis and James (1998)), and the study of generalized control problems (with "jumps", generalized boundary conditions, etc.).

To adequately address practical issues, extensive numerical simulations (and comparison to other methods) have to be conducted, and in order to deal with high dimensional state-spaces, future work should concentrate on designing relevant structural dynamics and condensed function representations.

Appendix A: Proof of Theorem 5

A.1. Outline of the proof

We use the Barles and Perthame procedure in Barles and Perthame (1988). First we give a definition of discontinuous viscosity solutions. Then we define the largest limit function $V_{\sup}$ and the smallest limit function $V_{\inf}$ and prove (following Barles & Souganidis, 1991), in Lemma (1), that $V_{\sup}$ (respectively $V_{\inf}$) is a discontinuous viscosity sub-solution (resp. super-solution). Then we use a strong comparison result (Lemma 2) which states that if (9) holds then viscosity sub-solutions are less than viscosity super-solutions, thus $V_{\sup} \leq V_{\inf}$. By definition $V_{\sup} \geq V_{\inf}$, thus $V_{\sup} = V_{\inf} = V$ and the limit function V is the viscosity solution of the HJB equation, and thus the value function of the problem.

A.2. Definition of discontinuous viscosity solutions

Let us recall the notions of the *upper semi-continuous envelope* W^* and the *lower semi-continuous envelope* W_* of a real valued function W:

$$W^*(x) = \limsup_{y \to x} W(y)$$
$$W_*(x) = \liminf_{y \to x} W(y)$$

Definition 5. Let W be a locally bounded real valued function defined on $\bar{O}$.

- W is a *viscosity sub-solution* of $H(x, W, DW) = 0$ in $\bar{O}$ if for all functions $\varphi \in C^1(\bar{O})$, for all $x \in \bar{O}$, local maximum of $W^* - \varphi$ such that $W^*(x) = \varphi(x)$, we have:

$$H_*(x, \varphi(x), D\varphi(x)) \leq 0$$

- W is a *viscosity super-solution* of $H(x, W, DW) = 0$ in $\bar{O}$ if for all functions $\varphi \in C^1(\bar{O})$, for all $x \in \bar{O}$, local minimum of $W_* - \varphi$ such that $W_*(x) = \varphi(x)$, we have:

$$H^*(x, \varphi(x), D\varphi(x)) \geq 0$$

- W is a *viscosity solution* of $H(x, W, DW) = 0$ in $\bar{O}$ if it is a viscosity sub-solution and a viscosity super-solution of $H(x, W, DW) = 0$ in $\bar{O}$.

A.3. V_{sup} and V_{inf} are viscosity sub- and super-solutions

Lemma 1. *The two limit functions $V_{\sup}$ and $V_{\inf}$:*

$$V_{\sup}(x) = \limsup_{\xi \overset{\delta\downarrow 0}{\rightarrow} x} V^{\delta}(\xi)$$

$$V_{\inf}(x) = \liminf_{\xi \overset{\delta\downarrow 0}{\rightarrow} x} V^{\delta}(\xi)$$

are respectively viscosity sub- and super-solutions.

Proof: Let us prove that $V_{\sup}$ is a sub-solution. The proof that $V_{\inf}$ is a super-solution is similar. Let φ be a smooth test function such that $V_{\sup} - \varphi$ has a maximum (which can be assumed to be strict) at x such that $V_{\sup}(x) = \varphi(x)$. Let δ_n be a sequence converging to zero. Then $V^{\delta_n} - \varphi$ has a maximum at ξ_n which tends to x as δ_n tends to 0. Thus, for all $\xi \in \Sigma^{\delta_n}$,

$$V^{\delta_n}(\xi) - \varphi(\xi) \leq V^{\delta_n}(\xi_n) - \varphi(\xi_n)$$

By (27), we have:

$$F^{\delta_n}[V^{\delta_n}(\xi) - V^{\delta_n}(\xi_n) - \varphi(\xi_n)] \leq F^{\delta_n}[\varphi(\xi)]$$

By (28), we obtain:

$$F^{\delta_n}[V^{\delta_n}(\cdot)](\xi_n) - (1 + O(\delta_n))[V^{\delta_n}(\xi_n) - \varphi(\xi_n)] \leq F^{\delta_n}[\varphi(\cdot)](\xi_n)$$

By (29), $F^{\delta_n}[V^{\delta_n}] = V^{\delta_n}$, thus:

$$\frac{1}{\delta_n} O(\delta_n)[V^{\delta_n}(\xi_n) - \varphi(\xi_n)] \leq \frac{1}{\delta_n}[F^{\delta_n}[\varphi(\cdot)](\xi_n) - \varphi(\xi_n)]$$

As $V^{\delta_n}(\xi_n) - \varphi(\xi_n)$ tends to 0, the left side of this inequality tends to 0 as $\delta_n \downarrow 0$. Thus, by (31), we have:

$$H(x, \varphi, D\varphi) \geq 0.$$

Thus $V_{\sup}$ is a viscosity sub-solution. □

A.4. Comparison principle between viscosity sub- and super-solutions

Lemma 2. *Assume* (9), *then* (7) *and* (6) *has a weak comparison principle, i.e. for any viscosity sub-solution* $\underline{W}$ *and super-solution* $\bar{W}$ *of* (7) *and* (6), *for all* $x \in O$ *we have*:

$$\underline{W}(x) \leq \bar{W}(x)$$

For a proof of this comparison result between viscosity sub- and super-solutions see Barles (1994) and Barles and Perthame (1998, 1990) or for slightly different hypothesis Fleming and Soner (1993).

A.5. Proof of Theorem 5

Proof: From Lemma 1, the largest limit function V_{sup} and the smallest limit function V_{inf} are respectively viscosity sub-solution and super-solution of the HJB equation. From the comparison result of Lemma 2, $V_{\text{sup}} \leq V_{\text{inf}}$. But by their definition $V_{\text{sup}} \geq V_{\text{inf}}$, thus $V_{\text{sup}} = V_{\text{inf}} = V$ and the approximation scheme V^δ converges to the limit function V, which is the viscosity solution of the HJB equation thus the value function of the problem, and (32) holds true. □

Appendix B: Proof of Theorem 6

B.1. Outline of the proof

We know that from the convergence of the scheme V^δ (Theorem 5), for any compact $\Omega \subset O$, for any $\varepsilon_1 > 0$, there exists a discretization step δ such that:

$$\sup_{x \in \Omega} |V^\delta(x) - V(x)| \leq \varepsilon_1.$$

Let us define:

$$E_n^\delta = \sup_{\xi \in \Sigma^\delta \cup \partial\Sigma^\delta} \left| V_n^\delta(\xi) - V^\delta(\xi) \right|$$

As we have seen in Section 5.1.1, if we had the strong contraction property (36), then for any δ, E_n^δ would converge to 0 as $n \to \infty$. As we only have the weak contraction property (34):

$$\left| V_{n+1}^\delta(\xi) - V^\delta(\xi) \right| \leq (1 - k \cdot \delta) E_n^\delta + e(\delta) \cdot \delta$$

the idea of the following proof is that for any $\varepsilon_2 > 0$, there exists δ and a stage N, such that for $n \geq N$,

$$E_n^\delta = \sup_{\xi \in \Sigma^\delta \cup \partial\Sigma^\delta} \left| V_n^\delta(\xi) - V^\delta(\xi) \right| \leq \varepsilon_2. \tag{B.1}$$

Then we deduce that for any $\varepsilon > 0$, we can find $\varepsilon_1 > 0$ and $\varepsilon_2 > 0$ such that $\varepsilon_1 + \varepsilon_2 = \varepsilon$ and:

$$\begin{aligned}\sup_{\xi\in\Omega\cap(\Sigma^\delta\cup\partial\Sigma^\delta)} \left|V_n^\delta(\xi) - V(\xi)\right| &\leq \sup_{x\in\Omega} |V^\delta(x) - V(x)| + \sup_{\xi\in\Sigma^\delta\cup\partial\Sigma^\delta} \left|V_n^\delta(\xi) - V^\delta(\xi)\right| \\ &\leq \varepsilon_1 + \varepsilon_2 = \varepsilon\end{aligned}$$

B.2. A sufficient condition for $E_n^\delta \leq \varepsilon_2$

Lemma 3. *Let us suppose that there exists some constant $\alpha > 0$ such that for any state ξ updated at stage n, the following condition hold*:

$$\textit{If } E_n^\delta > \varepsilon_2 \textit{ then } \left|V_{n+1}^\delta(\xi) - V^\delta(\xi)\right| \leq E_n^\delta - \alpha \tag{B.2}$$

$$\textit{If } E_n^\delta \leq \varepsilon_2 \textit{ then } \left|V_{n+1}^\delta(\xi) - V^\delta(\xi)\right| \leq \varepsilon_2 \tag{B.3}$$

then there exists N such that for $n \geq N$, $E_n^\delta \leq \varepsilon_2$.

Proof: As the algorithm updates every state $\xi \in \Sigma^\delta$ regularly, there exists an integer m such that at stage $n+m$ all the states $\xi \in \Sigma^\delta$ have been updated at least once since stage n. Thus, from (B.2) and (B.3) we have:

$$\text{If } E_n^\delta > \varepsilon_2 \text{ then } \sup_{\xi\in\Sigma^\delta} \left|V_{n+m}^\delta(\xi) - V^\delta(\xi)\right| \leq E_n^\delta - \alpha$$

$$\text{If } E_n^\delta \leq \varepsilon_2 \text{ then } \sup_{\xi\in\Sigma^\delta} \left|V_{n+m}^\delta(\xi) - V^\delta(\xi)\right| \leq \varepsilon_2$$

Thus, there exists N_1 such that: $\forall n \geq N_1$,

$$\sup_{\xi\in\Sigma^\delta} \left|V_n^\delta(\xi) - V^\delta(\xi)\right| \leq \varepsilon_2. \tag{B.4}$$

Moreover, all states $\xi \in \partial\Sigma^\delta$ are updated at least once, thus there exists N_2 such that: $\forall n \geq N_2$, for all $\xi \in \partial\Sigma^\delta$,

$$\left|V_{n+1}^\delta(\xi) - V^\delta(\xi)\right| \leq k_R \cdot \delta \leq \varepsilon_2 \tag{B.5}$$

for any $\delta \leq \Delta_1 = \frac{\varepsilon_2}{k_R}$.

Thus from (B.4) and (B.5), for $n \geq N = \max\{N_1, N_2\}$,

$$E_n^\delta = \sup_{\xi\in\Sigma^\delta\cup\partial\Sigma^\delta} \left|V_n^\delta(\xi) - V^\delta(\xi)\right| \leq \varepsilon_2.$$

□

Lemma 4. *For any $\varepsilon_1 > 0$, there exists Δ_2 such that for $\delta \leq \Delta_1$, the conditions* (*B*.2) *and* (*B*.3) *are satisfied.*

Proof: Let us consider a value $\varepsilon_2 > 0$. From the convergence of $e(\delta)$ to 0 when $\delta \downarrow 0$, there exists Δ_1 such that for $\delta \leq \Delta_1$, we have:

$$e(\delta) - k \cdot \frac{\varepsilon_2}{2} \leq 0. \tag{B.6}$$

Let us prove that (B.2) and (B.3) hold. Let $E_n^\delta > \varepsilon_2$, then from (34),

$$\left|V_{n+1}^\delta(\xi) - V^\delta(\xi)\right| \leq (1 - k \cdot \delta)E_n^\delta + e(\delta) \cdot \delta \leq E_n^\delta - k \cdot \delta \cdot \varepsilon_2 + e(\delta) \cdot \delta$$

From (B.6),

$$\left|V_{n+1}^\delta(\xi) - V^\delta(\xi)\right| \leq E_n^\delta - k \cdot \delta \cdot \frac{\varepsilon_2}{2} + e(\delta) \cdot \delta - k \cdot \delta \cdot \frac{\varepsilon_2}{2} \leq E_n^\delta - k \cdot \delta \cdot \frac{\varepsilon_2}{2}$$

and (B.2) holds for $\alpha = k \cdot \delta \cdot \frac{\varepsilon_2}{2}$.

Now if $E_n^\delta \leq \varepsilon_2$, from (34), we have:

$$\left|V_{n+1}^\delta(\xi) - V^\delta(\xi)\right| \leq (1 - k \cdot \delta)\frac{\varepsilon_2}{2} + \frac{\varepsilon_2}{2} + e(\delta)\delta - k \cdot \delta\frac{\varepsilon_2}{2} \leq \frac{\varepsilon_2}{2} + \frac{\varepsilon_2}{2} = \varepsilon_2.$$

and condition (B.3) holds. □

B.3. Convergence of the algorithm

Proof: Let us prove Theorem 6. For any compact $\Omega \subset O$, for all $\varepsilon > 0$, let us consider $\varepsilon_1 > 0$ and $\varepsilon_2 > 0$ such that $\varepsilon_1 + \varepsilon_2 = \varepsilon$. From Lemma 4, for $\delta \leq \Delta_1$, conditions (B.2) and (B.3) are satisfied, and from Lemma 3, there exists N, for all $n \geq N$,

$$E_n^\delta = \sup_{\xi \in \Sigma^\delta \cup \partial\Sigma^\delta} \left|V_n^\delta(\xi) - V^\delta(\xi)\right| \leq \varepsilon_2.$$

Moreover, from the convergence of the approximation scheme, Theorem 5 implies that for any compact $\Omega \subset O$, there exists Δ_2 such that for all $\delta \leq \Delta_2$,

$$\sup_{x \in \Omega} |V^\delta(x) - V(x)| \leq \varepsilon_1$$

Thus for $\delta \leq \Delta = \min\{\Delta_1, \Delta_2\}$, for any finite discretized state-space Σ^δ and $\partial\Sigma^\delta$ satisfying the properties of Section 4.4, there exists N, for all $n \geq N$,

$$\begin{aligned}\sup_{\xi \in \Omega \cap (\Sigma^\delta \cup \partial\Sigma^\delta)} \left|V_n^\delta(\xi) - V(\xi)\right| &\leq \sup_{x \in \Omega} |V^\delta(x) - V(x)| + \sup_{\xi \in \Sigma^\delta \cup \partial\Sigma^\delta} \left|V_n^\delta(\xi) - V^\delta(\xi)\right| \\ &\leq \varepsilon_1 + \varepsilon_2 = \varepsilon.\end{aligned}$$

□

Acknowledgments

This work has been funded by DASSAULT-AVIATION, France, and has been carried out at the *Laboratory of Engineering for Complex Systems* (LISC) of the CEMAGREF, France. I would like to thank Paul Bourgine, Martine Naillon, Guillaume Deffuant, Guy Barles and Andrew Moore.

I am also very grateful to my parents, my mentor Daisaku Ikeda and my best friend Guylène.

References

Akian, M. (1990). Méthodes multigrilles en contrôle stochastique. Ph.D. Thesis, University Paris IX Dauphine.

Baird, L. C. (1995). Residual algorithms: Reinforcement learning with function approximation. In *Machine Learning: Proceedings of the Twelfth International Conference*.

Barles, G. (1994). *Solutions de viscosité des équations de Hamilton-Jacobi*, Springer-Verlag. Mathématiques et Applications, Vol. 17.

Barles, G. & Perthame, B. (1988). Exit time problems in optimal control and vanishing viscosity solutions of hamilton-jacobi equations. *SIAM Control Optimization*, *26*, 1133–1148.

Barles, G. & Perthame, B. (1990). Comparison principle for dirichlet-type hamilton-jacobi equations and singular perturbations of degenerated elliptic equations. *Applied Mathematics and Optimization*, *21*, 21–44.

Barles, G. & Souganidis, P. (1991). Convergence of approximation schemes for fully nonlinear second order equations. *Asymptotic Analysis*, *4*, 271–283.

Barto, A. G. (1990). Connectionist learning for control: An overview. In W. T. Miller, R. S. Sutton, & P. J. Werbos (Eds.), *Neural Networks for Control* (pp. 5–58). Cambridge, Massachussetts: MIT Press.

Barto, A. G., Bradtke, S. J., & Singh, S. P. (1991). Real-time learning and control using asynchronous dynamic programming. Tech. Rep. 91-57, Computer Science Department, University of Massachusetts.

Barto, A. G., Sutton, R. S., & Anderson, C. W. (1983). Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE Transactions on Systems, Man and Sybernetics, 13*, 835–846.

Bellman, R. (1957). *Dynamic Programming*. Princeton Univ. Press.

Bersini, H. & Gorrini, V. (1997). A simplification of the back-propagation-through-time algorithm for optimal neurocontrol. *IEEE Transaction on Neural Networks*, *8*, 437–441.

Bertsekas, D. P. (1987). *Dynamic Programming: Deterministic and Stochastic Models*. Prentice Hall.

Bertsekas, D. P. & Tsitsiklis, J. (1996). *Neuro-Dynamic Programming*. Athena Scientific.

Boyan, J. & Moore, A. (1995). Generalization in reinforcement learning: Safely approximating the value function. *Advances in Neural Information Processing Systems*, *7*, 369–376.

Crandall, M., Ishii, H., & Lions, P. (1992). User's guide to viscosity solutions of second order partial differential equations. *Bulletin of the American Mathematical Society*, *27*(1), 1–67.

Crandall, M. & Lions, P. (1983). Viscosity solutions of hamilton-jacobi equations. *Trans. of the American Mathematical Society*, *277*, 1–42.

Doya, K. (1996). Temporal difference learning in continuous time and space. *Advances in Neural Information Processing Systems*, *8*, 1073–1079.

Dupuis, P. & James, M. R. (1998). Rates of convergence for approximation schemes in optimal control. *SIAM Journal Control and Optimization*, *360*(2).

Fleming, W. H. & Soner, H. M. (1993). *Controlled Markov Processes and Viscosity Solutions*. Springer-Verlag. Applications of Mathematics.

Glorennec, P. & Jouffe, L. (1997). Fuzzy q-learning. In *Sixth International Conference on Fuzzy Systems*.

Gordon, G. (1995). Stable function approximation in dynamic programming. In *International Conference on Machine Learning*.

Griebel, M. (1998). Adaptive sparse grid multilevel methods for elliptic pdes based on finite differences. In *Proceedings Large Scale Scientific Computations*. Notes on Numerical Fluid Mechanics: Computing, submitted.

Gullapalli, V. (1992). Reinforcement Learning and its application to control. Ph.D. Thesis, University of Massachussetts, Amherst.

Harmon, M. E., Baird, L. C., & Klopf, A. H. (1996). Reinforcement learning applied to a differential game. *Adaptive Behavior*, *4*, 3–28.

Kaelbling, L. P., Littman, M., & Moore, A. W. (1996). Reinforcement learning: A survey. *Journal of AI Research*, *4*, 237–285.

Kushner, H. J. (1990). Numerical methods for stochastic control problems in continuous time. *SIAM J. Control and Optimization*, *28*, 999–1048.

Kushner, H. J. & Dupuis, P. (1992). *Numerical Methods for Stochastic Control Problems in Continuous Time*. Springer-Verlag. Applications of Mathematics.

Lin, L.-J. (1993). Reinforcement learning for robots using neural networks. Ph.D. Thesis, Carnegie Mellon University, Pittsburg, Pennsylvania.

Mahadevan, S. & Connell, J. (1992). Automatic programming of behavior-based robots using reinforcement learning. *Artificial Intelligence*, *5*, 311–365.

Meuleau, N. (1996). Le dilemme Exploration/Exploitation dans les systèmes d'apprentissage par renforcement. Ph.D. Thesis, Université de Caen.

Moore, A. W. (1991). Variable resolution dynamic programming: Efficiently learning action maps in multivariate real-valued state-spaces. In *Machine Learning: Poceedings of the Eight International Workshop* (pp. 333–337).

Moore, A. W. & Atkeson, C. (1995). The parti-game algorithm for variable resolution reinforcement learning in multidimensional state space. *Machine Learning Journal*, *21*.

Munos, R. (1996). A convergent reinforcement learning algorithm in the continuous case: The finite-element reinforcement learning. In *International Conference on Machine Learning*.

Munos, R. (1997a). Apprentissage par Renforcement, étude du cas continu. Ph.D. Thesis, Ecole des Hautes Etudes en Sciences Sociales.

Munos, R. (1997b). A convergent reinforcement learning algorithm in the continuous case based on a finite difference method. In *International Joint Conference on Artificial Intelligence*.

Munos, R. (1997c). Finite-element methods with local triangulation refinement for continuous reinforcement learning problems. In *European Conference on Machine Learning*.

Munos, R. (1998). A general convergence theorem for reinforcement learning in the continuous case. In *European Conference on Machine Learning*.

Munos, R., Baird, L., & Moore, A. (1999). Gradient descent approaches to neural-net-based solutions of the hamilton-jacobi-bellman equation. In *International Joint Conference on Neural Networks*.

Munos, R. & Bourgine, P. (1997). Reinforcement learning for continuous stochastic control problems. *Advances in Neural Information Processing Systems, 10*.

Munos, R. & Moore, A. (1998). Barycentric interpolators for continuous space and time reinforcement learning. *Advances in Neural Information Processing Systems, 11*, 1024–1030.

Munos, R. & Moore, A. (1999). Variable resolution discretization for high-accuracy solutions of optimal control problems. In *International Joint Conference on Artificial Intelligence*, 1348–1355.

Nowé, A. (1995). Fuzzy reinforcement learning an overview. *Advances in Fuzzy Theory and Technology*.

Pareigis, S. (1996). Multi-grid methods for reinforcement learning in controlled diffusion processes. *Advances in Neural Information Processing Systems, 9*.

Pareigis, S. (1997). Adaptive choice of grid and time in reinforcement learning. *Advances in Neural Information Processing Systems*, *10*.

Pontryagin, L., Boltyanskii, V., Gamkriledze, R., & Mischenko, E. (1962). *The Mathematical Theory of Optimal Processes*. New York: Interscience.

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

Singh, S. P., Jaakkola, T., & Jordan, M. I. (1994). Reinforcement learning with soft state aggregation. *Advances in Neural Information Processing Systems*, *6*, 359–368.

Sutton, R. S. (1996). Generalization in reinforcement learning: Successful examples using sparse coarse coding. *Advances in Neural Information Processing Systems*, *8*, 1038–1044.

Sutton, R. & Whitehead, S. (1993). Online learning with random representations. In *International Conference on Machine Learning*.

Williams, R. J. (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, *8*, 229–256.

Received April 25, 1997
Revised July 19, 1999
Final manuscript July 19, 1999