Open access • Journal Article • DOI:10.1111/1467-9965.T01-1-00002

First-Order Schemes in the Numerical Quantization Method — Source link

Vlad Bally, Gilles Pagès, Jacques Printems

Institutions: French Institute for Research in Computer Science and Automation

Published on: 01 Jan 2003 - Mathematical Finance (INRIA)

Topics: Quantization (signal processing), Piecewise and Conditional expectation

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

First order schemes in the numerical quantization method

V. Bally — G. Pagès — J. Printems

N° 4424

Avril 2002

THÈME 4

ISSN 0249-6399 ISRN INRIA/RR--4424--FR+ENG

INRIA
ROCQUENCOURT

First order schemes in the numerical quantization method

V. Bally* , G. Pagès† , J. Printems‡

Thème 4 — Simulation et optimisation
de systèmes complexes
Projet Mathfi

Rapport de recherche n° 4424 — Avril 2002 — 20 pages

Abstract: The numerical quantization method (see [B.P.1, B.P.2, B.P.P.1]) is a grid method which relies on the approximation of the solution of a nonlinear problem (e.g. backward Kolmogorov equation) by piecewise constant functions. Its purpose is to compute a large number of conditional expectations along the path of the associated diffusion process. We give here an improvement of this method by describing a first order scheme based on piecewise *linear* approximations. Main ingredients are correction terms in the transition probabilities weights. We emphasize the fact that in the case of optimal quantization, a non neglectable number of correction terms vanish. We think that this is a strong argument to use it. The problem of pricing and hedging American options is investigated and *a priori* estimates of the errors are established.

Key-words: american options, quantization, Malliavin calculus, numerical schemes, Monte Carlo method.

* Université du Mans et Projet MathFi INRIA, bally@ccr.jussieu.fr
† Université de Paris 6, U.M.R. CNRS 7599, gpa@ccr.jussieu.fr
‡ Université de Paris 12, U.M.R. CNRS 8050, printems@univ-paris12.fr

Unité de recherche INRIA Rocquencourt
Domaine de Voluceau, Rocquencourt, BP 105, 78153 Le Chesnay Cedex (France)
Téléphone : +33 1 39 63 55 11 —Télécopie : +33 1 39 63 53 30

Schémas numériques d'ordre 1 pour la méthode de quantification

Résumé : La méthode de quantification (voir [B.P.1, B.P.2, B.P.P.1]) numérique est une méthode de grille qui repose sur l'approximation de la solution d'un problème non linéaire (par exemple une équation de Kolmogorov rétrograde) par des fonctions cconstantes par morceaux. Son but est de calculer un grand nombre d'espérances conditionnelles le long des trajectoires du processus de diffusion associé. Nous présentons ici une amélioration de cette méthode en introduisant un schéma d'ordre un basé sur des approximations linéaires par morceaux. Les outils principaux sont des termes de correction dans les poids de transitions de probabilités. Nous portons l'attention sur le fait que dans le cas de la quantification optimale, un nombre non négligeable de termes correcteurs s'annulent. C'est un argument fort qui plaide en faveur de son utilisation. Le problème de pricing et de couverture d'options américaines est étudié de façon théorique (erreurs *a priori*) et numérique (par des simulations).

Mots-clés : options américaines, quantification, calcul de Malliavin, schémas numériques, méthode de Monte Carlo.

1 Introduction

The numerical quantization method has been introduced in $[B.P.1]$, $[B.P.2]$ and $[B.P.P.1]$. It is a grid method which is conceived in order to solve non linear problems in large dimension. Since the problems which we have in mind have a $P.D.E.$ formulation, analytical methods like finite differences or finite elements are candidates in order to solve such problems but it is well known that the implementation of this type of methods is rather difficult in dimension larger than three. So one would like to use some probabilistic methods of Monte Carlo type (which have the advantage of being dimension free). But this may not be done directly for non linear problems because the resolution of such problems suppose the computation of a large number of conditional expectations and not only of a single expectation. The numerical quantization method is in-between the analytical approach and the Monte Carlo method. One uses some grids and some weights (like in the finite element method) but the weights are computed using a Monte Carlo method. Although the error depends on the dimension as in the analytical methods, the advantage of using Monte Carlo is that one may implement such algorithms in dimension larger than 3 (typically up to 10 - beyond, the number of points needed in the grids becomes huge).

The aim of this paper is to give a more efficient version of this algorithm. Roughly speaking, in [B.P.1], [B.P.2], [B.P.P.1], [B.P.P.2], we have studied approximation schemes of order zero and now we give approximation schemes of order one. Basically the schemes of order zero use piecewise constant approximations of the functions at hand and consequently employ the information in one point only - the center of the cell on which the approximation of the function is constant. The schemes of order one use linear interpolation and so put to work several points - the center of the cell but also the centers of their neighbors. For example the basic finite element method represents an algorithm of order one because it employes linear interpolations and, if one uses polynomial interpolations then one obtains schemes of higher order (but of course the algorithm becomes much more complex). In our frame we use the Malliavin integration by parts formula in order to compute some correctors which produce piecewise linear interpolations. The attractive thing in our approach is that although the scheme become more complicated as we pass from a 0^{th}-order scheme to 1^{st}-order scheme, the complexity of the algorithm remains of the same order and the correctors which come on are of the same nature. Consequently they may be computed rather simply by the Monte Carlo method with the sample used to compute the weights coming on in the original 0^{th}-order method.

As emphasized in Section 3, there are two types of projection errors coming on in our algorithm, say "ceil" and "ground" errors. So, a priori we need two types of correctors concerning each of these errors. But it turns out that if we use optimal grids (in the quantization sense) the ground correctors vanish. This is an enlightening fact concerning optimal quantization and a strong argument to use it.

The numerical quantization represents a quite general approach to non linear problems because its main purpose is to compute a large number of conditional expectations along the path of a diffusion process (see Section 3.3). But our specific initial motivation comes from

pricing American options, which is an optimal stopping problem and so a typical non linear problem. In this paper we also focus on this problem in order to illustrate our method.

Finally, we mention that the first order correctors are closely related to the strategy, so we can produce as well some proxy of the strategy as a by-product of this first order scheme (see [B.P.P.2] for an extensive discussion of hedging by quantization) (see Fig. 1 at the end of this paper).

2 The basic algorithm for pricing American options

2.1 The problem

We consider a market model containing a risk-less asset S_t^0 and a $d-$dimensional risky asset $S_t \in R^d, 0 \leq t \leq T$, whose dynamics read

$$dS_t^0 = S_t^0 r dt, \quad S_0^0 = 1,$$

and

$$dS_t = \mathrm{Diag}(S_t)(r\mathbf{1}dt + \overline{\sigma}(t, S_t)dB_t), \quad S_0 = 1.$$

Here r is the interest rate, $\overline{\sigma}$ is the volatility function and $B_t = (B_t^1, \ldots, B_t^d)$ is a standard Brownian Motion on some probability space $(\Omega, \mathcal{F}, P)$ with the standard filtration $(\mathcal{F}_t)_{0 \leq t \leq T}$ associated to the Brownian motion. T is a fixed time (maturity). In order to avoid some rather complicated formulae in our computations it is convenient to work with $X_t := \log S_t$ instead of S_t itself. It is easy to see that the dynamics of X obey

$$dX_t = \sigma(t, X_t)dB_t + b(t, X_t)dt, \quad X_0 = x =: \log s_0$$

with $\sigma(t, x) = \overline{\sigma}(t, e^x), b(t, x) = r - \frac{1}{2}\mathrm{Tr}(\sigma\sigma^*)(t, e^x)$.

Moreover, we consider a payoff function $h : [0, T] \times R^d \to R$ and we want to price an American option of payoff h. The price at time $t \in [0, T]$ is given by

$$Y_t = \mathrm{esssup}_{\tau \in \mathcal{T}_{t,T}} E(h(\tau, X_\tau) \mid \mathcal{F}_t)$$

where $\mathcal{T}_{t,T}$ denotes the set of all the stopping times τ taking values in $[t, T]$. This is the Snell envelope of the semimartingale $h(t, S_t)$ (if h is sufficiently smooth). It is well known that no closd formula holds for Y so we will consider a discrete approximation. To process, we consider the Euler scheme of step $\frac{1}{n}$,

$$\overline{X}_{t_{k+1}} := \overline{X}_{t_k} + \sigma(t_k, \overline{X}_{t_k})\Delta_k + b(t_k, \overline{X}_{t_k})\frac{T}{n}, \quad \overline{X}_{0:} = x,$$

where $t_k = \frac{k}{n}$ and $\Delta_k = B(t_{k+1}) - B(t_k)$. In order to lighten the notation we put $X_k =: \overline{X}_{t_k}, \sigma_k(x) = \sigma(t_k, x), b_k(x) = b(t_k, x)$ so that we are concerned with the Markov chain $(X_k)_{k=0,\ldots,n}$ given by the induction

$$X_{k+1} = X_k + \sigma_k(X_k)\Delta_k + b_k(X_k)\frac{1}{n}, \quad X_0 = x.$$

Then the discrete version of Y is defined by $\overline{Y}_{t_k} = \text{esssup}_{\overline{\tau}\in\overline{\mathcal{T}}_{t_k,T}} E(h_k(X_{\overline{\tau}}) \mid \mathcal{F}_{t_k})$ where $h_k(x) = h(t_k, x)$ and $\overline{\mathcal{T}}_{t_k,T}$ denotes the set of all the discrete stopping times with respect to the filtration $(\mathcal{F}_{t_i})_{i=0,n}$, which take values in $\{k, ..., n\}$. We shall work under two different sets of hypothesis:

$$(H_1) \equiv \begin{array}{ll} (i) & \sigma \text{ and } h \text{ are Lipschitz continuous functions.} \\ (ii) & \sigma\sigma^* \geq c\, I_d \text{ where } I_d \text{ is the identity matrix.} \end{array}$$

or

$$(H_2) \equiv \begin{array}{ll} (i) & \sigma, b \text{ are are Lipschitz continuous and} \\ & h \text{ is Lipschitz continuous in } t \text{ and semiconvex in } x. \\ (ii) & \sigma\sigma^* \geq c\, I_d. \end{array}$$

The precise definition of semi-convex functions is given in $[B.P, 1]$ - we mention that convex functions are semi-convex along with twice differentiable functions whith bounded derivatives.

It is proved in [B.P.2] that under these hypothesis

$$\left(E(\max_{k\leq n} \left| Y_{t_k} - \overline{Y}_{t_k} \right|^2 \right)^{1/2} \leq \frac{C}{n^\alpha} \tag{1}$$

with $\alpha = \frac{1}{2}$ under (H_1) and $\alpha = 1$ under (H_2) (when using the true "sampled" diffusion $(X_{t_k})_{0\leq k\leq n}$ as the Markov chain instead of its Euler scheme).

Now we compute $\overline{Y}_{t_k}$ using the dynamical programing principle:

$$\begin{aligned} \overline{Y}_{t_n} &= h_n(X_n) \\ \overline{Y}_{t_k} &= \max\{h_k(X_k), E(\overline{Y}_{t_{k+1}} \mid \mathcal{F}_{t_k})\}. \end{aligned}$$

The analytical counterpart of this scheme is obtained in the following way. One constructs recursively the functions u_k by

$$\begin{aligned} u_n(x) &= h_n(x) \\ u_k(x) &= \max\{h_k(x), E(u_{k+1}(X_{k+1}) \mid X_k = x)\}. \end{aligned}$$

Then $\overline{Y}_{t_k} = u_k(X_k)$ and consequently, up to the approximation of Y by $\overline{Y}$, the price at time zero is given by $u_0(x) = u_0(\log s_0)$.

2.2 The basic algorithm

We want to produce an algorithm in order to compute u_k. The difficult point will be of course to design an efficient method to compute $E(u_{k+1}(X_{k+1}) \mid X_k = x), k = 0, ..., n$. It

is clear that we cannot do it for every point $x \in R^d$, so we will settle some space grids $\Gamma_k = \{x_k^1, \dots, x_k^{N_k}\} \subset R^d, k = 0, \dots, n$ corresponding to the epochs $t_k = \frac{kT}{n}$. The way we choose the size N_k of the grid and the location of the points x_k^i of the grid Γ_k play a crucial part and the numerical efficiency of the algorithm heavily depends on this choice. But these problems have already been extensively discussed in [B.P.1] and so we leave them out here. So, in this paper the grids $\Gamma_k, k = 0, \dots, n$ are some exogenously designed objects. Moreover we define the Voronoi tessel of x_k^i by

$$C_k^i := \{x \in R^d / \ |x - x_k^i| \leq \inf_{0 \leq j \leq N_k} \left|x - x_k^j\right|\}$$

and we denote by Π_k the projection on the grid Γ_k *i.e* $\Pi_k(x) =: \sum_{i=1}^{N_k} x_k^i 1_{C_k^i}(x)$. Note that $C_k^i, i = 1, \dots, n$ is not a partition of R^d because the different tessels have boundary hyperplanes in common. But this is just a formal disagreement because the laws of the random variables we work with are absolutely continuous and so weights no hyperplane. The basic idea is to approximate

$$E(u_{k+1}(X_{k+1}) \mid X_k = x_k^i) \approx E(u_{k+1}(\Pi_{k+1}(X_{k+1})) \mid \Pi_k(X_k) = x_k^i). \tag{2}$$

Note that

$$\begin{aligned} E(u_{k+1}(\Pi_{k+1}(X_{k+1})) \mid \Pi_k(X_k) = x_k^i) &= \sum_{j=1}^{N_{k+1}} u_{k+1}(x_{k+1}^j) E(1_{C_{k+1}^j}(X_{k+1})) \mid 1_{C_k^i}(X_k)) \\ &= \sum_{j=1}^{N_{k+1}} u_{k+1}(x_{k+1}^j) \pi_k^{ij} \\ \text{with} \qquad \pi_k^{ij} &:= \frac{P(X_{k+1} \in C_{k+1}^j, X_k \in C_k^i)}{P(X_k \in C_k^i)}. \end{aligned}$$

The π_k^{ij}'s are the weights in our algorithm and we compute them using the Monte Carlo method. The important point here is that we may compute all $\pi_k^{ij}, i = 1, \dots, N_k, j = 1, \dots, N_{k+1}, k = 0, \dots, n$ using the same sample $(X_k^1)_{k \leq n}, \dots, (X_k^M)_{k \leq n}$ of the chain $(X_k)_{k \leq n}$ (see [B.P.,1]). In fact

$$\pi_k^{ij} \sim \widehat{\pi}_k^{ij} =: \frac{\sum_{q=1}^M 1_{C_{k+1}^j}(X_{k+1}^q) 1_{C_k^i}(X_k^q)}{\sum_{q=1}^M 1_{C_k^i}(X_k^q)}. \tag{3}$$

So we avoid using different Monte Carlo procedures in order to compute the conditional expectation at each point, which would be extremely expensive. In this sense our algorithm may be seen as a *compressed* Monte Carlo Method. Now the algorithm reads

$$\begin{aligned} \widehat{u}_n(x_n^i) &= h_n(x_n^i), \quad i = 1, ..., N_n \\ \widehat{u}_k(x_k^i) &= \max\{h_k(x_k^i), \sum_{j=1}^{N_{k+1}} \pi_k^{ij} \widehat{u}_{k+1}(x_{k+1}^j)\}, \ k = 0, \dots, n. \end{aligned} \tag{4}$$

Of course in true applications we do not know π_k^{ij}, so we use $\widehat{\pi}_k^{ij}$. This introduces one further error – the statistical error– which is not discussed here (see $[B.P.2]$).

This is our basic algorithm. It is an algorithm of order zero because we replace $u_{k+1}(x)$ by $u_{k+1}(\Pi_{k+1}(x)) = \sum_{j=1}^{N_{k+1}} x_{k+1}^j 1_{C_{k+1}^j}(x)$ and X_k (with respect to which one takes conditional expectation) by $\Pi_k(X_k) = \sum_{i=1}^{N_k} x_k^i 1_{C_k^i}(X_k)$. So we work with piecewise constant functions.

2.3 Optimal grids and error estimates

We give now some error evaluations which are obtained in $[B.P, 2]$. First, one proves that

$$\left(E(\max_{k\leq n}|Y_{t_k} - \widehat{u}_k(\Pi_k(X_k))|^2\right)^{1/2} \leq \frac{C}{n^\alpha} + \sum_{k=0}^{n} C_k \left(E(|X_k - \Pi_k(X_k)|^2\right)^{1/2} \tag{5}$$

with $\alpha = \frac{1}{2}$ under (H_1) and $\alpha = 1$ under (H_2).

The grids we use are optimal in the following sense. One denotes defines the distortion of a grid $\Gamma := \{x^1, \ldots, x^{N_k}\}$ (with obvious notations) as

$$(D_{X_k}(\Gamma))^2 := E\,|X_k - \Pi_\Gamma(X_k)|^2 = \sum_{i=1}^{N_k} E((X_k - x^i)^2 1_{C_\Gamma^i}(X_k)). \tag{6}$$

A grid Γ_k is *optimal* if

$$D_{X_k}(\Gamma_k) = \inf_{\Gamma, |\Gamma|\leq N_k} D_{X_k}(\Gamma).$$

A basic result from the quantization theory (the Bucklew & Wise Theorem - see [G.L.],[P.], [B.P.1] for the precise result) asserts that, if the grid is optimal, then there exists a real constant C_2 such that

$$(D_{X_k}(\Gamma_k))^{1/2} = \left(E(|X_k - \Pi_{\Gamma_k}(X_k)|^2\right)^{1/2} \leq \frac{C_2}{N_k^{1/d}}. \tag{7}$$

Plugging (7) in (5) and the structure of the constants C_k in (5) make possible to tune the $N_k's$ in an optimal way - and this is done in [B.P.2]. Since in this paper we are simply interested in the asymptotic order, we leave out this tedious analysis and just assume that $N_k \leq N$, $k = 1, \ldots, n$ for some N and express the above error in terms of N.

Proposition 1 *Assume that (7) holds true and $N_k \leq N, k = 1, \ldots, n$. Then*

$$\left(E(\max_{k\leq n}|Y_{t_k} - \widehat{u}_k(\Pi_k(X_k))|^2\right)^{1/2} \leq C\left(\frac{1}{n^\alpha} + \frac{n}{N^{1/d}}\right) \tag{8}$$

with $\alpha = \frac{1}{2}$ under (H_1) and $\alpha = 1$ under (H_2).

Let us take one step beyond into the numerical properties of optimal grids. Since $\Gamma_k := \{x_k^1, \dots, x^{N_k}\}$ achieves the minimum, formal derivation in (6) (see [P.] for the complete argument) yields

$$\frac{\partial}{\partial x_k^i} \left(D_{X_k}(\Gamma_k)\right)^2 = 2E((X_k - x_k^i)1_{C_k^i}(X_k)) = 0. \tag{9}$$

We will show in the next section that optimal grids produce an error of order N^{-2} instead of N^{-1} - and the relation (9) represents the key argument: it says that, if the grid is optimal, then the terms of order one in a certain Taylor expansion of order two fade.

3 Correctors of order one

In the approximation presented in (2) there are two different projection errors corresponding to Π_k ("ceil") and Π_{k+1} ("ground"). The aim of this section is to produce some correctors which reduce these errors. In order to enlighten the notation we put $\widehat{X}_k = \Pi_k(X_k)$ and, for every integrable random variable Φ and every set $A \in \mathcal{F}_T$ we denote $E_A(\Phi) := E(\Phi \mid A)$.

3.1 The basic integration by parts formula

In our very elementary setting the Malliavin integration by parts formula reads as follows. Given two differentiable functions $f, g : R^d \to R$, and any real constant C,

$$E(\frac{\partial f}{\partial x_i}(\Delta_k)g(\Delta_k)) = -E((f(\Delta_k) + C)(\frac{\partial g}{\partial x_i}(\Delta_k) - \frac{n}{T}\Delta_k^i g(\Delta_k)). \tag{10}$$

The proof is obtained by a usual integration by parts(with respect to the gaussian distribution). It represents the starting point of the Malliavin calculus - which goes far away - but we stop here.

It seems natural to take $C = 0$ in the above formula because anyway $\nabla(f+C) = \nabla f$. But in our frame f appears as an a priori given function and the fact that we have the freedom of choosing any C is crucial for simulation. The practical way of using this formula is to employ the Monte Carlo method for computing the expectation in the right hand side in order to obtain the expectation in the left hand side. So we would like to simulate the expectation of some variable with a small variation and consequently we would choose $C = -E(f(\Delta_k))$ (if we know it) for example.

Now let $U : R^d \to R$ be a measurable function with polynomial growth (with $U(X_{k+1})$ integrable). We define

$$P_k U(x) := E(U(X_{k+1}) \mid X_k = x)$$

which represents the semigroup of the Euler Scheme. Our problem is to compute the derivatives of $P_k U$. We define $\theta_k(x, y) := x + \sigma_k(x)y + b_k(x)\frac{T}{n}$ so that $X_{k+1} = \theta_k(X_k, \Delta_k)$ and consequently $P_k U(x) = E(U(\theta_k(x, \Delta_k)))$. Moreover, since σ_k is invertible, we may define

$$\lambda_k(x, y) \quad = \quad : \sigma_k^{-1}(x) \times \nabla_x \theta_k(x, y) \quad and \tag{11}$$

$$\rho_k^l(x,y) \quad = \quad -\sum_{q=1}^{d}\left(\frac{\partial\lambda_k^{ql}}{\partial y_q}(x,y) - \frac{n}{T}y_q\lambda_k^{ql}(x,y)\right).$$

Lemma 1 *The partial derivatives of P_kU are given by*

$$\frac{\partial P_kU}{\partial x_l}(x) = E((U(\theta_k(x,\Delta_k)) - C(x))\rho_k^l(x,\Delta_k)),\ l=1,\dots,d. \tag{12}$$

where C is any real function.

Proof. One may assume w.l.g. that U is smooth. Note that $\nabla_y(U\circ\theta_k) = (\nabla U)\circ\theta_k \times \nabla_y\theta_k = (\nabla U)\circ\theta_k \times \sigma_k$ and so $(\nabla U)\circ\theta_k = \nabla_y(U\circ\theta_k)\times\sigma_k^{-1}$. It follows that

$$\nabla P_kU(x) = E((\nabla U)(\theta_k(x,\Delta_k))\nabla_x\theta_k(x,\Delta_k)) = E(\nabla_y(U\circ\theta_k)(x,\Delta_k)\lambda_k(x,\Delta_k))$$

and now (12) follows from (10) with $C = C(x)$. □

Remark 1 *Let us emphasize the simple but important example of constant volatility σ_k and constant drift coefficient b_k (the log-normal model corresponding to the classical Black-Sholes model). Then $\theta_k(x,y) = x + \sigma_k y + b_k\frac{T}{n}$, $\lambda_k(x,y) = \sigma_k^{-1}$ and so $\rho_k^l(x,y) = \frac{n}{T}(y\sigma_k^{-1})_l$.*

Finally we give a priori estimates which we use in order to evaluate errors.

Proposition 2 *Suppose that U is $[U]_1$- Lipschitz continuous. Then*

$$\left\|\frac{\partial P_kU}{\partial x_l}\right\|_\infty \le C[U]_1 \quad \textit{and} \quad \left\|\frac{\partial^2 P_kU}{\partial x_l\partial x_{l'}}\right\|_\infty \le C[U]_1\sqrt{n} \tag{13}$$

Moreover, if U is just a bonded measurable function

$$\left\|\frac{\partial P_kU}{\partial x_l}\right\|_\infty \le C\left\|U\right\|_\infty\sqrt{n} \quad \textit{and} \quad \left\|\frac{\partial^2 P_kU}{\partial x_l\partial x_{l'}}\right\|_\infty \le C\left\|U\right\|_\infty n. \tag{14}$$

Proof. The first inequality is obtained by direct calculation and the second one is obtained using integration by parts once. The proof in the case of a bounded function is done using integration by parts once (respectively twice) in order to obtain the first (respectively the second) inequality. □

3.2 The Π_k projection error (ground error)

We consider the same measurable function U as in the previous subsection and we want to approximate $P_kU(x_k^i)$. We define

$$\phi_U^k(x_k^i) = \frac{E(U(X_{k+1})1_{C_k^i}(X_k))}{P(X_k\in C_k^i)}$$

and we note that, if $U(x) = 1_{C_{k+1}^j}(x)$ then $\phi_U^k(x_k^i) = \pi_k^{ij}$, *i.e.* the standard weight we use in our algorithm. We write

$$E(U(X_{k+1})1_{C_k^i}(X_k)) = E(P_kU(X_k)1_{C_k^i}(X_k)) = P_kU(x_k^i)P(X_k \in C_k^i) + \varepsilon_k^i$$

with

$$\sum_{i=1}^{N_k} \left|\varepsilon_k^i\right| \leq [P_kU]_1 E\left|X_k - \Pi_k(X_k)\right| \leq \frac{C[P_kU]_1}{N^{1/d}}$$

the last inequality being a consequence of (7). In particular the above relation gives $P_kU(x_k^i) = \phi_U^k(x_k^i) - \varepsilon_k^i / P(X_k \in C_k^i)$ and so

$$\begin{aligned} E\left|P_kU(\widehat{X}_k) - \phi_k^U(\widehat{X}_k)\right| &= \sum_{i=1}^{N_k} E(\left|P_kU(\widehat{X}_k) - \phi_k^U(\widehat{X}_k)\right| 1_{\{X_k \in C_k^i\}}) \\ &= \sum_{i=1}^{N_k} \left|P_kU(x_k^i) - \phi_k^U(x_k^i)\right| P(X_k \in C_k^i) \leq \frac{C[P_kU]_1}{N^{1/d}}. \end{aligned}$$

The aim of this section is to prove that, as a consequence of the optimality of the grid, the above error is of order $N^{2/d}$ instead of $N^{1/d}$.

Proposition 3 *Assume that U is Lipschitz continuous and that the grid Γ_k is optimal, so that (9) holds true. Then*

$$E\left|P_kU(\widehat{X}_k) - \phi_k^U(\widehat{X}_k)\right| \leq \frac{C[U]_1\sqrt{n}}{N^{2/d}}. \tag{15}$$

Proof. We use a Taylor expansion

$$\begin{aligned} E(U(X_{k+1})1_{C_k^i}(X_k)) &= E(P_kU(X_k)1_{C_k^i}(X_k)) = P_kU(x_k^i)P(X_k \in C_k^i) \\ &\quad + \sum_{l=1}^{d} \frac{\partial P_kU}{\partial x_l}(x_k^i) E((X_k - x_k^i)^l 1_{C_k^i}(X_k)) + R_k^i \end{aligned}$$

with (see (13))

$$\sum_{i=1}^{N_k} \left|R_k^i\right| \leq \left\|\partial^2 P_kU\right\|_\infty E\left|X_k - \Pi_k(X_k)\right|^2 \leq \frac{C[U]_1\sqrt{n}}{N^{2/d}}.$$

Since the grid is optimal, $E((X_k - x_k^i)^l 1_{C_k^i}(X_k)) = 0$ and the proof is complete. □

3.3 The Π_{k+1} projection error

The aim of this section is to compute $\phi_k^U(x_k^i)$. We cannot solve our problem for a general measurable function U so we need that U has the special form

$$U(x) = P_{k+1}V(x) = E(V(X_{k+2}) \mid X_{k+1} = x).$$

Moreover, in order to obtain reasonable error evaluations, we assume that $V = P_{k+2}W$ for some bounded measurable function W. This may be seen as a regularity property for V, in particular V is Lipschitz continuous. Assume that W is Lipschitz continuous as well. Then, as a consequence of (13),

$$\left\| \frac{\partial^2 U}{\partial x_l \partial x_{l'}} \right\|_\infty \leq C[V]_1\sqrt{n} \quad \text{and} \quad \left\| \frac{\partial V}{\partial x_l} \right\|_\infty \leq C[W]_1. \tag{16}$$

We define the new weights by seeting

$$\pi_k^{l,ijr} = \frac{E(\rho_{k+1}^l(X_{k+1}, \Delta_{k+1})(X_{k+1} - x_{k+1}^j)^l 1_{C_k^i \times C_{k+1}^j \times C_{k+2}^r}(X_k, X_{k+1}, X_{k+2}))}{P(X_k \in C_k^i)}, \tag{17}$$

$$l = 1, \ldots, d, i = 1, \ldots, N_k, j = 1, \ldots, N_{k+1}, r = 1, \ldots, N_{k+2}$$

and

$$P_{V,k}U(x_k^i) =: \sum_{j=1}^{N_{k+1}} U(x_{k+1}^j)\pi_k^{ij} + \sum_{l=1}^{d}\sum_{j=1}^{N_{k+1}}\sum_{r=1}^{N_{k+2}} (V(x_{k+2}^r) - C_{V,l}^j)\pi_k^{l,ijr} \tag{18}$$

where $C_{V,l}^j$ are arbitrary constants which have to be settled in order to reduce the variance in our algorithm.

Lemma 2 *Suppose that $U = P_kV$ with $V = P_{k+2}W$ for some Lipschitz continuous function W. Then*

$$E\left|P_kU(\widehat{X}_k) - P_{V,k}U(\widehat{X}_k)\right| \leq \frac{C[W]_1\sqrt{n}}{N^{2/d}} \tag{19}$$

where C depends on the diffusion coefficients.

Proof. The idea is similar to that in the previous section: we localize on C_{k+1}^j and we employ a Taylor expansion

$$\begin{aligned}
\phi_k^U(x_k^i) \times P(X_k \in C_k^i) &= E(U(X_{k+1})1_{C_k^i}(X_k)) \\
&= \sum_{j=1}^{N_{k+1}} E(U(X_{k+1})1_{C_{k+1}^j}(X_{k+1})1_{C_k^i}(X_k)) \\
&= \sum_{j=1}^{N_{k+1}} U(x_{k+1}^j)E(1_{C_{k+1}^j}(X_{k+1})1_{C_k^i}(X_k)) + \\
&\quad + \sum_{j=1}^{N_{k+1}}\sum_{l=1}^{d} E(\frac{\partial U}{\partial x_l}(X_{k+1})(X_{k+1} - x_{k+1}^j)^l 1_{C_{k+1}^j}(X_{k+1})1_{C_k^i}(X_k)) + Q_k^i
\end{aligned}$$

with

$$Q_k^i = \sum_{j=1}^{N_{k+1}} \sum_{l,l'=1}^{d} E(\frac{\partial^2 U}{\partial x_l \partial x_{l'}}(X'_{k+1})(X_{k+1}-x_{k+1}^j)^l (X_{k+1}-x_{k+1}^j)^{l'} 1_{C_{k+1}^j}(X_{k+1}) 1_{C_k^i}(X_k)) +$$
$$\sum_{j=1}^{N_{k+1}} \sum_{l=1}^{d} E((\frac{\partial U}{\partial x_l}(x_{k+1}^j) - \frac{\partial U}{\partial x_l}(X_{k+1}))(X_{k+1}-x_{k+1}^j)^l 1_{C_{k+1}^j}(X_{k+1}) 1_{C_k^i}(X_k))$$

We use (16) and we obtain

$$\sum_{i=1}^{N_k} E\left|Q_k^i\right| \le C \left\|\partial^2 U\right\|_\infty E\left|X_{k+1}-\widehat{X}_{k+1}\right|^2 \le \frac{C[W]_1\sqrt{n}}{N^{2/d}}. \tag{20}$$

In order to compute $\frac{\partial U}{\partial x_l}(X_{k+1})$ we use (12) with $x = X_{k+1}$ and we obtain

$$\begin{aligned}\frac{\partial U}{\partial x_l}(X_{k+1}) &= E((V(\theta_{k+1}(X_{k+1},\Delta_{k+1})) - C_{V,l}^j)\rho_{k+1}^l(X_{k+1},\Delta_{k+1}) \mid X_{k+1}) \\ &= E((V(X_{k+2}) - C_{V,l}^j)\rho_{k+1}^l(X_{k+1},\Delta_{k+1}) \mid X_{k+1}).\end{aligned}$$

Moreover, using the Markov property first and then localization for X_{k+2} yield

$$\begin{aligned}&E(\frac{\partial U}{\partial x_l}(X_{k+1})(X_{k+1}-x_{k+1}^j)^l 1_{C_k^i \times C_{k+1}^j}(X_k, X_{k+1})) \\ =\ & E((V(X_{k+2}) - C_{V,l}^j)\rho_{k+1}^l(X_{k+1},\Delta_{k+1})(X_{k+1}-x_{k+1}^j)^l 1_{C_k^i \times C_{k+1}^j}(X_k, X_{k+1})) \\ =\ & \sum_{r=1}^{N_{k+2}} E((V(X_{k+2}) - C_{V,l}^j)\rho_{k+1}^l(X_{k+1},\Delta_{k+1})(X_{k+1}-x_{k+1}^j)^l 1_{C_k^i \times C_{k+1}^j \times C_{k+2}^r}(X_k, X_{k+1}, X_{k+2})) \\ =\ & \sum_{r=1}^{N_{k+2}} (V(x_{k+2}^r) - C_{V,l}^j) E(\rho_{k+1}^l(X_{k+1},\Delta_{k+1})(X_{k+1}-x_{k+1}^j)^l 1_{C_k^i \times C_{k+1}^j \times C_{k+2}^r}(X_k, X_{k+1}, X_{k+2})) \\ &+ H_k^{l,ij}.\end{aligned}$$

Recall that V is Lipschitz continuous . Recall also that $\left\|\rho_{k+1}^l(X_{k+1},\Delta_{k+1})\right\|_2 \le C\sqrt{n}$. Then

$$\begin{aligned}&\sum_{i=1}^{N_k} \sum_{j=1}^{N_{k+1}} \left|H_k^{l,ij}\right| \\ \le\ & C[V]_1\sqrt{n} \sum_{i=1}^{N_k} E(\left|X_{k+2}-\widehat{X}_{k+2}\right| \left|X_{k+1}-\widehat{X}_{k+1}\right| 1_{C_k^i}(X_k)) \\ \le\ & C\sqrt{n} E(\left|X_{k+2}-\widehat{X}_{k+2}\right|^2 + \left|X_{k+1}-\widehat{X}_{k+1}\right|^2)) \le \frac{C\sqrt{n}}{N^{2/d}}.\end{aligned}$$

Finally, using the result from the previous section

$$\begin{aligned}
& E\left|P_kU(\widehat{X}_k) - P_{V,k}U(\widehat{X}_k)\right| \\
\leq\ & \frac{C[W]_1\sqrt{n}}{N^{2/d}} + E\left|\phi_k^U(\widehat{X}_k) - P_{V,k}U(\widehat{X}_k)\right| \\
\leq\ & \frac{C[W]_1\sqrt{n}}{N^{2/d}} + \sum_{i=1}^{N_k}\left|\phi_k^U(x_k^i) - P_{V,k}U(x_k^i)\right| \times P(X_k \in C_k^i) \\
\leq\ & \frac{C[W]_1\sqrt{n}}{N^{2/d}} + \sum_{i=1}^{N_k} E\left|Q_k^i\right| + \sum_{l=1}^{d}\sum_{i=1}^{N_k}\sum_{j=1}^{N_{k+1}}\left|H_k^{l,ij}\right| \leq \frac{C[W]_1\sqrt{n}}{N^{2/d}}
\end{aligned}$$

and the proof is completed. □

In the algorithm we have in mind we want to compute P_kU but we do not have the exact value neither of U nor of V but just some approximations $U' : \Gamma_{k+1} \to R$ and $V' : \Gamma_{k+2} \to R$. So we are interested to evaluate the impact of the error $U - U'$ and $V - V'$. The functions U' and V' are not related by $U' = P_kV'$ and this relation makes actually no sense because these functions are only defined on some grids. Anyway, one may define $P_{V',k}U'(x_k^i)$ by (18).

Lemma 3 *Suppose that $\sqrt{n} \leq N^{1/d}$. For every $p > 1$ there exists an universal constant C_p such that for every $\varepsilon > 0$ and every $U, U' : \Gamma_{k+1} \to R, V, V' : \Gamma_{k+2} \to R$*

$$\begin{aligned}
& E\left|\left(P_{V,k}U - P_{V',k}U'\right)(\widehat{X}_k)\right| \leq E\left|(U - U')\,(\widehat{X}_{k+1})\right| \\
& + \frac{n^{\varepsilon}\sqrt{n}}{N^{1/d}} E\left|(V - V')(\widehat{X}_{k+2})\right| + \frac{C_p}{n^{p\varepsilon/8}}\left\|(V - V')(\widehat{X}_{k+2})\right\|_2 \qquad (21)
\end{aligned}$$

where C_p is a constant which depends on the coefficients of the diffusion process and on the constant in (7).

Proof. We assume that $C_{V,l}^j = C_{V',l}^j$ so that these terms disappear when taking the difference. Moreover, having in mind the expression of our weights we obtain

$$\begin{aligned}
& E\left|\left(P_{V,k}U - P_{V',k}U'\right)(\widehat{X}_k)\right| \\
=\ & \sum_{i=1}^{N_k}\left|\left(P_{V,k}U - P_{V',k}U'\right)(x_k^i)\right| P(X_k \in C_k^i) \\
\leq\ & E\Big(\Big|(U - U')(\widehat{X}_{k+1})\Big| + E\Big(\Big|(V - V')(\widehat{X}_{k+2})\Big| \sum_{l=1}^{d}\left|\rho_{k+1}^l(X_{k+1}, \Delta_{k+1})\right| \Big|X_{k+1} - \widehat{X}_{k+1}\Big|\Big) \\
=\ & : A + B.
\end{aligned}$$

Let us evaluate B. We denote $Q := \sum_{l=1}^{d} \left|\rho_{k+1}^{l}(X_{k+1}, \Delta_{k+1})\right| \left|X_{k+1} - \widehat{X}_{k+1}\right|$ and write

$$\begin{aligned} B &= E(\left|(V-V')(\widehat{X}_{k+2})\right| Q) \\ &\leq \frac{n^{\varepsilon}\sqrt{n}}{N^{1/d}} E\left|(V-V')(\widehat{X}_{k+2})\right| + \left\|(V-V')(\widehat{X}_{k+2})\right\|_2 \|Q\|_4 \, P(Q \geq \frac{n^{\varepsilon}\sqrt{n}}{N^{1/d}})^{1/4}. \end{aligned}$$

Note that $\left|\rho_k^l(x,y)\right| \leq C(1+\frac{n}{T}|y|\,(1+|y|))$. Then it is easy to see that $\|Q\|_4 \leq C\sqrt{n}/N^{1/d} \leq C$. Moreover, since $\sqrt{n}\Delta_{k+1}$ is standard normal distributed

$$\begin{aligned} P(Q &\geq \frac{n^{\varepsilon}\sqrt{n}}{N^{1/d}}) \leq Ce^{-C'n^{\varepsilon}} + P(\left|X_{k+1} - \widehat{X}_{k+1}\right| \geq \frac{n^{\varepsilon/2}}{N^{1/d}}) \\ &\leq Ce^{-C'n^{\varepsilon}} + P(\left|X_{k+1} - \widehat{X}_{k+1}\right| \geq \frac{n^{\varepsilon/2}}{N^{1/d}}) \\ &\leq Ce^{-C'n^{\varepsilon}} + \frac{C_p}{n^{\varepsilon p/2}} \end{aligned}$$

the last inequality being a consequence of (7). We may assume that the exponential term is dominated by $C_p/n^{\varepsilon p/2}$ and so the proof is completed. □

As an immediate consequence of the above lemmas we obtain

Theorem 1 *Let $U = P_k V$ with $V = P_{k+1}W$ where W is a Lipschitz continuous function on R^d and let $U' : \Gamma_{k+1} \to R$ and $V' : \Gamma_{k+2} \to R$ be any real functions. Suppose also that $\sqrt{n} \leq N^{1/d}$. Then, for every $p > 1, \varepsilon > 0$*

$$\begin{aligned} & E\left|(P_k U - P_{V',k}U')\,(\widehat{X}_k)\right| \qquad (22) \\ \leq\ & \frac{C[W]_1\sqrt{n}}{N^{2/d}} + E\left|(U-U')\,(\widehat{X}_{k+1})\right| + \frac{n^{\varepsilon}\sqrt{n}}{N^{1/d}} E\left|(V-V')\,(\widehat{X}_{k+2})\right| \\ & + \frac{C_p}{n^{p\varepsilon/8}} \left\|(V-V')\,(\widehat{X}_{k+2})\right\|_2 . \end{aligned}$$

where C_p is a constant which depends on the coefficients of the diffusion process and on the constant in (7).

3.4 The algorithm for the Snell envelope

In this section we give the analogues of the algorithm in (4):

$$\begin{aligned} \widehat{u}_n(x_n^i) &= h_n(x_n^i), \quad i = 1, ..., N_n \\ \widehat{P}_k \widehat{u}_{k+1} &= P_{\widehat{u}_{k+2},k}\widehat{u}_{k+1}. \\ \widehat{u}_k(x_k^i) &= \max\{h_k(x_k^i), \widehat{P}_k\widehat{u}_{k+1}(x_k^i)\}. \end{aligned}$$

Note that

$$\widehat{P}_k\widehat{u}_{k+1}(x_k^i) =: \sum_{j=1}^{N_{k+1}} u_{k+1}(x_{k+1}^j)\pi_k^{ij} + \sum_{l=1}^{d}\sum_{j=1}^{N_{k+1}}\sum_{r=1}^{N_{k+2}} (u_{k+1}(x_{k+2}^r) - C^j_{u_{k+1},l})\pi_k^{l,ijr}.$$

The constants $C^j_{u_{k+1},l}$ are to be chosen in order to reduce the variance.

The definition of $\pi_k^{l,ijr}$ makes not sense for $k = n-1$, so, in this case we do not use the corresponding corrector and we just put $\widehat{P}_{n-1}\widehat{u}_n(x_k^i) =: \sum_{j=1}^{N_n} h(x_n^j)\pi_{n-1}^{ij}$.

Now our problem is to evaluate the error given by our algorithm. A straightfoward argument shows that, if h has polynomial grows, then $\max_{k\le n}\left\|u_{k+1}(\widehat{X}_{k+1})\right\|_2 \le C < +\infty$. Then

$$\begin{aligned} & E\left|u_k(\widehat{X}_k) - \widehat{u}_k(\widehat{X}_k)\right| \\ \le\ & E\left|P_k u_{k+1}(\widehat{X}_k) - \widehat{P}_k\widehat{u}_{k+1}(\widehat{X}_k)\right| = E\left|P_k u_{k+1}(\widehat{X}_k) - P_{\widehat{u}_{k+2},k}\widehat{u}_{k+1}(\widehat{X}_k)\right| \\ \le\ & \frac{C\sqrt{n}}{N^{2/d}} + E\left|(u_{k+1} - \widehat{u}_{k+1})(\widehat{X}_{k+1})\right| + \frac{n^\varepsilon\sqrt{n}}{N^{1/d}} E\left|(u_{k+2} - \widehat{u}_{k+2})(\widehat{X}_{k+2})\right|) + \frac{C_p}{n^{p\varepsilon/8}}. \end{aligned}$$

We take now N suficiently large in order that $n^\varepsilon\sqrt{n}N^{-1/d} \le 1/n$. The iteration of the above inequality gives

$$E\left|u_k(\widehat{X}_k) - \widehat{u}_k(\widehat{X}_k)\right| \le n\left(\frac{C\sqrt{n}}{N^{2/d}} + \frac{C_p}{n^{p\varepsilon/8}}\right) \le \frac{Cn\sqrt{n}}{N^{2/d}} + \frac{C_p}{n}$$

the last inequality being true if $p \ge 16/\varepsilon$.

Remark 2 *Recall that $u_k = \max\{h_k, P_k u_{k+1}\} \ne P_k u_{k+1}$ so the above argument is not really rigorous. Some troubles may appear in the computation of $\partial P_k u_{k+1}$ when hitting the obstacle. It seems difficult to give a precise description of this error but we notice that this occurs rather seldom on one hand, and, on the other hand the derivatives appear in the correctors, so they are already multiplied by small quantities. So this does not seem to be a bad error. Anyway, numerical evidences show that this work well.*

Assume now that $n^\varepsilon\sqrt{n}N^{-1/d} \le 1/n$ and $p \ge 16/\varepsilon$ and let us see which is the impact of this inequality. Recall that the error which we compute is

$$|Y_0 - \widehat{u}_0(x_0)| \le \frac{C}{n^\alpha} + |u_0(x_0) - \widehat{u}_0(x_0)| \le \frac{C}{n^\alpha} + \frac{Cn\sqrt{n}}{N^{2/d}} + \frac{C_p}{n}$$

with $\alpha = 1$ for a semi-convex obstacle and $\alpha = \frac{1}{2}$ for a Lipschitz continuous obstacle. Our restriction on N ensures that $N^{1/d} \ge n^{\varepsilon+3/2}$ so that $n\sqrt{n}N^{-2/d} \le 1/n$. So we obtain $|Y_0 - \widehat{u}_0(x_0)| \le \frac{C}{n^\alpha} + \frac{C_p}{n}$. We have (almost) proved the following theorem:

Theorem 2 *Let us choose $\varepsilon > 0$ and $p \geq 16/\varepsilon$ and suppose that $N^{1/d} \geq n^{\varepsilon+3/2}$. Then*

$$|Y_0 - \widehat{u}_0(x_0)| \leq \frac{C_p}{n^\alpha}$$

where $\alpha = 1/2$ under (H_1) and $\alpha = 1$ under (H_2). C_p is a constant which depends on the coefficients of the diffusion process and on the constant in (7).

Remark 3 *Recall that if we use the basic algorithm we need $N = n^{2d}$ in order to obtain an error of order $1/n$ and $N = n^{3d/2}$ in order to obtain an error of order $1/\sqrt{n}$. So there is a gain at least if we want to obtain an error of order $1/n$. In fact the error evaluations we are able to obtain here are not very sharp and it seems clear that there is a gain in any case.*

3.5 Geometrical interpretation

We mentioned above that the 1−schemes corespond to the linear interpolation for the function u_{k+1}. The aim of this subsection is to make this assertion more precise. For simplicity we consider the one dimensional case only. So $x \in R$ and we denote $I_k^i = [x_k^i, x_k^{i+1})$. We also denote

$$u'^{,r}_{k+1}(x^j_{k+1}) := \frac{u_{k+1}(x^{j+1}_{k+1}) - u_{k+1}(x^j_{k+1})}{x^{j+1}_{k+1} - x^j_{k+1}} \quad u'^{,l}_{k+1}(x^j_{k+1}) := \frac{u_{k+1}(x^j_{k+1}) - u_{k+1}(x^{j-1}_{k+1})}{x^j_{k+1} - x^{j-1}_{k+1}}$$

and we think to $u'^{,r}_{k+1}(x^j_{k+1})$ (respectively to $u'^{,l}_{k+1}(x^j_{k+1})$) as an approximation of the right hand side (respectively of the left hand side) derivative of u_{k+1} in x^j_{k+1}. The linear interpolation for $\{u_{k+1}(x^j_{k+1}), j = 1, \ldots, N_{k+1}\}$ on $[x^1_{k+1}, x^{N_{k+1}}_{k+1}]$ is given by

$$\widetilde{u}_{k+1}(y) = \sum_{j=1}^{N_{k+1}} 1_{I^j_{k+1}}(y) \left(u_{k+1}(x^j_{k+1}) + u'^{,r}_{k+1}(x^j_{k+1})(y - x^j_{k+1}) \right)$$

In order to express this in terms of Voronoi tessels we denote $\overrightarrow{C}^j_{k+1} = [x^j_{k+1}, \frac{x^j_{k+1}+x^{j+1}_{k+1}}{2})$ and $\overleftarrow{C}^j_{k+1} = (\frac{x^{j-1}_{k+1}+x^j_{k+1}}{2}, x^j_{k+1})$ so that $C^j_{k+1} = \overleftarrow{C}^j_{k+1} \cup \overrightarrow{C}^j_{k+1}$ and $I^j_{k+1} = \overrightarrow{C}^j_{k+1} \cup \overleftarrow{C}^{j+1}_{k+1}$. With this notation

$$\begin{aligned} \widetilde{u}_{k+1}(y) &= \sum_{j=1}^{N_{k+1}} 1_{C^j_{k+1}}(y) u_{k+1}(x^j_{k+1}) \\ &\quad + \sum_{j=1}^{N_{k+1}} (1_{\overrightarrow{C}^j_{k+1}}(y) u'^{,r}_{k+1}(x^j_{k+1})(y - x^j_{k+1}) + 1_{\overleftarrow{C}^j_{k+1}}(y) u'^{,l}_{k+1}(x^j_{k+1})(x^j_{k+1} - y)). \end{aligned}$$

Suppose that we are far from the free boundary. Then using the above expression we obtain

$$u_k(x^i_k) = E_{\{B_{t_k} = x^i_k\}}(\widetilde{u}_{k+1}(B_{t_{k+1}}))$$

$$
\begin{aligned}
= & \sum_{j=1}^{N_{k+1}} u_{k+1}(x_{k+1}^j) E_{\{B_{t_k}=x_k^i\}}(1_{C_{k+1}^j}(B_{t_{k+1}})) \\
& + \sum_{j=1}^{N_{k+1}} u'^{,r}_{k+1}(x_{k+1}^j) E_{\{B_{t_k}=x_k^i\}}(1_{\overrightarrow{C}_{k+1}^j}(B_{t_{k+1}})(B_{t_{k+1}} - x_{k+1}^j)) \\
& + \sum_{j=1}^{N_{k+1}} u'^{,l}_{k+1}(x_{k+1}^j) E_{\{B_{t_k}=x_k^i\}}(1_{\overleftarrow{C}_{k+1}^j}(B_{t_{k+1}})(x_{k+1}^j - B_{t_{k+1}})).
\end{aligned}
$$

So the linear interpolation may be seen as a Taylor expansion of order one, with the derivative approximated by finite differences (in a different way in the left hand side and in the right hand side). This is exactly what we are doing in the 1− schemes. The only difference concerns the approximation that we use for the first order derivatives. The reason for which we do not use finite differences approximations is because this kind of scheme is not available in the multi-dimensional case, when the grid is not regular (we mean hypercubes): optimal grids are never regular.

Let us now come back to our way of computing derivatives. We stress that this is based on the fact that $u_{k+1}(x_{k+1}^j)$ is itself an expectation. The formula

$$
u'^{,l}_{k+1}(x_{k+1}^j) = E((u_{k+2}(\theta_k(x_{k+1}^j, \Delta_{k+1})) - C)\rho_k^l(x_{k+1}^j, \Delta_{k+1}))
$$

gives a pathways interpretation of the derivative and this is the basic fact which allows us to compute the derivatives using a Monte Carlo method.

There is one more difference between our method and the linear interpolation method. In the computation of $u'^{,l}_{k+1}(x_{k+1}^j)$ using the finite difference method one uses two values of u_{k+1} : at x_{k+1}^j and at x_{k+1}^{j+1} (respectively in x_{k+1}^{j-1}). In our method we do not use the values of u_{k+1} but of u_{k+2}. Moreover, we employ all the values $u_{k+2}(x_{k+2}^r), r = 1, ..., N_{k+2}$. Finally we stress that our interpolation is picewise linear but not continuous.

3.6 Numerical results

In this section we give some numerical results which illustrates our method and the contribution of the correctors. We will take for the obstacle

$$
h(t,x) = \left(\prod_{i=1}^{d/2} x_i - \prod_{i=d/2+1}^{d} x_i \right)_+ .
$$

This corresponds to an American style exchange option based on the d- dimensional asset $x = S_t = (S_t^1, \ldots, S_t^d)$. The fact that we take the special form $(\prod_{i=1}^{d/2} S_t^i - \prod_{i=d/2+1}^{d} S_t^i)_+$ for the payoff is motivated by the following technical reason. We have to compare our result with an (almost) perfectly computed price, and this is done in dimension 2 by Villeneuve & Zanetti in [V.Z.], using a finite difference method. So we use the fact that $\prod_{i=1}^{d/2} S_t^i$ (respectively

$\prod_{i=d/2+1}^{d} S_t^i$) has the same distribution as some standard one dimensional Black & Scholes diffusion $\widetilde{S}_t^1$ (respectively $\widetilde{S}_t^2$) with constant volatility. Note anyway that in our algorithm we simulate independently each $S_t^i, i = 1, \ldots, d$ so that our algorithm is a true d-dimensional one. We give the results for $d = 4, 6, 10$.

References

[B.P.P.1] V. Bally, G. Pagès, J. Printems (2001): A stochastic quantization method for non linear problems, *Monte Carlo Methods and Applications*, **7**, n° 1-2, pp.21-34.

[B.P.P.2] V. Bally, G. Pagès, J. Printems: (2002): A stochastic quantization method for non linear problem quantization tree method for pricing and hedging multi-dimensiuonal American options, Preprint de l'INRIA & de l'Université Paris VI.

[B.P.1] V. Bally, G. Pagès: (2000): A quantization algorithm for solving multi-dimensional optimal stopping problems, Pre-print n°628, Laboratoire de Probabilités &modèles aléatoires, Université Paris VI (France).

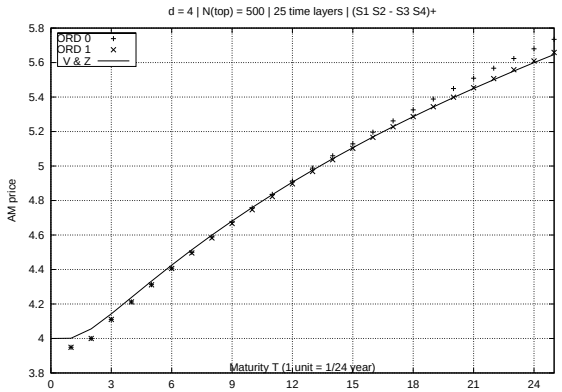
[B.P.2] V. Bally, G. Pagès (2001): Error analysis of the quantization algorithm for obstacle problems, Pre-print n°642, Laboratoire de Probabilités & modèles aléatoires, Université Paris VI (France).

[G.L.] S. Graf, H. Luschgy (2000): Foundations of quantization for probability distributions, Lecture Notes in Mathematics n°1730, Springer, 230p.

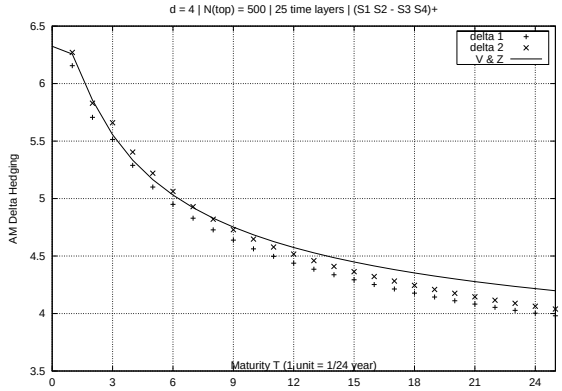
[P.] G. Pagès (1997): A space vector quantization method for numerical integration. Journal of Applied and Computational Mathematics, 89,pp.1-38.

[V.Z.] S. Villeneuve, A. Zanette (2002), Parabolic A.D.I. methods for pricing american option on two stocks, forthcoming in *Mathematics of Operation Research*.

a)



b)



c)

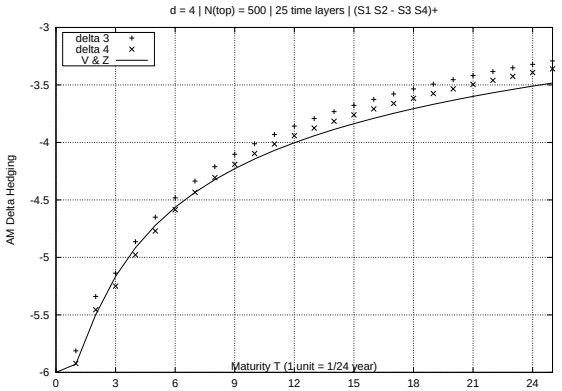
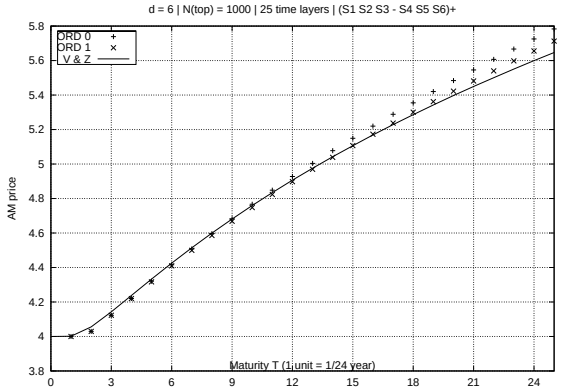
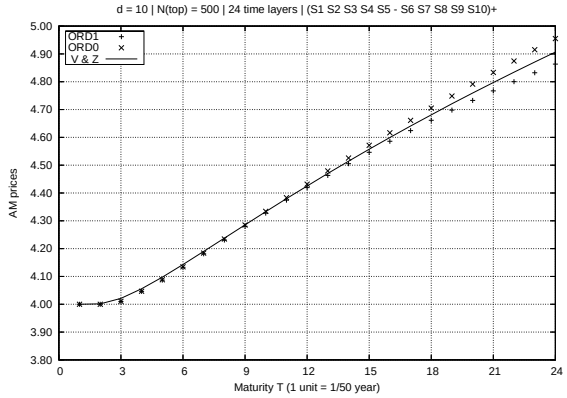


Figure 1: Dimension $d = 4$, $n = 25$ and $N_{25} = 500$, In-the-money case. a) Value of the American option function of the maturity T. The crosses denote the quantized version with order 0 (+) and order 1 (×). b) Hedging δ_1 and δ_2 function of maturity T computed with the first order scheme. c) Hedging δ_3 and δ_4 function of maturity T computed as the former. The reference price and hedging (V&Z) are denoted by solid lines.

a)



b)



c)

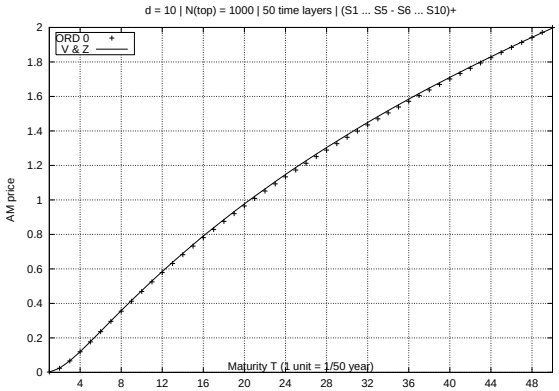


Figure 2: Crosses denote the quantized version with order 0 (+) and order 1 (×). a) Dimension $d = 6$, $n = 25$, $N_{25} = 1000$, In-the-money case. Value of the American option function of the maturity T. b) Dimension $d = 10$, $n = 50$, $N_{25} = 1000$, In-the-money case. Value of the American option function of the maturity T. c) Dimension $d = 10$, $n = 50$, $N_{25} = 1000$, Out-the-money case. Value of the American option function of the maturity T.

Unité de recherche INRIA Rocquencourt
Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38330 Montbonnot-St-Martin (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

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