HAL
open science

A non linear approximation method for solving high dimensional partial differential equations: Application in finance

José Infante Acevedo, Tony Lelièvre

To cite this version:

José Infante Acevedo, Tony Lelièvre. A non linear approximation method for solving high dimensional partial differential equations: Application in finance. Mathematics and Computers in Simulation, 2018, 143, pp.14-34. 10.1016/j.matcom.2016.07.013 . hal-00861892v2

HAL Id: hal-00861892

https://enpc.hal.science/hal-00861892v2

Submitted on 17 Sep 2013

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

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

A non linear approximation method for solving high dimensional partial differential equations: Application in Finance.

José Infante Acevedo and Tony Lelièvre
Université Paris-Est, CERMICS,
Ecole des Ponts, 6-8 avenue Blaise Pascal, 77455 Marne-la-vallée, France
jose-infante.acevedo@cermics.enpc.fr, lelievre@cermics.enpc.fr

September 17, 2013

Abstract: We study an algorithm which has been proposed in [2, 7] to solve high-dimensional partial differential equations. The idea is to represent the solution as a sum of tensor products and to compute iteratively the terms of this sum. This algorithm is related to the so-called greedy algorithm introduced by Temlyakov in [10]. In this paper, we investigate the application of the greedy algorithm in finance and more precisely to the option pricing problem. We approximate the solution to the Black-Scholes equation and we propose a variance reduction method. In numerical experiments, we obtain results for up to 10 underlyings. Besides, the proposed variance reduction method permits an important reduction of the variance in comparison with a classical Monte Carlo method.

Key words: Greedy algorithms, Black-Scholes partial differential equation

Acknowledgements: José Infante Acevedo is grateful to AXA Research Fund for his doctoral fellowship.

Introduction

Many problems of interest for various applications (for example material sciences and finance) involve high-dimensional partial differential equations (PDEs). The typical example in finance is the pricing of a basket option, which can be obtained by solving the Black-Scholes PDE with dimension the number of underlying assets.

We propose to investigate an algorithm which has been recently proposed by Chinesta *et al.* [2] for solving high-dimensional Fokker-Planck equations in the context of kinetic models for polymers, and by Nouy *et al.* [8] in uncertainty quantification framework based on previous works by Ladevèze [5]. This approach is also studied in [6] to try to circumvent the curse of dimensionality for the Poisson problem in high-dimension. This approach is a nonlinear approximation method called below the *greedy algorithm* because it is related to the so-called greedy algorithms introduced in nonlinear approximation theory, see for example [10]. The main idea is to represent the solution as a sum of tensor products (referred to as a *separated representation* in the following):

$$\begin{aligned} u(x_1,\dots,x_d) &= \sum_{k\geq 1} r_k^1(x_1) r_k^2(x_2)\dots r_k^d(x_d) \\ &= \sum_{k\geq 1} \left(r_k^1 \otimes r_k^2 \otimes \dots \otimes r_k^d\right)(x_1,\dots,x_d) \end{aligned} \tag{1}$$

and to compute iteratively each term of this sum using a greedy algorithm. This greedy algorithm can be applied to any PDE which admits a variational interpretation as a minimization problem. The practical interest of this algorithm has been demonstrated in various contexts (see for example [2] for applications in fluid mechanics).

Our contribution is to complete the first application of this algorithm in finance, investigating the interest of this approach for option pricing. In this work, our aim is to study the problem of pricing vanilla basket options of European type using two numerical methods: first a discretization technique for the Black-Scholes PDE and a variance reduction method for the pricing of the same type of financial products.

For option pricing, we will discuss in particular the key points to be solved to address problems in finance, compared to the situations studied in [6] or in [2], that is, the treatment of the non zero boundary conditions and the approximation of the solution to the Black-Scholes PDE as a sequence of minimization problems. We will study also the practical implementation of the algorithm. We will not solve the minimization problems associated to the PDE, but the first-order optimality conditions of these minimization problems, namely the Euler equation. This leads to a system of equations where the number of degrees of freedom does not grow exponentially with respect to the dimension, and this fact will be very important in order to attain high-dimensional frameworks in practical applications. More precisely, the Euler equation writes as a system of d nonlinear equations, where d is the considered dimension. The maximum dimension that can be treated by this technique is limited by the *non-linearity* of the system of d equations that has to be solved.

The variance reduction method relies on the backward Kolmogorov equation which yields an exact control variate. We propose to solve the high-dimensional Kolmogorov equation using the greedy algorithm. This yields an efficient pricing method which combines deterministic and stochastic techniques.

We would like to point out that in order to circumvent the curse of dimensionality using the greedy algorithm, the initial condition of the Black-Scholes equation has to be set out in separated representation with respect to the different coordinates, namely a sum of tensor products. As the initial condition is not always expressed in a separated representation, we first need to investigate the problem of approximating the initial condition by a sum of tensor products. This problem will be solved using again the greedy algorithm. We will provide examples to illustrate that this approximation is suitable.

Other deterministic techniques have been applied to solve the Black-Scholes PDE in a high-dimensional framework. Classical methods such as finite differences and finite elements are limited in their application when the dimension increases (typically $d \leq 4$), because the number of degrees of freedom increases exponentially with respect to the dimension and rapidly exceed the limited storage capacity. Financial applications of the sparse tensor product methods have been studied by Pommier in [9]. These sparse methods also use the representation of the solution as a sum of tensor products, and assume that the solution is regular enough to obviate fine discretizations in each direction. In practice, this method may be difficult to apply for reasons such as the lack of regularity of the solution and the difficulty to implement it.

In our numerical experiments, the greedy algorithm gives results for up to 10 underlyings; that means the dimension d is equal to 10. To the best of our knowledge, this is higher than results obtained using other deterministic approaches, such as the tensor product method, for which examples up to the dimension $d = 5$ have been reported in the literature, see [9]. In addition, the variance reduction method that we are proposing permits the variance to be reduced in comparison with a classic Monte Carlo method.

As future research, we have in mind to generalize the method to American options. This has a theoretical counterpart, namely the generalization of greedy algorithms to free boundary problems that has already been analyzed in [3].

The plan of the paper is the following. In Section 1, we introduce the general setting for the greedy algorithm and we give some theoretical results that have been proved in the literature and that ensure the convergence of the greedy algorithm. Section 2.1 will discuss the practical implementation of the greedy algorithm in the case of the approximation of a function by a sum of tensor products. Following this, in Section 2.2 and 2.3, we will present results and applications for the approximation of a basket put option. The purpose of Section 3 is to apply the greedy algorithm to solve the Black-Scholes PDE. After introducing the weak formulation of this equation in Section 3.1, we treat the difficulties that arise when applying the greedy algorithm such as posing the problem in a bounded domain (Section 3.2) and recasting the PDE as a minimization problem (Section 3.3). The final section (Section 4) contains numerical results for the solution of the Black-Scholes PDE and for the variance reduction method.

1 Greedy algorithms for high dimensional problems

In this section, we define a general framework for the greedy algorithm that we will use to solve the high-dimensional problems studied in this paper.

The bottom line of deterministic approaches for high-dimensional problems is to represent the solutions as linear combinations of tensor products of one-dimensional functions as in (1). If the number of terms in the expansion remains small, this enables us to approximate efficiently the solution, avoiding the curse of dimensionality.

The greedy algorithm proposed in [2, 6, 8] is based on two important points. The first one is that we need to recast the original problem (in our case, this is the option pricing problem) as a minimization problem:

$$u = \operatorname*{argmin}_{v \in V} \mathcal{E}(v), \tag{2}$$

where $\mathcal{E} : V \mapsto \mathbb{R}$ is a functional with a unique global minimizer $u \in V$ with V a Hilbert space. For example, for approximating the solution to a Poisson problem with homogeneous Dirichlet boundary conditions, one would consider $\mathcal{E}(v) = \frac{1}{2}\int_{\mathcal{X}} |\nabla v|^2 - \int_{\mathcal{X}} fv$ and $V = H_0^1(\mathcal{X}^d)$ with $\mathcal{X}$ a bounded one-dimensional domain and d large (see [6]).

The second point is to look iteratively for the best tensor product in the expansion of the solution as a sum of tensor products of lower-dimensional functions

$$u_n(x_1, x_2, \ldots, x_d) = \sum_{k=1}^{n} r_k^1 \otimes r_k^2 \ldots \otimes r_k^d(x_1, \ldots, x_d) \tag{3}$$

where for all $i = 1, \ldots, d$ and $k = 1, \ldots, n$, the functions $r_k^i \in V_{x_i}$, with V_{x_i} Hilbert spaces of functions depending on the one-dimensional variable x_i. This sequential search for the terms in the sum (3) is related to the so-called greedy algorithms introduced in the nonlinear approximation theory by Temlyakov in [10] and by De Vore and Temlyakov in [4].

In what follows, we assume that $V, V_{x_1}, V_{x_2}, \ldots, V_{x_d}$ are Hilbert spaces such that

(H1) $\mathrm{Vect}\{r^1 \otimes r^2 \otimes, \ldots \otimes r^d, r^1 \in V_{x_1}, r^2 \in V_{x_2}, \ldots, r^d \in V_{x_d}\} \subset V$ is dense.

To compute u_n in the separated form (3), u_n being the approximation of u solution of the problem (2), we define the greedy algorithm as follows,

Iterating for all $n \geq 1$:

$$(r_n^1, r_n^2, \ldots, r_n^d) \in \operatorname*{argmin}_{r^1 \in V_{x_1},\, r^2 \in V_{x_2}, \ldots, r^d \in V_{x_d}} \mathcal{E}\left(\sum_{k=1}^{n-1} r_k^1 \otimes r_k^2 \otimes \ldots \otimes r_k^d + r^1 \otimes r^2 \otimes \ldots \otimes r^d\right). \tag{4}$$

The following result given in [6] ensures the convergence of the greedy algorithm defined in (4) for a functional $\mathcal{E}$ of the following form

$$\mathcal{E}(v) = \|u - v\|_V^2. \tag{5}$$

Theorem 1.1. *Let us assume that the assumption (H1) holds and that the functional $\mathcal{E}$ has the form* (5)*. Then,*

$$\|u_n - u\|_V \underset{n \to \infty}{\longrightarrow} 0. \tag{6}$$

An estimate of the error is also proposed in [6]. To state this result, we need to introduce the functional space adapted to the convergence analysis

$$\mathcal{A}^1 = \left\{ u = \sum_{k=1}^{+\infty} r_k^1 \otimes r_k^2 \otimes \ldots \otimes r_k^d, \text{ where } r_k^i \in V_{x_i}, i = 1, \ldots, d, \sum_{k=1}^{+\infty} \|r_k^1 \otimes r_k^2 \ldots \otimes r_k^d\|_V < +\infty \right\} \tag{7}$$

and the associated norm which is

$$\|u\|_{\mathcal{A}^1} = \inf\left\{\sum_{k=1}^{+\infty} \|r_k^1 \otimes r_k^2 \ldots \otimes r_k^d\|_V \mid u = \sum_{k=1}^{+\infty} r_k^1 \otimes r_k^2 \otimes \ldots \otimes r_k^d\right\}. \tag{8}$$

Theorem 1.2. *Let us assume that the assumption (H1) is verified and that the functional $\mathcal{E}$ has the form* (5)*. Then, for a function $u \in \mathcal{A}^1$, there exists a constant $C > 0$ such that*

$$\|u_n - u\|_V \leq C n^{-1/6}, \tag{9}$$

for all $n \in \mathbb{N}^*$.

We note that the convergence rate factor of $\frac{1}{6}$ can be improved to $\frac{11}{62}$ and that the constant C depends on the norm $\|u\|_{\mathcal{A}^1}$, see [6, 4].

Our work relies on these theoretical results because in the setting of this paper, we will consider a functional $\mathcal{E}$ and a Hilbert space V such that $\mathcal{E}(v) = \|u - v\|_V^2$, (see equation (44) below).

2 Implementation of the algorithm in the case of the approximation of a square-integrable function

In this section, we will discuss the implementation of the algorithm defined by (4) in the case of the approximation of a given function f by a sum of tensor products. We will then provide numerical examples of this approach. This particular case has the advantage of being an easy example to understand the implementation of the greedy algorithm. Moreover, this procedure is useful in a preliminary step to approximate the initial condition of the Black-Scholes PDE (see Section 3), namely in order to get a separated representation of the payoff function.

2.1 Greedy algorithm for the approximation of a square-integrable function

In order to show the implementation that we use for the algorithm (4), let us present the simple problem of approximating a square-integrable function f by a sum of tensor products. Mathematically, we consider the spaces $V = L^2(\Omega_1 \times \Omega_2 \times \ldots \times \Omega_d)$, $V_{x_i} = L^2(\Omega_i)$ for $i = 1, \ldots, d$, where $\Omega_i \subset \mathbb{R}$ is a bounded domain for i such that $1 \leq i \leq d$. We recall that we are looking for a separated representation $f = \sum_{k \geq 1} r_k^1 \otimes r_k^2 \otimes \ldots \otimes r_k^d$. Let us consider the following minimization problem:

$$\text{Find } u \in V \text{ such that } u = \arg\min_{v \in V} \left(\frac{1}{2}\int_{\Omega_1 \times \Omega_2 \times \ldots \times \Omega_d} v^2 - \int_{\Omega_1 \times \Omega_2 \times \ldots \times \Omega_d} vf\right) \tag{10}$$

whose solution is obviously $u = f$. In this context, the greedy algorithm (4) can be rewritten as follows:

Iterate for all $n \geq 1$: Find $(r_n^1, r_n^2, \ldots, r_n^d) \in V_{x_1} \times V_{x_2} \times \ldots \times V_{x_d}$ such that $(r_n^1, r_n^2, \ldots, r_n^d)$ belongs to

$$\begin{aligned} \underset{r^1 \in V_{x_1}, \ldots, r^d \in V_{x_d}}{\operatorname{argmin}} \; & \frac{1}{2}\int_{\Omega_1 \times \Omega_2 \times \ldots \times \Omega_d} \left|\sum_{k=1}^{n-1} r_k^1 \otimes r_k^2 \otimes \ldots \otimes r_k^d + r^1 \otimes r^2 \otimes \ldots \otimes r^d\right|^2 \\ & - \int_{\Omega_1 \times \Omega_2 \times \ldots \times \Omega_d} \left(\sum_{k=1}^{n-1} r_k^1 \otimes r_k^2 \otimes \ldots r_k^d + r^1 \otimes r^2 \otimes \ldots \otimes r^d\right) f. \end{aligned} \tag{11}$$

As proposed in [6], instead of solving the problem (11), we will determine the solutions of the Euler equation for (11). Notice that, in general, the solutions of the Euler equation are not necessarily the solutions of the minimization problem, given the nonlinearity of the tensor product space $L^2(\Omega_1) \otimes L^2(\Omega_2) \otimes \ldots \otimes L^2(\Omega_d)$.

The Euler equation for (11) has the following form:

Find $(r_n^1, r_n^2, \ldots, r_n^d) \in V_{x_1} \times V_{x_2} \times \ldots \times V_{x_d}$ such that for any functions $(r^1, r^2, \ldots, r^d) \in V_{x_1} \times V_{x_2} \times \ldots \times V_{x_d}$

$$\int_{\Omega_1\times\Omega_2\times\ldots\times\Omega_d} (r_n^1\otimes r_n^2\otimes\ldots\otimes r_n^d)\left(r^1\otimes r_n^2\otimes\ldots\otimes r_n^d + r_n^1\otimes r^2\otimes\ldots\otimes r_n^d+\ldots+r_n^1\otimes r_n^2\otimes\ldots\otimes r^d\right)$$
$$= \int_{\Omega_1\times\Omega_2\times\ldots\times\Omega_d} f_{n-1}\left(r^1\otimes r_n^2\otimes\ldots\otimes r_n^d + r_n^1\otimes r^2\otimes\ldots\otimes r_n^d+\ldots+r_n^1\otimes r_n^2\otimes\ldots\otimes r^d\right) \tag{12}$$

where $f_{n-1} = f - \sum_{k=1}^{n-1} r_k^1\otimes r_k^2\otimes\ldots\otimes r_k^d$.

Equation (12) can be written equivalently as

$$\begin{cases} \|r_n^2\|^2\|r_n^3\|^2\ldots\|r_n^d\|^2\, r_n^1 = \int_{\Omega_2\times\Omega_3\times\ldots\times\Omega_d}\left(r_n^2\otimes\ldots\otimes r_n^d\right)\, f_{n-1}, \\ \|r_n^1\|^2\|r_n^3\|^2\|r_n^4\|^2\ldots\|r_n^d\|^2\, r_n^2 = \int_{\Omega_1\times\Omega_3\times\Omega_4\ldots\times\Omega_d}\left(r_n^1\otimes r_n^3\otimes r_n^4\otimes\ldots\otimes r_n^d\right)\, f_{n-1}, \\ \quad\vdots \\ \|r_n^1\|^2\|r_n^2\|^2\|r_n^3\|^2\ldots\|r_n^{d-1}\|^2\, r_n^d = \int_{\Omega_1\times\Omega_2\times\Omega_3\ldots\times\Omega_{d-1}}\left(r_n^1\otimes r_n^2\otimes r_n^3\otimes\ldots\otimes r_n^{d-1}\right)\, f_{n-1}, \end{cases} \tag{13}$$

where $\|r_n^i\|^2$ denotes the square L^2-norm: $\|r_n^i\|^2 = \int_{\Omega_i}|r_n^i|^2$.

The system (13) is a non linear coupled system of equations on which a fixed point procedure can be applied as proposed in [2]. Choose $(r_n^{1,(0)}, r_n^{2,(0)},\ldots,r^{d,(0)}) \in L^2(\Omega_1)\times L^2(\Omega_2)\times\ldots\times L^2(\Omega_d)$, and at iteration $k\geq 0$, compute $(r_n^{1,(k)}, r_n^{2,(k)},\ldots,r^{d,(k)}) \in L^2(\Omega_1)\times L^2(\Omega_2)\times\ldots\times L^2(\Omega_d)$ which is the solution to

$$\begin{cases} \|r_n^{2,(k)}\|^2\|r_n^{3,(k)}\|^2\ldots\|r_n^{d,(k)}\|^2\, r_n^{1,(k+1)} = \int_{\Omega_2\times\Omega_3\times\ldots\times\Omega_d}\left(r_n^{2,(k)}\otimes\ldots\otimes r_n^{d,(k)}\right)\, f_{n-1}, \\ \|r_n^{1,(k+1)}\|^2\|r_n^{3,(k)}\|^2\|r_n^{4,(k)}\|^2\ldots\|r_n^{d,(k)}\|^2\, r_n^{2,(k+1)} = \int_{\Omega_1\times\Omega_3\times\Omega_4\ldots\times\Omega_d}\left(r_n^{1,(k+1)}\otimes r_n^{3,(k)}\otimes r_n^{4,(k)}\otimes\ldots\otimes r_n^{d,(k)}\right)\, f_{n-1}, \\ \quad\vdots \\ \|r_n^{1,(k+1)}\|^2\|r_n^{2,(k+1)}|^2\|r_n^{3,(k+1)}\|^2\ldots\|r_n^{d-1,(k+1)}\|^2\, r_n^{d,(k+1)} \\ = \int_{\Omega_1\times\Omega_2\times\Omega_3\ldots\times\Omega_{d-1}}\left(r_n^{1,(k+1)}\otimes r_n^{2,(k+1)}\otimes r_n^{3,(k+1)}\otimes\ldots\otimes r_n^{d-1,(k+1)}\right)\, f_{n-1}, \end{cases} \tag{14}$$

until convergence is reached.

An important point to note is that we start with a linear problem (10) with exponential complexity with respect to the dimension, and at the end, we obtain a nonlinear problem (13) with at each iteration linear complexity with respect to the dimension. This is a general feature of the greedy algorithm: the curse of dimensionality is circumvented, but the linearity of the original problem is lost because the space of tensor products is non-linear.

In the two-dimensional case, the algorithm given by (11) is related to the Singular Value Decomposition (or rank one decomposition), as it is explained in [6]. In this case, the solutions of the variational problem (11) are exactly the solutions to the Euler equation (12) that verify the second-order optimality conditions. This property does not hold in a d-dimensional framework with $d\geq 3$.

2.2 Example of a separated representation of a put payoff

In this section we will apply the algorithm (11) to obtain an approximation of the payoff of a basket put option. For the practical implementation of the greedy algorithm, we need to introduce the space discretization. In practice, the spaces $V_{x_i}^{\Delta x}$ for $i=1,\ldots,d$ that are used to discretize $V_{x_i} = L^2(\Omega)$ with $\Omega_i = (0,1)$ for $i=1,\ldots d$ are the P1 finite elements on a uniform mesh with space step Δx. The integer $N = \frac{1}{\Delta x}$ is the number of intervals in each direction. For each k and $i=1,\ldots,d$, we discretize the functions r_k^i that appear in the approximation of the solution given by the expression (12) as follows:

$$r_k^{i,\Delta x}(x_i) = \sum_{j=0}^{N} r_k^{i,j}\phi_j(x_i), \quad r_k^{i,j}\in\mathbb{R},\ \forall j,k, \tag{15}$$

where $\phi_i(x) = \phi\left(\frac{x-x_i}{\Delta x}\right)$ with $\phi(x) = \begin{cases} 1-|x| & \text{if } |x| \leq 1, \\ 0 & \text{if } |x| > 0. \end{cases}$

This type of discretization and its generalization to the d-dimensional case will be used for all the numerical simulations of this paper.

Let us now consider the problem (11) with $f(x_1, \ldots, x_d) = \left(K - \frac{1}{d}\sum_{i=1}^{d} x_i\right)_+$. Figure 1 shows how the algorithm approximates the basket put payoff in a two-dimensional framework ($d = 2$). We observe that, as the number of iterations of the greedy algorithm increases, the approximation of the function $f(x_1, x_2)$ improves.

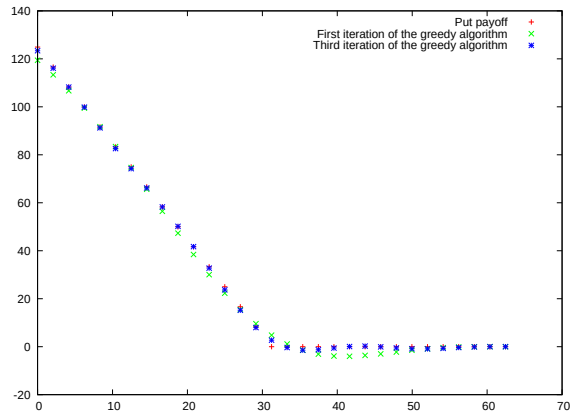

Figure 1: Basket put option with two assets. We consider here the intersection between the surface of prices and the plane $S_1 = S_2$. To obtain this approximation we take 31 points of discretization per dimension (N= 30). In this figure, we show the approximation given after the first and third iteration of the algorithm.

Figure 2 shows the convergence curves that we obtain for this problem as the dimension d increases. We observe that, as the dimension increases, the number of iterations needed to obtain the convergence increases as well.

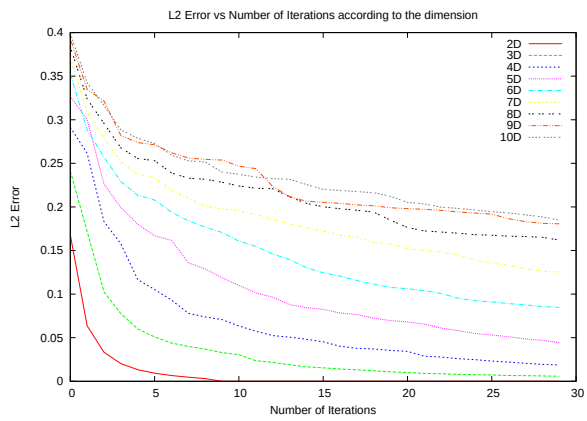

Figure 2: Convergence curves for the approximation of a basket put payoff by a sum of tensor products. We observe that if the dimension increases, then the number of iterations needed to obtain the convergence increases as well. The error calculated is given by the equation (16) below.

We also provide in Table 1 the number of iterations needed in order to obtain a relative error of 10^{-5} when we consider 11 points of discretization per dimension ($N = 10$). The relative error calculated is the discrete L^2 error

Dimension	Number of iterations
1	1
2	2
3	10
4	22
5	101
6	228
7	1077
8	3974

Table 1: Number of iterations needed to obtain a relative error of 10^{-5} when we take 11 discretization points per dimension

$$e_n = \frac{\sqrt{\frac{1}{N}\sum_{i_1=1}^{N}\sum_{i_2=1}^{N}\cdots\sum_{i_d=1}^{N}\left(f(x_{i_1},x_{i_2},\ldots,x_{i_d}) - u^n(x_{i_1},x_{i_2},\ldots,x_{i_d})\right)^2}}{\sqrt{\frac{1}{N}\sum_{i_1=1}^{N}\sum_{i_2=1}^{N}\cdots\sum_{i_d=1}^{N} f(x_1,\ldots,x_d)^2}} \tag{16}$$

where $u^n(x_1,x_2,\ldots,x_d) = \sum_{k=1}^{n} r_k^1 \otimes r_k^2 \otimes \ldots \otimes r_k^d$ is the solution obtained with the greedy algorithm at the iteration n. It is to be outlined that computing the norm is more costly than the greedy algorithm itself.

Notice that the full tensor product approximation would require $11^8 \simeq 2.10^8$ degrees of freedom in an 8-dimensional case to be compared with the $3974 \times 8 \times 11 \simeq 350000$ degrees of freedom that we obtained. For the evaluation of this number, we used the fact that at each iteration of the algorithm we get 8 functions that are determined by 11 discretization points.

In order to reduce the computational time of this calculation, we use the specific form of the payoff function to deduce, in a preliminary step, the points that belong to the support of this function. Therefore, when we calculate numerically the integral term

$$\int_{\Omega_2\times\Omega_3\times\ldots\times\Omega_d} \left(r_n^2 \otimes \ldots \otimes r_n^d\right)\ f\ dx_2 dx_3 \ldots dx_d \tag{17}$$

in (13), we do not need to pass through the points where the function f vanishes. In practice, we have used a backtracking algorithm to describe the support of the payoff function. This type of algorithm consists in constructing candidates sequentially and neglecting them when they do not verify the conditions required as a solution, in this case to belong to the support of the payoff function. For instance, in a 5-dimensional case, the computational time is reduced by a factor of $\frac{4}{5}$ by taking into account the support of the payoff function in the computations of the integral term.

Let us now make a few remarks.

Remark 2.1. *In our numerical experiments, the initial conditions needed to begin the iterations in the fixed point procedure are taken randomly. We indeed observed from numerical experiments that it yields better results in terms of convergence than in the case where constant initial conditions are used.*

Remark 2.2. *Concerning the computational time, we note that if the dimension increases, one iteration of the algorithm takes more time to be computed because the number of equations in the Euler system* (13) *increases linearly with respect to the dimension. The integral terms of type* (17) *also demand more time of execution because the domain has a new variable.*

2.3 Pricing of a basket put using the separated approximation of the payoff

As an example to show that the approximation by a sum of tensor products makes sense, we can use this approximation as a method to obtain prices of options in the Black-Scholes framework. The price of a European option in the Black-Scholes model (see Section 3.1) is given by

$$
\begin{aligned}
P_t &= \mathbb{E}\left[e^{-r(T-t)} f(S_1(T),\dots,S_d(T))|\mathcal{F}_t\right] \\
&= \int_{\Omega_1\times\dots\times\Omega_d} e^{-r(T-t)} f(y_1,\dots,y_d) g(T,y_1,\dots,y_d|t,S_1(t),\dots,S_d(t)) dy_1\dots dy_d
\end{aligned} \tag{18}
$$

where $(\mathcal{F}_t)_{t\geq 0}$ is the natural filtration generated by the d assets prices $S_i(t)$ $(i=1,\dots,d)$, f is the payoff of the option, $g(T,.|t,S_1(t),\dots,S_d(t))$ is the joint density of the variables $S_1(T),\dots,S_d(T)$ given the values $(S_1(t),\dots,S_d(t))$ of the underlying assets at time t. This joint density is a log-normal law and thus has an explicit analytical expression.

Using the greedy algorithm as we saw in the previous section, we can obtain a separable approximation of the product

$$
f(y_1,\dots,y_d) g(T,y_1,\dots,y_d|t,S_1(t),\dots,S_d(t)),
$$

and thus the integral (18) can be calculated very efficiently using the Fubini's rule.
In Figure 3, we apply this idea for the case of a basket put option on seven assets.

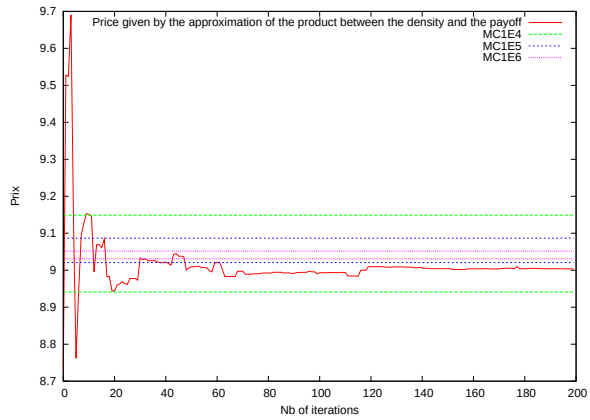

Figure 3: Price of put basket option with 7 assets. The continuous curve gives the price of this financial product with respect to the number of iterations of the algorithm. The horizontal lines represent the confidence interval obtained with a Monte Carlo method using respectively 10^4, 10^5 and 10^6 iterations.

3 Greedy algorithm for solving the Black-Scholes partial differential equation

Now, we will apply the greedy algorithm (4) presented in Section 1 to solve the Black-Scholes equation, and obtain the price of a European option.

3.1 Weak formulation of the Black-Scholes partial differential equation

The Black-Scholes model in a d-dimensional framework describes the dynamics of d risky assets that satisfy the following stochastic differential equations:

$$
\frac{dS_i(t)}{S_i(t)} = rdt + \sigma_i dB_i(t) \text{ for all } i=1,\dots,d, \tag{19}
$$

with

$$
d\langle B_i, B_j\rangle_t = \rho_{ij} dt. \tag{20}
$$

The number ρ_{ij} is the correlation between the Brownian motions B_i and B_j that drive the dynamics of the assets S_i and S_j respectively.

The coefficient σ_i represents the volatility of the asset S_i at time t, and r is the risk-free instantaneous interest rate. To simplify, we assume that r and σ_i for $i = 1, \dots d$ are constant during the period $[0, T]$. We note that the greedy algorithm that we are proposing can be used when the risk-free interest rate is a continuous function of time and the volatility is a continuous function of time and of the asset under standard regularity assumptions (See Chapter 2 in [1]).

We recall that the price of a European option with payoff f and maturity T is given by the following formula

$$\mathbb{E}\left[e^{r(T-t)} f(S_1(T), \dots, S_d(T)) | \mathcal{F}_t\right].$$

Using the Markovianity of the process $(S_1(t), \dots, S_d(t))$, it can be written as:

$$\mathbb{E}\left[e^{r(T-t)} f(S_1(T), \dots, S_d(T)) | \mathcal{F}_t\right] = P(t, S_1(t), \dots S_d(t)), \tag{21}$$

where P is a deterministic function.

The function $P(t, S_1, \dots, S_d)$ satisfies the Black-Scholes PDE which can be obtained using the Feynman-Kac theorem. The Black-Scholes equation in a d-dimensional framework is a parabolic PDE that has the following form:

$$\begin{cases} \frac{\partial P}{\partial t} + \mathcal{L}P = 0, \quad t < T, (S_1, \dots, S_d) \in \mathbb{R}^d_+, \\ P(T, S_1, \dots, S_d) = f(S_1, \dots, S_d), \quad (S_1, \dots, S_d) \in \mathbb{R}^d_+, \end{cases} \tag{22}$$

where the operator $\mathcal{L}$ is given by

$$\mathcal{L}P = \frac{1}{2} \sum_{i,j=1}^{d} \frac{\partial^2 P}{\partial S_i \partial S_j} \rho_{ij} \sigma_i \sigma_j S_i S_j + \sum_{i=1}^{d} r S_i \frac{\partial P}{\partial S_i} - rP.$$

Let us recall the standard framework for problem (22). Setting $\tau := T - t$, the time to maturity, we get the following forward parabolic problem for $\hat{P}(\tau, S_1 \dots, S_d) = P(t, S_1 \dots, S_d)$

$$\begin{cases} \frac{\partial \hat{P}}{\partial \tau} - \mathcal{L}\hat{P} = 0, \quad 0 < \tau \leq T, (S_1, \dots, S_d) \in \mathbb{R}^d_+ \\ \hat{P}(0, S_1, \dots, S_d) = f(S_1, \dots, S_d), \quad (S_1, \dots, S_d) \in \mathbb{R}^d_+. \end{cases} \tag{23}$$

We note that it is possible to write the diffusion term in the operator $\mathcal{L} + r$ in a divergence form as follows:

$$\mathcal{L}\hat{P} + r\hat{P} = \frac{1}{2} \sum_{i=1}^{d} \frac{\partial}{\partial S_i} \left(\sum_{j=1}^{d} \rho_{i,j} \sigma_i \sigma_j S_i S_j \frac{\partial \hat{P}}{\partial S_j} \right) + \sum_{j=1}^{d} \left(r S_j - \frac{1}{2} \sum_{i=1}^{d} \frac{\partial}{\partial S_i} (\rho_{i,j} \sigma_i \sigma_j S_i S_j) \right) \frac{\partial \hat{P}}{\partial S_j}.$$

Therefore, if we multiply $-\mathcal{L}\hat{P}$ by a test function Q and then we integrate on $\mathbb{R}^d_+$, we obtain the following bilinear form:

$$\begin{aligned} b_t(\hat{P}, Q) = & \frac{1}{2} \sum_{i=1}^{d} \sum_{j=1}^{d} \int_{\mathbb{R}^d_+} \rho_{i,j} \sigma_i \sigma_j S_i S_j \frac{\partial \hat{P}}{\partial S_j} \frac{\partial Q}{\partial S_i} \\ & - \sum_{j=1}^{d} \int_{\mathbb{R}^d_+} \left(r S_j - \frac{1}{2} \sum_{i=1}^{d} \frac{\partial}{\partial S_i} (\rho_{i,j} \sigma_i \sigma_j S_i S_j) \right) \frac{\partial \hat{P}}{\partial S_j} Q + r \int_{\mathbb{R}^d_+} \hat{P} Q. \end{aligned} \tag{24}$$

Let us now introduce the Hilbert space

$$\mathcal{V}(\mathbb{R}^d_+) = \left\{ v : v \in L^2(\mathbb{R}^d_+), S_i \frac{\partial v}{\partial S_i} \in L^2(\mathbb{R}^d_+), i = 1, \dots, d \right\}$$

and its norm

$$\|v\|_{\mathcal{V}} = \left(\|v\|^2_{L^2(\mathbb{R}^d_+)} + \sum_{i=1}^{d} \left\| S_i \frac{\partial v}{\partial S_i} \right\|^2_{L^2(\mathbb{R}^d_+)} \right)^{\frac{1}{2}}.$$

We have the following result for problem (23) (see Theorem 2.11 in [1]).

Theorem 3.1. *Let us assume that the matrix Ξ defined by $\Xi_{i,j} = \rho_{i,j}\sigma_i\sigma_j$ is positive-definite. Then for all $f \in L^2(\mathbb{R}^d_+)$, there exists a unique function $\hat{P} \in L^2(0,T;\mathcal{V}) \cap \mathcal{C}^0([0,T];L^2(\mathbb{R}^d_+))$, with $\frac{\partial \hat{P}}{\partial t} \in L^2(0,T;\mathcal{V}')$ such that, for any function $\phi \in \mathcal{D}(0,T)$, for all $v \in \mathcal{V}$,*

$$-\int_0^T \phi'(t) \left(\int_{\mathbb{R}^d_+} \hat{P}(t) v \right) dt + \int_0^T \phi(t) b_t(\hat{P}(t), v) dt = 0 \tag{25}$$

and

$$\hat{P}(t=0) = f. \tag{26}$$

This result shows the existence and uniqueness of a weak solution for the problem (23).

In this work, our goal is to obtain the curve of prices for a put basket option which has a square-integrable payoff. The price of call basket options can be obtained by the well-known put-call parity. So, as initial condition we consider the payoff function:

$$f(S_1, \ldots, S_d) = \left(K - \frac{1}{d} \sum_{i=1}^{d} S_i(0) \right)_+ \tag{27}$$

where the constant K is the strike of the option.

Three new difficulties appear when we want to apply the greedy algorithm (4) to solve the problem (23) when we compare it with the application of the greedy algorithm to the case of the Poisson problem studied in [6]:

1. It is a problem posed on an infinite domain.
2. It is a time-dependent problem.
3. We cannot simply recast the weak formulation (25) of the problem (23) as a minimization problem because the bilinear form (24) is non-symmetric.

3.2 Formulation on a bounded domain

The financial assets S_i for $i = 1, \ldots, d$ take values in $[0, \infty)$. Consequently, we have to deal with an infinite domain. Let us then introduce the following transformations:

$$\Psi : \mathbb{R}_+ \mapsto [0,1], \quad s \mapsto \frac{s}{s + \frac{K}{d}}, \tag{28}$$

$$\Psi^{-1} : [0,1] \mapsto \mathbb{R}_+, \quad x \mapsto \frac{xK}{d(1-x)}. \tag{29}$$

As remarked by Pommier in [9], the change of variables (28) maps bijectively $\mathbb{R}_+$ to the interval $[0,1]$ and appears to be efficient in practice since it leads to a refined mesh around the singularity line of the payoff function. In [9], Pommier explains that if we set a classical localized boundary-domain then the volume next to this singularity decays exponentially with the dimension. This change of variables allows us not to impose artificial boundary conditions contrary to classical truncation techniques. Proposition 3.1 below shows that with the change of variables (28), we get a well-posed problem in a bounded domain without boundary conditions.

Applying the change of variable (28) into the equation (22), we obtain:

$$\begin{cases} -\frac{\partial u}{\partial t} + \tilde{\mathcal{L}}u = 0, \\ u(0, x_1, \dots, x_d) = (K - \frac{K}{d}\sum_{i=1}^{d} \frac{x_i}{1-x_i}), \end{cases} \tag{30}$$

where $u(t, x_1, \dots, x_d) = P(t, S_1, \dots, S_d)$ with $S_i = \Psi(x_i)$,$(x_1, \dots, x_d) \in \Omega = (0,1)^d$ and

$$\tilde{\mathcal{L}}u = \operatorname{div}(A\nabla u) + \sum_{i=1}^{d} \left[r + \sigma_i^2 x_i - \sigma_i^2 + \frac{\sigma_i}{2} \sum_{j=1, j\neq i}^{d} \rho_{i,j}\sigma_j(2x_j - 1) \right] x_i(1-x_i)\frac{\partial u}{\partial x_i} - ru,$$

with the matrix A given by

$$A_{i,j}(x_1, \dots, x_d) := \frac{\rho_{i,j}\sigma_i\sigma_j}{2} x_i x_j (1-x_i)(1-x_j). \tag{31}$$

Then let us introduce the following Hilbert space

$$\tilde{\mathcal{V}}(\Omega) = \left\{ v \in L^2(\Omega) \mid \forall\, 1 \le i \le d,\ (1-x_i)x_i \frac{\partial v}{\partial x_i} \in L^2(\Omega) \right\}, \tag{32}$$

endowed with the norm

$$\|v\|_{\tilde{\mathcal{V}}} = \left(\|v\|^2_{L^2(\Omega)} + |v|^2_{\tilde{\mathcal{V}}} \right)^{\frac{1}{2}} \tag{33}$$

where

$$|v|^2_{\tilde{\mathcal{V}}} = \sum_{i=1}^{d} \left\| x_i(1-x_i)\frac{\partial v}{\partial x_i} \right\|^2_{L^2(\Omega)}. \tag{34}$$

In what follows, we need the following lemma:

Lemma 3.1. *The space $\mathcal{C}_c^\infty(\Omega)$ is dense in $\tilde{\mathcal{V}}(\Omega)$.*

This lemma can be directly deduced from Lemma 2.6 in [1].

Corollary 3.1. *The following integration by parts formula holds:*

$$\int_\Omega \operatorname{div}(A\nabla u)v = -\int_\Omega (A\nabla u)\nabla v, \quad \forall u, v \in \tilde{\mathcal{V}}(\Omega). \tag{35}$$

Proof. It follows from Lemma 3.1. □

Therefore, multiplying $-\tilde{\mathcal{L}}u$ by a test function $v \in \mathcal{C}_c^\infty(\Omega)$ and then using (35), we get the following bilinear form:

$$\tilde{b}_t(u, v) = \int_\Omega (A\nabla u)\nabla v - \int_\Omega (a\nabla u)v + \int_\Omega ruv. \tag{36}$$

where $a = (a_1, \dots, a_d) : \Omega \mapsto \mathbb{R}^d$ is the vector field with i-th component given by

$$a_i(x_1, \dots, x_d) = x_i(1-x_i)\left[r + \sigma_i^2 x_i - \sigma_i^2 + \frac{\sigma_i}{2} \sum_{j=1, j\neq i}^{d} \rho_{ij}\sigma_j(2x_j - 1) \right], \tag{37}$$

The following result holds for the the bilinear form defined in (36)

Lemma 3.2. *The bilinear form $\tilde{b}_t$ is continuous from $\tilde{\mathcal{V}} \times \tilde{\mathcal{V}}$, that is, there exists a constant $\overline{c}$ that does not depend on t such that for all functions $v, w \in \tilde{\mathcal{V}}$*

$$\tilde{b}_t(v, w) \le \overline{c}\|v\|_{\tilde{\mathcal{V}}}|w|_{\tilde{\mathcal{V}}} \tag{38}$$

Moreover, the bilinear form $\tilde{b}_t$ verifies a Garding inequality, that is, there exist two positive constants $\underline{c} > 0$ and $\lambda > 0$ such that for all functions $v \in \tilde{\mathcal{V}}$

$$b(v, v) \ge \underline{c}|v|^2_{\tilde{\mathcal{V}}} - \lambda\|v\|^2_{L^2(\Omega)} \tag{39}$$

Proof. The Garding inequality is obtained by observing that the first order term satisfies the following:

$$\left|\int_\Omega p(x_1,\dots,x_d)\frac{\partial v}{\partial x_i}v\right| \le \frac{1}{2}\left|\int_\Omega \frac{\partial p}{\partial x_i}(x_1,\dots,x_d)v^2\right|,$$

where $p(x_1,\dots,x_d)$ is a polynomial. The proof of the continuity uses the same arguments. □

Thus, we obtain the following result concerning the existence and uniqueness of the solution for the weak formulation associated to the problem (30).

Proposition 3.1. *For all function $g \in L^2(\Omega)$, there exists a unique $u \in L^2(0,T;\tilde{\mathcal{V}}) \cap \mathcal{C}^0([0,T];L^2(\Omega))$, with $\frac{\partial u}{\partial t} \in L^2(0,T;\tilde{\mathcal{V}}')$ such that for any function $\phi \in \mathcal{D}(0,T)$, for all function $v \in \tilde{\mathcal{V}}$,*

$$-\int_0^T \phi'(t)\int_\Omega u(t)v dt + \int_0^T \phi(t)\tilde{b}(u,v)dt = 0, \tag{40}$$

and

$$u(t=0) = g \tag{41}$$

Moreover, u solution of (40) *is related to $\hat{P}$ solution of* (25) *by the functions defined in* (28) *and* (29).

This proposition can be deduced from Lemma 3.2 using standard techniques, see [1].

3.3 The IMEX scheme and the Black-Scholes equation as a minimization problem

To apply the greedy algorithm (4), our goal is to rewrite the problem (30) as a minimization problem. As a first step, we propose to use an Euler scheme to discretize the problem in time. Let us consider a time discretization grid of $M+1$ points, $\tau_0 = 0 \le \ldots \le \tau_M = T$, where $\tau_i = i\Delta t$ and $\Delta t = \frac{T}{M}$. We introduce a time discretization of the variational formulation (40) where we treat explicitly the non-symmetric part of $\tilde{b}_t$ and implicitly its symmetric terms (IMEX scheme).

For $i = 1,\dots,M$, find $u^i \in \tilde{\mathcal{V}}$ such that

$$\begin{aligned}&\int_\Omega u^i v + \frac{\Delta t}{1+r\Delta t}\int_\Omega (A\nabla u^i)\nabla v - \frac{\Delta t}{2(1+r\Delta t)}\left[\int_\Omega (a\nabla u^i)v + \int_\Omega (a\nabla v)u^i\right]\\ &= \frac{1}{1+r\Delta t}\int_\Omega u^{i-1}v + \frac{\Delta t}{2(1+r\Delta t)}\left[\int_\Omega (a\nabla u^{i-1})v - \int_\Omega (a\nabla v)u^{i-1}\right], \quad \forall v \in \tilde{\mathcal{V}}.\end{aligned} \tag{42}$$

Thus, using that the left hand side of the equation (42) is symmetric in u^i and v, we are led to solve the following sequence of minimization problems

For $i = 1,\dots,M$:

$$\text{Find } u^i \in \tilde{\mathcal{V}}(\Omega) \text{ such that } u^i = \underset{u\in\tilde{\mathcal{V}}(\Omega)}{\operatorname{argmin}}\, \mathcal{E}_i(u) \tag{43}$$

where

$$\begin{aligned}\mathcal{E}_i(u) = &\frac{1}{2}\int_\Omega |u|^2 + \frac{\Delta t}{2(1+r\Delta t)}\left[\int_\Omega (A\nabla u)\nabla u - \int_\Omega (a\nabla u)u\right]\\ &- \frac{1}{1+r\Delta t}\int_\Omega u^{i-1}u - \frac{\Delta t}{2(1+r\Delta t)}\left[\int_\Omega (a\nabla u^{i-1})u - \int_\Omega (a\nabla u)u^{i-1}\right].\end{aligned} \tag{44}$$

Let us introduce the bilinear symmetric form $\hat{a}(u,v)$

$$\hat{a}(u,v) = \int_\Omega uv + \frac{\Delta t}{1+r\Delta t}\int_\Omega (A\nabla u)\nabla v - \frac{\Delta t}{2(1+r\Delta t)}\left[\int_\Omega (a\nabla u)v + \int_\Omega (a\nabla v)u\right], \quad \forall u,v \in \tilde{\mathcal{V}} \tag{45}$$

and the linear form

$$L_{i-1}(v) = \frac{1}{1+r\Delta t}\int_\Omega u^{i-1}v + \frac{\Delta t}{2(1+r\Delta t)}\left[\int_\Omega (a\nabla u^{i-1})v - \int_\Omega (a\nabla v)u^{i-1}\right], \quad \forall v \in \tilde{\mathcal{V}}. \tag{46}$$

We have that $\mathcal{E}_i(u) = \frac{1}{2}\hat{a}(u,u) - L_{i-1}(u)$.

3.4 Stability analysis for the IMEX scheme

In this section, we study the L^2-stability of the IMEX scheme (42). Let us consider u^n the solution of the problem (43) at time τ_n. In our context, the meaning of stability is given in the following definition.

Definition 3.1. *The numerical scheme* (42) *is L^2-stable if there exists a constant $C > 0$ that does not depend on the discretization parameter Δt such that for any initial condition u^0 and for all $n \geq 0$*

$$\|u^n\|_{L^2} \leq C\|u^0\|_{L^2}$$

We will see that the scheme (42) is L^2-stable under a condition on the time step Δt. For the sake of simplicity we take $r = 0$. Let us begin with the following lemma:

Lemma 3.3. *Let us assume that the matrix Ξ defined by $\Xi_{i,j} = \rho_{i,j}\sigma_i\sigma_j,\ i,j = 1,\dots,d$, is positive-definite. Then there exists a constant $\alpha > 0$ such that*

$$\int_\Omega (A\nabla u)\,\nabla u \geq \alpha|u|^2_{\tilde{\mathcal{V}}}, \quad \forall u \in \tilde{\mathcal{V}}. \tag{47}$$

Proof. Using that the matrix Ξ is positive-definite, we have

$$\int_\Omega \sum_{i,j=1}^d \frac{\rho_{ij}\sigma_i\sigma_j}{2} x_i(1-x_i)\frac{\partial u}{\partial x_i} x_j(1-x_j)\frac{\partial u}{x_j} = \int_\Omega Y^T\Xi Y \geq \left(\min_{\lambda\in Sp(\Xi)}\lambda\right)|u|^2_{\tilde{\mathcal{V}}}$$

where $Sp(\Xi)$ denotes the set of eigenvalues of the matrix Ξ and Y is the vector such that $Y_i = x_i(1-x_i)\frac{\partial u}{\partial x_i}$. This proves (47) with $\alpha = \min_{\lambda\in Sp(\Xi)}\lambda$. □

Now, we can state the following proposition:

Proposition 3.2. *The scheme proposed in* (42) *is L^2-stable under the following condition on the time step Δt*

$$\Delta t < \frac{1}{2\left(\frac{4(\|\tilde{a}\|_\infty + \|\operatorname{div}(a)\|_\infty)}{\alpha} + \frac{\alpha}{2}\right)} \tag{48}$$

where the constant α is defined in Lemma 3.3 and $\tilde{a}$ is the vector such that for all $i = 1,\dots d$

$$\tilde{a}_i = \left[r + \sigma_i^2 x_i - \sigma_i^2 + \frac{\sigma_i}{2}\sum_{j=1, j\neq i}^d \rho_{i,j}\sigma_j(2x_j - 1)\right].$$

Proof.
Let us take $r = 0$ and $v = u^i$ in the variational formulation (42). Thus, we obtain

$$\begin{aligned}
&\frac{1}{2\Delta t}\left(\int_\Omega |u^i|^2 - |u^{i-1}|^2\right) + \frac{1}{2\Delta t}\int_\Omega |u^i - u^{i-1}|^2 + \int_\Omega (A\nabla u^i)\nabla u^i - \int_\Omega (a\nabla u^i)u^i \\
&= \frac{1}{2}\left[\int_\Omega (a\nabla u^{i-1})u^i - \int_\Omega (a\nabla u^i)u^{i-1}\right] \\
&\Leftrightarrow \frac{1}{2\Delta t}\left(\int_\Omega |u^i|^2 - |u^{i-1}|^2\right) + \frac{1}{2\Delta t}\int_\Omega |u^i - u^{i-1}|^2 + \int_\Omega (A\nabla u^i)\nabla u^i \\
&-\frac{1}{2}\int_\Omega (a\nabla(u^i + u^{i-1}))u^i - \frac{1}{2}\int_\Omega (a\nabla u^i)(u^i - u^{i-1}) = 0.
\end{aligned} \tag{49}$$

Moreover, we note that $a_i = x_i(1-x_i)\tilde{a}_i$ and that $\tilde{a} \in L^\infty(\Omega)$.
Thus, for all $\epsilon > 0$ we have that

$$\left| \int_\Omega (a\nabla u^i)(u^i - u^{i-1}) \right| \leq \epsilon |u^i|^2_{\tilde{\mathcal{V}}} + \frac{\|\tilde{a}\|_\infty}{4\epsilon} \int_\Omega |u^i - u^{i-1}|^2. \tag{50}$$

Besides, using the integration by parts given by (35) to study the term $\int_\Omega (a\nabla(u^i + u^{i-1}))u^i$, we observe that

$$\begin{aligned} \left| \int_\Omega a\nabla u^i (u^i + u^{i-1}) \right| &= \left| \int_\Omega a\nabla u^i \left(2u^i + (u^{i-1} - u^i)\right) \right|, \\ &\leq \epsilon |u^i|^2_{\tilde{\mathcal{V}}} + \frac{\|\tilde{a}\|_\infty}{4\epsilon} \left(\int_\Omega 2|u^i - u^{i-1}|^2 + \int_\Omega 8|u^i|^2 \right) \end{aligned} \tag{51}$$

and

$$\left| \int_\Omega \operatorname{div}(a) u^i (u^i + u^{i-1}) \right| \leq \frac{\|\operatorname{div}(a)\|_\infty}{4\epsilon} \left(\int_\Omega 2|u^i - u^{i-1}|^2 + \int_\Omega 8|u^i|^2 \right) + \epsilon \int_\Omega |u^i|^2. \tag{52}$$

Then, using (50), (51) and (52), we deduce from (49) that

$$\begin{aligned} &\left[\frac{1}{2\Delta t} - \frac{\|\tilde{a}\|_\infty}{\epsilon} - \frac{\|\operatorname{div}(a)\|_\infty}{\epsilon} - \frac{\epsilon}{2} \right] \int_\Omega |u^i|^2 + (\alpha - \epsilon)|u^i|^2_{\tilde{\mathcal{V}}} \\ &+ \left[\frac{1}{2\Delta t} - \frac{\|\tilde{a}\|_\infty}{4\epsilon} - \frac{\|\operatorname{div}(a)\|_\infty}{4\epsilon} \right] \int_\Omega |u^i - u^{i-1}|^2 \leq \frac{1}{2\Delta t} \int_\Omega |u^{i-1}|^2. \end{aligned} \tag{53}$$

So, if we assume the condition (48) and we choose $\epsilon = \frac{\alpha}{2}$, then we can deduce the following three inequalities:

$$\alpha > \epsilon, \ \frac{1}{2\Delta t} - \frac{\|\tilde{a}\|_\infty + \|\operatorname{div}(a)\|_\infty}{\epsilon} - \frac{\epsilon}{2} > 0 \text{ and } \frac{1}{2\Delta t} > \frac{\|\tilde{a}\|_\infty + \|\operatorname{div}(a)\|_\infty}{4\epsilon}. \tag{54}$$

Consequently, we get

$$\begin{aligned} \int_\Omega |u^i|^2 &\leq \frac{1}{1 - C\Delta t} \int_\Omega |u^{i-1}|^2 \\ &\leq (1 + 2C\Delta t) \int_\Omega |u^{i-1}|^2 \\ &\leq (1 + 2C\Delta t)^M \int_\Omega |u^0|^2 \\ &\leq e^{2CT} \int_\Omega |u^0|^2, \end{aligned}$$

where $C = 2\left(\frac{(\|\tilde{a}\|_\infty + \|\operatorname{div}(a)\|_\infty)}{\epsilon} + \frac{\epsilon}{2} \right)$ is a constant which is independent of the discretization parameter Δt. □

3.5 Implementation of the greedy algorithm for the Black-Scholes PDE

To simplify the notation we consider the case of only three dimensions, but the definition of the algorithm and all the equations below can be easily generalized to a d-dimensional framework.

We recall that the greedy algorithm will generate the function u^i in the separated representation:

$$u^i(x_1, x_2, x_3) = \sum_{k \geq 1} r^i_k \otimes s^i_k \otimes t^i_k(x_1, x_2, x_3).$$

The greedy algorithm (43) is defined as follows: For $i = 1, \dots, M$, iterate on $n \geq 1$

$$(r_n^i, s_n^i, t_n^i) \in \underset{\substack{r \in \tilde{\mathcal{V}}(\Omega_1),\\ s \in \tilde{\mathcal{V}}(\Omega_2),\\ t \in \tilde{\mathcal{V}}(\Omega_3)}}{\operatorname{argmin}} \frac{1}{2}\hat{a}(r \otimes s \otimes t, r \otimes s \otimes t) - L_{i-1}(r \otimes s \otimes t) - \hat{a}\left(\sum_{k=1}^{n-1} r_k^i \otimes s_k^i \otimes t_k^i, r \otimes s \otimes t\right) \quad (55)$$

where $\hat{a}$ is defined by (45) and L_{i-1} by (46).

Then, the Euler equation associated with the problem (55), that is used in practice to implement the algorithm, is given by:

Find $(r_n^i, s_n^i, t_n^i) \in \tilde{\mathcal{V}}(\Omega_1) \times \tilde{\mathcal{V}}(\Omega_2) \times \tilde{\mathcal{V}}(\Omega_3)$ such that for any functions $(r, s, t) \in \tilde{\mathcal{V}}(\Omega_1) \times \tilde{\mathcal{V}}(\Omega_2) \times \tilde{\mathcal{V}}(\Omega_3)$

$$\begin{aligned} &\hat{a}(r_n^i \otimes s_n^i \otimes t_n^i, r \otimes s_n^i \otimes t_n^i + r_n^i \otimes s \otimes t_n^i + r_n^i \otimes s_n^i \otimes t) = L_{i-1}(r \otimes s_n^i \otimes t_n^i + r_n^i \otimes s \otimes t_n^i + r_n^i \otimes s_n^i \otimes t) \\ &+\hat{a}\left(r \otimes s_n^i \otimes t_n^i + r_n^i \otimes s \otimes t_n^i + r_n^i \otimes s_n^i \otimes t, \sum_{k=1}^{n-1} r_k^i \otimes s_k^i \otimes t_k^i\right) \end{aligned} \quad (56)$$

Henceforth, we will consider without loss of generality $s = 0$ and $t = 0$ in order to study in detail each term of this Euler equation (56). We recall that the Euler equation is solved using a fixed point procedure as in (14).

Remark 3.1. *All the high-dimensional integrals in* (56) *are easily calculated using Fubini's rule because the functions in these integrals are separable except for $i = 1$ where the term u^0 appears as follows:*

$$\int_{\Omega_1 \times \Omega_2 \times \Omega_3} r \otimes s_n^i \otimes t_n^i(x_1, x_2, x_3) u^0(x_1, x_2, x_3) dx_1, dx_2 dx_3, \quad (57)$$

The idea used to overcome this practical obstacle is to approximate, in a preliminary step, the initial condition u^0 of the Black-Scholes PDE by a sum of tensor products as explained in Section 2.2. Once the initial condition u^0 has a separated approximation:

$$u^0(x_1, x_2, x_3) = \sum_{k \geq 1} r_k^0 \otimes s_k^0 \otimes t_k^0(x_1, x_2, x_3).$$

the integral (57) *is easy to compute using Fubini's rule.*

Using the space discretization described in Section 2.2 and the notation given by (15), the following vectors will be used:

$$\mathbf{r_n^i} = [r_{n,0}^i, \dots, r_{n,N}^i]^T, \quad \mathbf{s_n^i} = [s_{n,0}^i, \dots, s_{n,N}^i]^T, \quad \mathbf{t_n^i} = [t_{n,0}^i, \dots, t_{n,N}^i]^T,$$

Given the fact that all the terms in the equation (56) admits a separated representation, the equation (56) can be written in a matrix form.

The following matricial expressions allow us to deduce the matricial form for the equation (56):

$$\int_{\Omega_1 \times \Omega_2 \times \Omega_3} (r_n^i \otimes s_n^i \otimes t_n^i)(r \otimes s_n^i \otimes t_n^i) = [\mathbf{t_n^i}^T M \mathbf{t_n^i}][\mathbf{s_n^i}^T M \mathbf{s_n^i}] M \mathbf{r_n^i},$$

$$\begin{aligned} \int_{\Omega_1 \times \Omega_2 \times \Omega_3} \left(A \nabla (r_n^i \otimes s_n^i \otimes t_n^i)\right) \nabla (r \otimes s_n^i \otimes t_n^i) = &\left(\frac{\sigma_2^2}{2}[\mathbf{t_n^i}^T M \mathbf{t_n^i}][\mathbf{s_n^i}^T L \mathbf{s_n^i}] + \frac{\sigma_3^2}{2}[\mathbf{t_n^i}^T L \mathbf{t_n^i}][\mathbf{s_n^i}^T M \mathbf{s_n^i}]\right) M \mathbf{r_n^i} \\ &+ \left(\frac{\rho_{1,2}\sigma_1\sigma_2}{2}[\mathbf{t_n^i}^T M \mathbf{t_n^i}][\mathbf{s_n^i}^T D \mathbf{s_n^i}] + \frac{\rho_{1,3}\sigma_1\sigma_3}{2}[\mathbf{t_n^i}^T D \mathbf{t_n^i}][\mathbf{s_n^i}^T M \mathbf{s_n^i}]\right) \\ &(D + D^T)\mathbf{r_n^i} + \frac{\sigma_1^2}{2}[\mathbf{s_n^i}^T M \mathbf{s_n^i}][\mathbf{t_n^i}^T M \mathbf{t_n^i}] L \mathbf{r_n^i}, \end{aligned}$$

$$\int_{\Omega_1\times\Omega_2\times\Omega_3}\left(a\nabla(r_n^i\otimes s_n^i\otimes t_n^i)\right)(r\otimes s_n^i\otimes t_n^i) = [\mathbf{s_n^i}^T M\mathbf{s_n^i}][\mathbf{t_n^i}^T M\mathbf{t_n^i}]B\mathbf{r_n^i} + \Big(\frac{\rho_{1,2}\sigma_1\sigma_2}{2}[\mathbf{s_n^i}^T C\mathbf{s_n^i}][\mathbf{t_n^i}^T M\mathbf{t_n^i}]$$
$$+\frac{\rho_{1,3}\sigma_1\sigma_3}{2}[\mathbf{t_n^i}^T C\mathbf{t_n^i}][\mathbf{s_n^i}^T M\mathbf{s_n^i}]\Big)D\mathbf{r_n^i} + \Big(\frac{\rho_{1,2}\sigma_1\sigma_2}{2}[\mathbf{t_n^i}^T M\mathbf{t_n^i}][\mathbf{s_n^i}^T D\mathbf{s_n^i}]$$
$$+\frac{\rho_{1,3}\sigma_1\sigma_3}{2}[\mathbf{t_n^i}^T D\mathbf{t_n^i}][\mathbf{s_n^i}^T M\mathbf{s_n^i}]\Big)C\mathbf{r_n^i} + \Big([\mathbf{t_n^i}^T M\mathbf{t_n^i}][\mathbf{s_n^i}^T B\mathbf{s_n^i}]$$
$$+\frac{\rho_{2,3}\sigma_2\sigma_3}{2}[\mathbf{t_n^i}^T C\mathbf{t_n^i}][\mathbf{s_n^i}^T D\mathbf{s_n^i}] + [\mathbf{t_n^i}^T B\mathbf{t_n^i}][\mathbf{s_n^i}^T M\mathbf{s_n^i}]$$
$$+\frac{\rho_{2,3}\sigma_2\sigma_3}{2}[\mathbf{t_n^i}^T D\mathbf{t_n^i}][\mathbf{s_n^i}^T C\mathbf{s_n^i}]\Big)M\mathbf{r_n^i}$$

where the matrices M, L, B, C, D are explicitly computable tridiagonal matrices of size $N\times N$, with N the number of intervals in each direction. The computation of the components for these matrices boils down to one-dimensional integrals.

In this way, solving (56) with a fixed point procedure allows us to obtain for a fixed i such that $1\le i\le N$, the n-th term of the sum $\sum_{k=1}^n r_k^i\otimes s_k^i\otimes t_k^i$ which is an approximation of the solution at time $t_i = i\Delta t$ of the problem (30).

4 Numerical results

4.1 Testing the method against an analytical solution

In this part, we apply our greedy algorithm (55) to solve the problem (30) with the following initial condition:

$$u(0,x_1,\dots,x_d) = \left(K - \prod_{i=1}^d \frac{x_i}{(1-x_i)}\right)^+ \tag{58}$$

for which the solution is analytically known.

Using the Feynman-Kac theorem (21) we get that the solution of the PDE (23) is given by $\mathbb{E}\left[e^{-rT}(K-\prod_{i=1}^d S_T^i)^+\right]$, which is possible to calculate analytically in the Black-Scholes model. We have:

$$\mathbb{E}\left[e^{-rT}(K-\prod_{i=1}^d S_T^i)^+\right] = e^{-rT}K\mathbb{P}\left(K>\prod_{i=1}^d S_T^i\right) - e^{-rT}\mathbb{E}\left[\prod_{i=1}^d S_T^i\mathbf{1}_{\{K>\prod_{i=1}^d S_T^i\}}\right] \tag{59}$$

We get the quantity $\mathbb{P}\left(K>\prod_{i=1}^d S_T^i\right)$ as follows:

$$\begin{aligned}\mathbb{P}\left(K>\prod_{i=1}^d S_T^i\right) &= \mathbb{P}\left(e^{\sum_{i=1}^d X_T^i} < \frac{K}{\prod_{i=1}^d S_0^i}\right)\\ &= \mathbb{P}\left(Y<\log\left(\frac{K}{\prod_{i=1}^d S_0^i}\right)\right)\end{aligned}$$

where $X_T^i = (r-\frac{\sigma_i^2}{2})T + \sigma_i W_T^i$ and $Y=\sum_{i=1}^d X_T^i$ is a normal random variable with mean equals to $\sum_{i=1}^d\left(r-\frac{\sigma_i^2}{2}\right)T$ and variance given by $\sum_{i=1}^d\sum_{j=1}^d \rho_{ij}\sigma_i\sigma_j T$.

Besides, we remark that:

$$\mathbb{E}\left[\prod_{i=1}^d S_T^i\mathbf{1}_{\{K>\prod_{i=1}^d S_T^i\}}\right] = \mathbb{E}\left[e^Y\mathbf{1}_{\{e^Y<\frac{K}{\prod_{i=1}^d S_0^i}\}}\right]\prod_{i=1}^d S_0^i$$

where the last term can be calculated analytically. We remark that the analytic solution (59) is not separable with respect to each coordinate.

We present in figure 4 a numerical example of the solution obtained with our algorithm and the analytic solution. In figure 5 we see the same surface but intersected with the plane $x_1 = x_2$.

Figure 6 shows the convergence curves, i.e, the L^2 relative error with respect to the number of iterations of the algorithm according to the dimension. We note that the number of iterations needed to obtain convergence increases as the dimension of the problem increases.

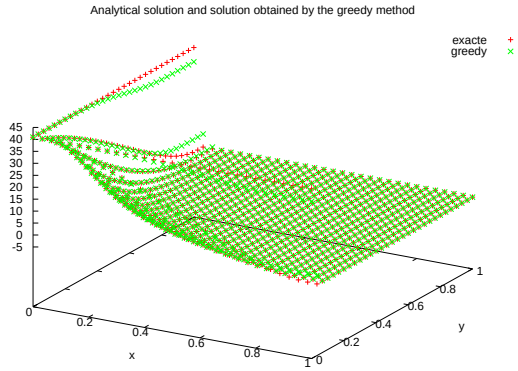

Figure 4: The analytical solution and the numerical one obtained with our method for the problem (23) with the initial condition (58) in a two-dimensional framework. For this example, we consider $\Delta t = \frac{1}{100}$ and $\Delta x = \frac{1}{30}$.

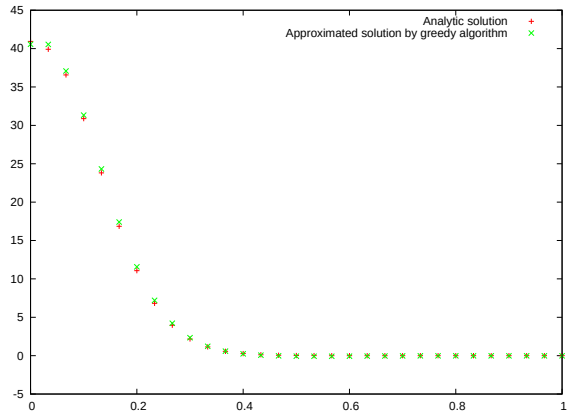

Figure 5: The analytical solution and the numerical one obtained with our method for the problem (23) with the initial condition (58) in a two-dimensional framework. We represent the intersection between the surface in figure 4 and the plane $x_1 = x_2$. For this example, we consider $\Delta t = \frac{1}{100}$ and $\Delta x = \frac{1}{30}$.

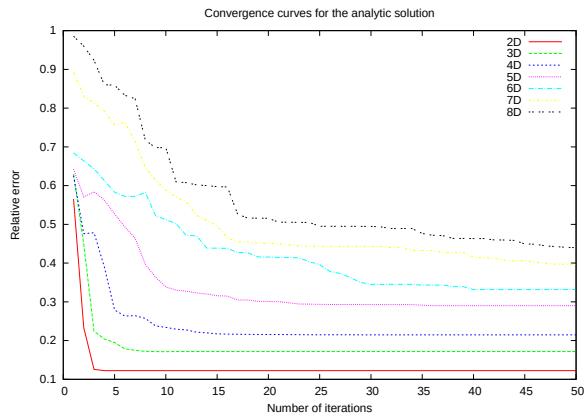

Figure 6: Convergence curves for the solution at time T of the equation (23) with initial condition given by (58). To obtain this curves, we consider $\Delta x = 0.1$ for each dimension and $\Delta t = 0.01$.

4.2 Results on the Black-Scholes equation

In this section, we show the results that we obtained applying our greedy algorithm described in the previous section to the Black-Scholes equation.

Figure 7 represents the approximation of the solution at time T to the problem (23) with initial condition (27) obtained using our greedy algorithm defined by the equations (56)

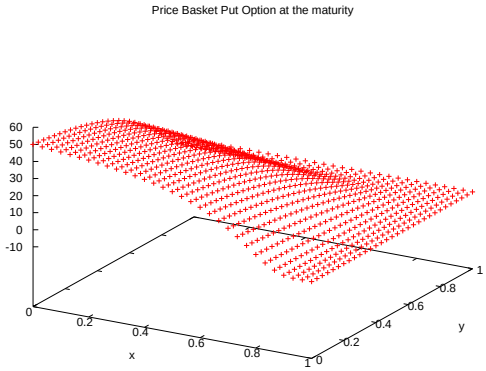

Figure 7: The approximated solution obtained with our method for the problem (23) with the initial condition (27).

Figure 8 represent the price of a basket put option when all the assets take the same value, i.e. $S_1 = \ldots = S_d$. Precisely, Figure 8 compares prices obtained with different discretizations Δx.

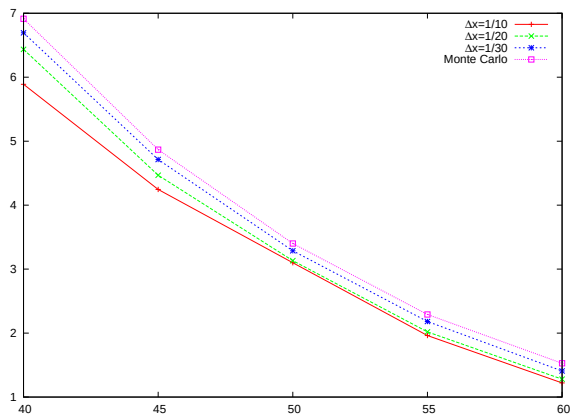

Figure 8: The approximated solution obtained with our method for the problem (23) with the initial condition (27) in a four-dimensional framework. For these results, we set $\Delta x = 0.1, 0.2$ and $\Delta t = 0.01$.

In terms of computational time, the greedy approach (55) is not competitive compared to Monte Carlo methods when one is interested in the price of only one value of the spot, but on the other side the curve of prices is obtained for any time $t \in [0, T]$ and any price spot.

4.3 Application as a variance reduction method

In this part, we show that we can use the solution obtained by the greedy method described above in order to find a control variable to reduce the variance when calculating the price an of option.

We can re-write the equations (19) and (20) which define the Black-Scholes model as follows:

$$\frac{dS_t^i}{S_t^i} = rdt + \sigma^i \sum_{j=1}^{d} H_{i,j} dW_t^j \tag{60}$$

Dimension	Without variance reduction	With variance reduction
4	0.1233	0.0012
5	0.1204	0.0034
6	0.1197	0.0078
7	0.1245	0.0113
8	0.1257	0.0254

Table 2: Variance with a correlation parameter $\rho_{i,j} = 0.9$ constant between all the assets.

where W is a d-dimensional Brownian motion and the matrix H verifies $HH^t = \Sigma$ where Σ is a $d \times d$ matrix such that $\Sigma_{i,j} = 1$ if $j = i$ and $\Sigma_{i,j} = \rho_{ij}$ otherwise.

We recall that the price of a basket put option is given by

$$\mathbb{E}\left[e^{-rT} f(S_T^1, S_T^2, \ldots, S_T^d)\right]$$

where $f(S_T^1, S_T^2, \ldots, S_T^d) = \left(K - \frac{1}{d}\sum_{i=1}^d S_T^i\right)_+$.

Now, for the sake of simplicity, let us consider $r = 0$. Generalization to $r \neq 0$ is straightforward. Let us introduce the Kolmogorov equation:

$$\begin{cases} \partial_t \hat{P} - \frac{1}{2} A : \nabla^2 \hat{P} = 0 \\ \hat{P}(0, x) = f(x) \end{cases} \tag{61}$$

where $A = FH(FH)^T$ and F is a diagonal matrix such that $F_{i,i} = \sigma^i S^i$ for $i = 1, \ldots, d$. Notice that

$$\hat{P}(T, S_0) = \mathbb{E}\left[f(S_T^1, S_T^2, \ldots, S_T^d)\right].$$

Therefore, we have

$$\hat{P}(0, S_T) - \hat{P}(T, S_0) = \int_0^T FH\ \nabla \hat{P}(T - t, S_t) dB_t$$

and thus,

$$\hat{P}(T, S_0) = f(S_T) - \int_0^T FH\ \nabla \hat{P}(T - t, S_t) dB_t \tag{62}$$

The random variable $Y = \int_0^T FH\ \nabla \hat{P}(T - t, S_t) dB_t$ has zero mean and is a perfect control variable since

$$\mathrm{Var}\left[f(S_T) - Y\right] = 0.$$

As we do not know the solution $\hat{P}$, in practice, we calculate an approximation $\hat{P}^\star$ of $\hat{P}$ using the greedy algorithm presented in Section 3. Therefore, we obtain an approximated control variable $Y^\star = \int_0^T FH\ \nabla \hat{P}^\star(T - t, S_t) dB_t$ and we can compute an approximation of $\hat{P}(T, S_0)$ by Monte Carlo, computing the following quantity:

$$\mathbb{E}\left[f(S_T) - \int_0^T FH\ \nabla \hat{P}^\star(T - t, S_t) dB_t\right].$$

We remark that this idea can be applied to any payoff function and that for a new value of S_0 we use the same approximation $\hat{P}^\star$.

In Table 2 and 3 we present the performance of our variance reduction method compared with the variance obtained with the classical method, i.e. calculating directly $\mathbb{E}[f(S_T)]$.

For two typical values of the correlation, we observe that the reduction of the variance is important, for example, up to a factor 6 in dimension 8.

Dimension	Without variance reduction	With variance reduction
4	0.1256	0.0023
5	0.1248	0.0045
6	0.1230	0.0096
7	0.1199	0.0158
8	0.1232	0.0296

Table 3: Variance with a correlation parameter $\rho_{i,j} = 0.1$ constant between all the assets.

References

[1] Y. Achdou and O. Pironneau. *Computational methods for option pricing*, volume 30 of *Frontiers in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2005.

[2] A. Ammar, B. Mokdad, F. Chinesta, and R. Keunings. A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids. *J. Non-Newtonian Fluid Mech.*, 139:153–176, 2006.

[3] E. Cancès, V. Ehrlacher, and T. Lelievre. Convergence of a greedy algorithm for high-dimensional convex nonlinear problems. *Mathematical Models and Methods in Applied Sciences*, 21(12):2433–2467, 2011.

[4] R. A. DeVore and V. N. Temlyakov. Some remarks on greedy algorithms. *Adv. Comput. Math.*, 5:173–187, 1996.

[5] P. Ladevèze and J. G. Simmonds. *Nonlinear computational structural mechanics: new approaches and non-incremental methods of calculations*. Springer, 1999.

[6] C. Le Bris, T. Lelièvre, and Y. Maday. Results and questions on a nonlinear approximation approach for solving high-dimensional partial differential equations. *Constructive Approximation*, 30(3):621–651, 2009.

[7] A. Nouy. A generalized spectral decomposition technique to solve a class of linear stochastic partial differential equations. *Comput. Methods Appl. Mech. Engrg.*, 196:4521–4537, 2007.

[8] A. Nouy. Proper generalized decompositions for a priori model reduction of problems formulated in tensor product spaces: Alternative definitions and algorithms. *Proceedings of the Seventh International Conference on Engineering Computational Technology, Civil-Comp Press*, 2010.

[9] D. Pommier. *Méthodes numériques sur des grilles sparse appliquées à l'évaluation d'options en finance*. PhD thesis, Université Pierre et Marie Curie, 2008.

[10] V. N. Temlyakov. Greedy approximation. *Acta-Numerica*, 17:235–409, 2008.