REVISTA DE ECONOMÍA FINANCIERA
2010; **20**:51–68

Algorithms and Numerical Methods for High Dimensional Financial Market Models

Antonio Falcó[1]

ABSTRACT

A major challenge in Computational Finance is the pricing of options that depend on a large number of risk factors. Prominent examples are basket or index options where dozens or even hundreds of stocks constitute the underlying asset and determine the dimensionality of the corresponding parabolic equation. A number of problems in high-dimensional spaces have been addressed by the usual technique of separation of variables. In order to use the separated representation for numerical analysis applications, many algorithms and operations need to be translated into this framework. The aim this paper is present and review some of these techniques in the context of High Dimensional Financial Markets Models.

Keywords: Option Pricing, Numerical Methods.

JEL Classification: G13, C02.

Algoritmos y métodos numéricos para modelos de alta dimensión en mercados financieros

RESUMEN

Uno de los mayores retos de las finanzas computacionales consiste en la valoración de opciones que dependen de un elevado número de factores de riesgo. Ejemplos prominentes son las opciones sobre índices y cestas en los cuales docenas o incluso centenares de activos constituyen el activo subyacente y determinan la dimensionalidad de la ecuación parabólica correspondiente. Un buen número de problemas han sido analizados empleando la técnica usual de separación de variables. Con el fin de emplear la representación de separación de variables, muchos algoritmos y operaciones tienen que ser transformados bajo este nuevo enfoque. El objetivo del presente trabajo en presentar y revisar algunas de estas técnicas en el contexto de los modelos de elevada dimensión en los mercados financieros españoles.

Palabras Clave: Valoración de Opciones, Métodos Numéricos.

Clasificación JEL: G13, C02

1. INTRODUCTION

Numerous models encountered in science and engineering remain nowadays, despite the impressive recent progresses attained in computational simulation techniques, intractable when the usual and well experienced discretization methods are applied for their numerical simulation. Thus, different challenging issues are waiting for the proposal of a new alternative advanced simulation framework, the brute force approach being no more a valuable alternative.

[1] Departamento de Ciencias Físicas, Matemáticas y de la Computación
Universidad CEU Cardenal Herrera
San Bartolomé 55, 46115 Alfara del Patriarca (Valencia), Spain.
afalco@uch.ceu.es

Recibido el 9 de Septiembre de 2009
Aceptado el 4 de Enero de 2010

A first challenging issue concerns the treatment of highly multidimensional models arising from Quantum Mechanics or Kinetic Theory descriptions of solids and Complex Fluids (Chinesta *et al.* (2007)), including micro and nano-structured materials (Chinesta, Ammar and Joyot (2008)). Other multidimensional models are encountered in biology, genetics and chemistry where the so called chemical master equation plays a key role (Sreenath, Cho and Wellstead (2009)). Curse of dimensionality also arises in stochastic models involving numerous random variables (Nouy (2009)). The main challenge in the treatment of this kind of models is related to their multidimensionality because when one applies standard mesh based discretization techniques the number of degrees of freedom involved scales exponentially with the dimension of the space concerned. Thus, in high dimensions, usual meshes cannot be defined at all, needing the proposal of new advanced strategies able to circumvent the terrible curse of dimensionality. Until now, the treatment of this kind of models was restricted to the ones defined in moderate multidimensional spaces where for example the sparse grid based methods works (Bungartz and Griebel (2004)).

The computational complexity of most algorithms in dimension d grows exponentially in d. Even simply accessing a vector in dimension d requires N^d operations, here N represents the number of entries in each direction. This effect has been dubbed the curse of dimensionality (Bellman (1961)), and it is the single greatest impediment to computing in higher dimensions.

A number of problems in high-dimensional spaces have been addressed by the usual technique of separation of variables. Pierre Ladèveze proposed several years ago a powerful technique for addressing this kind of challenging models that he called the LATIN method (Ladeveze (1999)). The LATIN method integrates many ingredients leading to a robust, powerful, efficient and accurate discretization technique especially well adapted for treating transient multi-scale non-linear models. The two most outstanding ingredients are (i) the decoupling between a linear-global problem and a non- linear-local one, both defined in the whole space-time domain; and (ii) a space- time separated representation of the model variables in order to accelerate the solution of the linear-global problem. The former separated representation was called by Ladeveze in the 80's "radial approximation", and in our knowledge it was the first time that separated representations were applied, in this case in the field of Computational Mechanics.

The concept of separated representation was first introduced by Beylkin and Mohlenkamp (2005). More precisely, for a given map

$$u : [0,1]^d \subset \mathbb{R}^d \longrightarrow \mathbb{R},$$

we say that it has a *separable representation* if

$$u(x_1,\ldots,x_d) = \sum_{j=1}^{\infty} u_1^{(j)}(x_1)\cdots u_d^{(j)}(x_d) \tag{1}$$

Thus, if we consider a mesh of $[0,1]$ in the x_k-variable given by N_k-mesh points, $1 \leq k \leq d$, then we can write a discrete version of (1):

$$u(x_{i_1},\ldots,x_{i_d}) = \sum_{j=1}^{\infty} u_1^{(j)}(x_{i_1})\cdots u_d^{(j)}(x_{i_d}), \tag{2}$$

where $1 \leq i_k \leq N_k$ for $1 \leq k \leq d$. Observe that if for each $1 \leq k \leq d$, $\mathbf{x}_k^j \in \mathbb{R}^{N_k}$ denotes the vector with components $u_k^{(j)}(x_{i_k})$ for $1 \leq i_k \leq N_k$, then, by using a $\otimes$-tensor product representation, (2) it is equivalent to

$$\mathbf{u} = \sum_{j=1}^{\infty} \mathbf{x}_1^j \otimes \cdots \otimes \mathbf{x}_d^j. \tag{3}$$

We point out that (3) is an useful expression to implemented numerical algorithms using the Matlab and Octave function `kron`.

The main point of the separated representation is that since we only operate on low dimensional objects, the computational complexities are formally linear in the dimension of this objects rather than exponential. In particular a vector $\mathbf{x}_1 \otimes \cdots \otimes \mathbf{x}_d \in \mathbb{R}^{N^d}$ needs dN entries to store it in the memory of the computer. Assume that

$$\mathbf{y} \approx \widehat{\mathbf{y}} = \sum_{j=1}^{r} \mathbf{x}_1^j \otimes \cdots \otimes \mathbf{x}_d^j,$$

for some $r \geq 1$, then to store $\widehat{\mathbf{y}}$ we only need rdN entries.

Recently, in Ammar *et al.* (2001) and Ammar *et al.* (2007), Ammar, Mokdad, Chinesta and Keunings propose the use of a separated representation, which allows to define a tensor product approximation basis as well as to decouple the numerical integration of a high dimensional model in each dimension.

From the mathematical point of view some results concerning the convergence were obtained by Ammar, Chinesta and Falcó (2009) in a finite dimensional setting. From a variational point of view and for infinite dimension, some results has been obtained by Le Bris, Lelièvre and Maday (2009). Its connection with the POD (Proper Orthogonal Decomposition) was also analyzed in Ladeveze, Passieux and Neron (2009). In fact, introducing the separated representation into the weak formulation of the model the different functions involved in the separated representation can be computed. However, these functions that are optimal from the point of view of the strategy considered, are no more orthogonal. For this reason, the resulting approximation is called Proper Generalized Decomposition (PGD).

The paper is organized as follows. in the next section we introduce relationship between separated representation and the best n-term approximation. In Section 3 we give as a case study a computational approach to the Fundamental Theorem of Asset Pricing in single period market. Section 4 is devoted to the problem of the first passage time for a high-dimensional Brownian motion. Finally, some comments and remarks are given.

2. ON THE RELATIONSHIP BETWEEN SEPARATED REPRESENTATION AND THE BEST N-TERM APPROXIMATION

Before to end this section we describe some of the notation used in this paper. We denote the set of $N \times M$-matrices by $\mathbb{R}^{N \times M}$, and the transpose of a matrix A is denoted A^T. By $\langle \mathbf{x}, \mathbf{y} \rangle$ we denote the usual Euclidean inner product given by $\mathbf{x}^T\,\mathbf{y} = \mathbf{y}^T\,\mathbf{x}$ and its corresponding 2-norm, $\|x\|_2 = \langle \mathbf{x}, \mathbf{x} \rangle^{1/2}$. The matrix I_N is the $N \times N$-identity matrix and when the dimension is clear from the context, we simply denote it by I. Given a sequence $\{\mathbf{u}_j\}_{j=0}^{\infty} \subset \mathbb{R}^N$, we say that a vector $\mathbf{u} \in \mathbb{R}^N$ can be written as

$$\mathbf{u} = \sum_{j=0}^{\infty} \mathbf{u}_j$$

if and only if

$$\lim_{n \to \infty} \sum_{j=0}^{n} \mathbf{u}_j = \mathbf{u}$$

holds in the $\| \cdot \|_2$-topology. Now, we recall the definition and some properties of the Kronecker product. The Kronecker product of $A \in \mathbb{R}^{N_1' \times N_1}$ and $B \in \mathbb{R}^{N_2' \times N_2}$, written $A \otimes_K B$, is the tensor algebraic operation defined as

$$A \otimes_K B = \begin{bmatrix} A_{1,1}B & A_{1,2}B & \cdots & A_{1,N_2N_2'}B \\ A_{2,1}B & A_{2,2}B & \cdots & A_{2,N_2N_2'}B \\ \vdots & \vdots & \ddots & \vdots \\ A_{N_1N_1',1}B & A_{N_1N_1',2}B & \cdots & A_{N_1N_1',N_2N_2'}B \end{bmatrix} \in \mathbb{R}^{N_1'N_2' \times N_1N_2}.$$

ISSN: 1697-9761

Suppose that for given a linear Partial Differential Equation, and after a discretization by means Finite Elements, we need to solve the linear system

$$A\mathbf{u} = \mathbf{f}, \tag{4}$$

where A is a $(N_1 \cdots N_d) \times (N_1 \cdots N_d)$-dimensional invertible matrix, for some $N_1, N_2, \ldots, N_d \in \mathbb{N}$. Then from all said above, it seems reasonable to find an approximate solution

$$A^{-1}\mathbf{f} \approx \mathbf{u}_n = \sum_{j=1}^{n} \mathbf{x}_1^j \otimes_K \cdots \otimes_K \mathbf{x}_d^j$$

for some $n \geq 1$ and where $\mathbf{x}_i^j \in \mathbb{R}^{N_i}$ for $i = 1, 2, \ldots, d$ and $j = 1, 2, \ldots, n$; satisfying that

$$\lim_{n\to\infty} \left\| A^{-1}\mathbf{f} - \mathbf{u}_n \right\|_2 = 0,$$

that is,

$$A^{-1}\mathbf{f} = \sum_{j=1}^{\infty} \mathbf{x}_1^j \otimes_K \cdots \otimes_K \mathbf{x}_d^j.$$

For each $n \in \mathbb{N}$, we define the set

$$\mathcal{S}_n = \{\mathbf{x} \in \mathbb{R}^{N_1 \cdots N_d} : \operatorname{rank}_{\otimes_K} \mathbf{x} \leq n\},$$

introduced in de Silva and Lim (2009), in the following way. Given $\mathbf{x} \in \mathbb{R}^{N_1 \cdots N_d}$ we say that $\mathbf{x} \in \mathcal{S}_1 = \mathcal{S}_1(N_1, N_2, \ldots, N_d)$ if $\mathbf{x} = \mathbf{x}_1 \otimes_K \mathbf{x}_2 \otimes_K \cdots \otimes_K \mathbf{x}_d$, where $\mathbf{x}_i \in \mathbb{R}^{N_i}$, for $i = 1, \ldots, d$. For $n \geq 2$ we define inductively $\mathcal{S}_n = \mathcal{S}_n(N_1, N_2, \ldots, N_d) = \mathcal{S}_{n-1} + \mathcal{S}_1$, that is,

$$\mathcal{S}_n = \left\{ \mathbf{x} : \mathbf{x} = \sum_{i=1}^{k} \mathbf{x}^{(i)},\ \mathbf{x}^{(i)} \in \mathcal{S}_1 \text{ for } 1 \leq i \leq k \leq n \right\}.$$

Note that $\mathcal{S}_n \subset \mathcal{S}_{n+1}$ for all $n \geq 1$.

We say that $\mathbf{u}_n$ is the n-best term approximation solution of the linear system (4) if

$$\mathbf{u}_n \in \operatorname{argmin}_{\mathbf{x}\in\mathcal{S}_n} \|\mathbf{b} - A\mathbf{x}\|_2. \tag{5}$$

Unfortunately, in de Silva and Lim (2009), the authors prove that $\mathcal{S}_n = \mathcal{S}_n(N_1, N_2, \ldots, N_d)$ for $n \geq 2$ and $d \geq 3$ is not a closed set in any norm-topology in $\mathbb{R}^{N_1 \cdots N_d}$. Thus, the following problem

$$\min_{\mathbf{x}\in\mathcal{S}_n} \|\mathbf{b} - A\mathbf{x}\|_2, \tag{6}$$

is ill posed for $n \geq 2$ and $d \geq 3$. However, it is possible to show that the set $\mathcal{S}_1$ is a closed set in $\mathbb{R}^{N_1 \cdots N_d}$ in any norm-topology. In consequence given an invertible matrix $A \in \mathbb{R}^{N_1 N_2 \cdots N_d \times N_1 N_2 \cdots N_d}$, and for each $\mathbf{b} \in \mathbb{R}^{N_1 \cdots N_d}$ we obtain that

$$\operatorname{argmin}_{\mathbf{x}\in\mathcal{S}_1} \|\mathbf{b} - A\mathbf{x}\|_2 \neq \emptyset. \tag{7}$$

This allow to consider the following iterative scheme. Let $\mathbf{u}_0 = \mathbf{y}_0 = \mathbf{0}$, and for each $n \geq 1$ take

$$\begin{aligned} \mathbf{r}_{n-1} &= \mathbf{f} - A\mathbf{u}_{n-1}, && (8)\\ \mathbf{u}_n &= \mathbf{u}_{n-1} + \mathbf{y}_n \text{ where } \mathbf{y}_n \in \operatorname{argmin}_{\mathbf{y}\in\mathcal{S}_1} \|\mathbf{r}_{n-1} - A\mathbf{y}\|_2. && (9) \end{aligned}$$

Note that for each vector $\mathbf{f} \in \mathbb{R}^{N_1 \cdots N_d}$ and each invertible matrix $A \in \mathbb{R}^{N_1 N_2 \cdots N_d \times N_1 N_2 \cdots N_d}$, we can construct for each n, by using (8)-(9), a vector

$$\mathbf{u}_n = \sum_{j=1}^{n} \mathbf{y}_n \in \mathcal{S}_n \setminus \mathcal{S}_{n-1},$$

here we assume that $\mathbf{y}_j \neq \mathbf{0}$ for $1 \leq j \leq n$, that is, $\operatorname{rank}_{\otimes} \mathbf{u}_n = n$. Since $\mathbf{u}_n \approx A^{-1}\mathbf{f}$, we define the $\operatorname{rank}_{\otimes}$ for $A^{-1}\mathbf{f}$ obtained by the Greedy Rank-One Update Algorithm (8)-(9) as

$$\operatorname{rank}_{\otimes}^{G}(A^{-1}\mathbf{f}) = \begin{cases} \infty & \text{if} \quad \{j \geq 1 : \mathbf{y}_j = \mathbf{0}\} = \emptyset, \\ \min\{j \geq 1 : \mathbf{y}_j = \mathbf{0}\} - 1 & \text{otherwise.} \end{cases}$$

The following theorem due to Amine, Chinesta and Falcó (2009), gives the convergence of the Greedy Rank-One Update Procedure (8)-(9) (given in pseudocode form in Algorithm 1) for solving linear systems with full rank matrix.

Theorem 1. *Let* $\mathbf{f} \in \mathbb{R}^{N_1 N_2 \cdots N_d}$ *and* $A \in \mathbb{R}^{N_1 N_2 \cdots N_d \times N_1 N_2 \cdots N_d}$, *be an invertible matrix. Then, by using the iterative scheme (8)-(9), we obtain that the sequence* $\{\|\mathbf{r}_n\|_2\}_{n=0}^{\operatorname{rank}_{\otimes}^{G}(A^{-1}\mathbf{f})}$, *is strictly decreasing and*

$$A^{-1}\mathbf{f} = \lim_{n\to\infty} \mathbf{u}_n = \sum_{j=0}^{\operatorname{rank}_{\otimes}^{G}(A^{-1}\mathbf{f})} \mathbf{y}_j. \tag{10}$$

Moreover, the rate of convergence is given by

$$\frac{\|\mathbf{r}_n\|_2}{\|\mathbf{r}_0\|_2} = \prod_{j=1}^{n} \sin\theta_j \tag{11}$$

for $1 \leq n \leq \operatorname{rank}_{\otimes}^{G}(A^{-1}\mathbf{f})$ *where*

$$\theta_j = \arccos\left(\frac{\langle \mathbf{r}_{j-1}, A\mathbf{y}_j\rangle}{\|\mathbf{r}_{j-1}\|_2 \|A\mathbf{y}_j\|_2}\right) \in (0, \pi/2) \tag{12}$$

for $1 \leq j \leq n$.

From (10) we obtain that if

$$\operatorname{rank}_{\otimes}^{G}(A^{-1}\mathbf{f}) < \infty,$$

then $\|\mathbf{r}_n\|_2 = 0$ for all $n > \operatorname{rank}_{\otimes}^{G}(A^{-1}\mathbf{f})$. Thus, the above theorem allow to us to construct a procedure, that we give in the pseudocode form in Algorithm 1, under the assumption that we have a numerical method in order to find a $\mathbf{y}$ solving (7) (see the step 5 in Algorithm 1) and that we introduce below.

Algorithm 1 Greedy Rank-One Update

```
 1: procedure GROU($\mathbf{f}, A, \varepsilon$, tol, rank_max)
 2:     $\mathbf{r}_0 = \mathbf{f}$
 3:     $\mathbf{u} = \mathbf{0}$
 4:     for $i = 0, 1, 2, \ldots,$ rank_max do
 5:         $\mathbf{y} =$ procedure $(\min_{\operatorname{rank}_{\otimes}\mathbf{x} \leq 1} \|\mathbf{r}_i - A\mathbf{y}\|_2^2)$
 6:         $\mathbf{r}_{i+1} = \mathbf{r}_i - A\mathbf{y}$
 7:         $\mathbf{u} \leftarrow \mathbf{u} + \mathbf{y}$
 8:         if $\|\mathbf{r}_{i+1}\|_2 < \varepsilon$ or $|\|\mathbf{r}_{i+1}\|_2 - \|\mathbf{r}_i\|_2| <$ tol then goto 13
 9:         end if
10:     end for
11:     return $\mathbf{u}$ and $\|\mathbf{r}_{\texttt{rank\_max}}\|_2$.
12:     break
13:     return $\mathbf{u}$ and $\|\mathbf{r}_{i+1}\|_2$
14: end procedure
```

2.1. A Block Coordinated Descent Approach for the Rank-One Minimization Problem

Now, we study the Rank-One minimization problem

$$\min_{\mathbf{x}\in\mathcal{S}_1} \|\mathbf{b}-A\mathbf{x}\|_2\,, \tag{13}$$

that we can write as the following unconstrained optimization problem:

$$\min_{(\mathbf{x}_1,\cdots,\mathbf{x}_d)\in\mathbb{R}^{N_1+\cdots+N_d}} \|\mathbf{b}-A\left(\mathbf{x}_1\otimes\cdots\otimes\mathbf{x}_d\right)\|_2\,. \tag{14}$$

A popular method for minimizing a real-valued continuously differentiable function Φ of $N_1+\cdots+N_d$ real variables, subject to bound constrains, is the (block) coordinate descend method. In this method, the coordinates are partitioned into N_k blocks and, at each iteration, $\mathbf{b}$ is minimized with respect to one of the coordinate blocks while the others are held fixed (see Algorithm 2). These cyclic methods have the advantage of not requiring any information about the gradient to determine the descent directions. However, their convergence properties are poorer than steepest descend methods. Moreover, its are attractive because of their easy implementation in some particular cases as we will see below.

Algorithm 2 A Block Coordinated Descent Algorithm

```
Given $\Phi : \mathbb{R}^{N_1+\cdots+N_d}\to\mathbb{R}$
Initialize $\mathbf{x}_i^0\in\mathbb{R}^{N_i}$ for $i=1,2\ldots,d$.
for $n=1,2,\ldots$ do
    for $k=1,2,\ldots,d$ do
        $\mathbf{x}_k^n\in\arg\min_{\mathbf{x}_k\in\mathbb{R}^{N_k}}\Phi(\mathbf{x}_1^n,\ldots,\mathbf{x}_{k-1}^n,\mathbf{x}_k,\mathbf{x}_{k+1}^{n-1},\ldots,\mathbf{x}_d^{n-1})$
    end for
end for
```

It is possible to show the following result (see Ammar, Chinesta and Falcó (2009)).

Theorem 2. *Let* $\mathbf{b}\in\mathbb{R}^{N_1N_2\cdots N_d}$ *and* $A\in\mathbb{R}^{N_1N_2\cdots N_d\times N_1N_2\cdots N_d}$, *be an invertible matrix. Assume that for each* $k\in\{1,2,\ldots,d\}$ *the* $(N_1\cdots N_d)\times N_k$*-matrix*

$$Z_k = A\left(\mathbf{x}_1\otimes\cdots\otimes\mathbf{x}_{k-1}\otimes I_{N_k}\otimes\mathbf{x}_{k+1}\cdots\otimes\mathbf{x}_d\right)$$

has linearly independent columns for every $(\mathbf{x}_1,\ldots,\mathbf{x}_d)\in\mathbb{R}^{N_1+\cdots+N_d}$ *satisfying*

$$\|\mathbf{b}-A\left(\mathbf{x}_1\otimes\cdots\otimes\mathbf{x}_d\right)\|_2 \le \left\|\mathbf{b}-A\left(\mathbf{x}_1^0\otimes\cdots\otimes\mathbf{x}_d^0\right)\right\|_2\,. \tag{15}$$

Then every accumulation point $\left(\mathbf{x}_1^*,\ldots,\mathbf{x}_d^*\right)$ *of the sequence*

$$\{(\mathbf{x}_1^n,\ldots,\mathbf{x}_d^n)\}_{n=0}^{\infty}\,,$$

generated by Algorithm 2 using the map

$$\Phi\left(\mathbf{x}_1,\ldots,\mathbf{x}_d\right)=\|\mathbf{b}-A(\mathbf{x}_1\otimes\cdots\otimes\mathbf{x}_d)\|_2,$$

satisfies the equation $\nabla\Phi\left(\mathbf{x}_1^*,\ldots,\mathbf{x}_d^*\right)=\mathbf{0}$. *Moreover, assume that*

$$\mathbf{x}_1,\ldots,\mathbf{x}_{k-1},\mathbf{x}_{k+1},\ldots,\mathbf{x}_d,$$

are fixed for some $k\in\{1,2,\ldots,d\}$, *then*

$$\mathbf{x}_k=(Z_k^TZ_k)^{-1}Z_k^T\mathbf{b},$$

is the global minimum of the directional minimization problem

$$\min_{\mathbf{x}\in\mathbb{R}^{N_k}} \|\mathbf{b}-A\left(\mathbf{x}_1\otimes\cdots\otimes\mathbf{x}_{k-1}\otimes\mathbf{x}\otimes\mathbf{x}_{k+1}\otimes\cdots\otimes\mathbf{x}_d\right)\|_2\,. \tag{16}$$

ISSN: 1697-9761

Note that given a point $(\mathbf{x}_1, \ldots, \mathbf{x}_d)$, descend with respect to the coordinate $\mathbf{x}_k$ means, from Theorem 2, that we need to solve the standard least squares problem

$$\min_{\mathbf{x} \in \mathbb{R}^{N_k}} \|\mathbf{b} - Z_k \mathbf{x}\|_2 \,. \tag{17}$$

In particular we minimize Φ cyclically with respect to the coordinate variables Thus, Theorem 2 allow to us to solve the rank-one minimization problem (13) by means the Alternate Least Squares (ALS) Algorithm 3. We point out that for high-dimensional problems the numerical implementation of solving the equation

$$Z_k^T Z_k \mathbf{x} = Z_k^T \mathbf{x}, \tag{18}$$

can be a hardly task. However, if the matrix A can be represented also in separated representation form, then as the following corollary shows (18) can be implemented in a more easy way by using the properties of the Kronecker product.

Algorithm 3 ALS Algorithm

```
 1: procedure ALS($\mathbf{b}, A, \texttt{iter\_max}, \texttt{tol}$)
 2:     Initialize $\mathbf{x}_i^0$ for $i = 1, 2 \ldots, d$.
 3:     $\texttt{iter} = 1$
 4:     while $\texttt{iter} < \texttt{iter\_max}$ do
 5:         $\hat{\mathbf{x}}_k \leftarrow \mathbf{x}_k^0, k = 1, \ldots, d$
 6:         for $k = 1, 2, \ldots, d$ do
 7:             $Z = A(\mathbf{x}_1^0 \otimes \cdots \otimes \mathbf{x}_{k-1}^0 \otimes I_{N_k} \otimes \hat{\mathbf{x}}_{k+1} \otimes \cdots \otimes \hat{\mathbf{x}}_d)$
 8:             $\mathbf{x}_k^0 = (Z^T Z)^{-1} Z^T \mathbf{b}$
 9:         end for
10:         if $\prod_{k=1}^d \|\mathbf{x}_k^0 - \hat{\mathbf{x}}_k\|_2 < \texttt{tol}$ then goto 14
11:         end if
12:         $\texttt{iter} = \texttt{iter} + 1$
13:     end while
14:     return $\mathbf{x}^0 = (\mathbf{x}_1^0, \ldots, \mathbf{x}_d^0)$
15: end procedure
```

Corollary 3. *Assume that*

$$A = \sum_{j=1}^{r_A} A_1^j \otimes \cdots \otimes A_d^j,$$

where $A_i^j \in \mathbb{R}^{N_i \times N_i}$ *for* $i = 1, 2, \ldots, d$ *and* $j = 1, 2, \ldots, r_A$*. Let* $k \in \{1, 2, \ldots, d\}$ *and assume that for* $\mathbf{x}_1, \ldots, \mathbf{x}_{k-1}, \mathbf{x}_{k+1}, \ldots, \mathbf{x}_d$ *fixed, the* $(N_1 \cdots N_d) \times N_k$*-matrix*

$$Z_k = \sum_{j=1}^{r_A} A_1^{(j)} \mathbf{x}_1 \otimes \cdots \otimes A_{k-1}^{(j)} \mathbf{x}_{k-1} \otimes A_k^{(j)} \otimes A_{k+1}^{(j)} \mathbf{x}_{k+1} \cdot \cdots \otimes A_d^{(j)} \mathbf{x}_d,$$

has linearly independent columns. Then

$$\mathbf{x}_k^* = (Z_k^T Z_k)^{-1} Z_k^T \mathbf{b}. \tag{19}$$

is the global minimum of of the directional minimization problem (16).

This corollary implies that we can solve the minimization problem

$$\min_{(\mathbf{x}_1, \ldots, \mathbf{x}_d)} \left\| \mathbf{b} - \sum_{i=1}^{r_A} A_1^i \mathbf{x}_1 \otimes \cdots \otimes A_d^i \mathbf{x}_d \right\|_2 , \tag{20}$$

by means the ALS Algorithm 4.

Algorithm 4 ALS where A is given in separable form

```
 1: procedure ALS($\mathbf{b}, \sum_{i=1}^{r_A} A_1^i \otimes \cdots \otimes A_d^i$, iter_max, tol)
 2:     Initialize $\mathbf{x}_i^0$ for $i = 1, 2 \ldots, d$.
 3:     iter $= 1$
 4:     while iter $<$ iter_max do
 5:         $\hat{\mathbf{x}}_k \leftarrow \mathbf{x}_k^0, k = 1, \ldots, d$
 6:         for $k = 1, 2, \ldots, d$ do
 7:             $Z = \sum_{j=1}^{r_A} A_1^{(j)} \mathbf{x}_1^0 \otimes \cdots \otimes A_{k-1}^{(j)} \mathbf{x}_{k-1}^0 \otimes A_k^{(j)} \otimes A_{k+1}^{(j)} \hat{\mathbf{x}}_{k+1} \cdots \otimes A_d^{(j)} \hat{\mathbf{x}}_d$
 8:             $\mathbf{x}_k^0 = (Z^T Z)^{-1} Z^T \mathbf{b}$
 9:         end for
10:         if $\prod_{k=1}^d \|\mathbf{x}_k^0 - \hat{\mathbf{x}}_k\|_2 <$ tol then goto 14
11:         end if
12:         iter $=$ iter $+ 1$
13:     end while
14:     return $\mathbf{x}^0 = (\mathbf{x}_1^0, \ldots, \mathbf{x}_d^0)$
15: end procedure
```

3. A CASE STUDY I: THE FUNDAMENTAL THEOREM OF ASSET PRICING

Now, consider a financial market which contains N traded financial assets, whose prices at time $t = 0$ are denoted by

$$\mathbf{S}_0 = \begin{bmatrix} S_0^1 & S_0^2 & \cdots & S_0^N \end{bmatrix}^T \geq \mathbf{0}.$$

At time Δt, the owner of financial asset number i receives a random payment depending on the state of the world. We model this randomness by introducing a finite probability space $(\Omega, \mathcal{F}, \mathbb{P})$, where $\Omega = \{\omega_1, \omega_2, \ldots, \omega_M\}$, $\mathcal{F} = \mathcal{P}(\Omega)$ and $\mathbb{P}(\omega_i) > 0$ for all $i \in \{1, 2, \ldots, M\}$.

We note that the random payment arising from financial asset i is a $\mathbb{R}^M$–vector

$$\left[S_{\Delta t}^i(\omega_1), S_{\Delta t}^i(\omega_2), \ldots, S_{\Delta t}^i(\omega_M)\right]^T \geq \mathbf{0}.$$

At time $t = 0$ the agents can buy and sell financial assets. The portfolio position of an individual agent is given by a trading strategy, which is a vector

$$\boldsymbol{\theta} = [\theta_1, \theta_2, \ldots, \theta_N]^T \in \mathbb{R}^N.$$

Here θ_i denotes the quantity if the i–th asset is bought at time $t = 0$, which may be negative, if the agent has a short position, as well as positive, if he has a long position.

The dynamics of this model using the trading strategy $\boldsymbol{\theta}$ are as follows:

1. At time $t = 0$ the agent invests the amount

$$\mathbf{S}_0^T \boldsymbol{\theta} = \theta_1 S_0^1 + \theta_2 S_0^2 + \cdots + \theta_N S_0^N,$$

2. and at time $t = \Delta t$ the agent receives a random payment $\mathbf{P}$ that we can represent by using a matrix as follows, let

$$\mathbf{S}_{\Delta t} = \begin{bmatrix} S_{\Delta t}^1(\omega_1) & S_{\Delta t}^1(\omega_2) & \cdots & S_{\Delta t}^1(\omega_M) \\ S_{\Delta t}^2(\omega_1) & S_{\Delta t}^2(\omega_2) & \cdots & S_{\Delta t}^2(\omega_M) \\ \vdots & \vdots & \ddots & \vdots \\ S_{\Delta t}^N(\omega_1) & S_{\Delta t}^N(\omega_2) & \cdots & S_{\Delta t}^N(\omega_M) \end{bmatrix},$$

then

$$\mathbf{P} = \mathbf{S}_{\Delta t}^T \boldsymbol{\theta}.$$

We remark that each component of vector $\mathbf{P}$ represents the payment received depending on the realized state of the world ω.

Then we can define an *arbitrage opportunity* as a vector $\theta \in \mathbb{R}^N$ such that one of the following two conditions holds.

(Arb1) $\mathbf{S}_0^T \boldsymbol{\theta} = 0$ and $\mathbf{P} = \mathbf{S}_{\Delta t}^T \boldsymbol{\theta} \geq \mathbf{0}$, with $\mathbf{S}_{\Delta t}^T \boldsymbol{\theta} \neq \mathbf{0}$.

(Arb2) $\mathbf{S}_0^T \boldsymbol{\theta} < 0$ and $\mathbf{P} = \mathbf{S}_{\Delta t}^T \boldsymbol{\theta} \geq \mathbf{0}$.

Note that in the case of an arbitrage opportunity which satisfies (Arb 1) the agent's net investment at time $t = 0$ is zero, and there exists a $\omega \in \Omega$ such that

$$\sum_{i=1}^{N} S_{\Delta t}^{i} (\omega) \theta_i > 0,$$

that is, there exists non–zero probability to obtain a "free lunch". In the case of condition (Arb 2), we have that $\mathbf{S}_0^T \theta < 0$, that is, the agent borrows money for consumption at time $t = 0$, and he does not have to repay anything at the time Δt.

By using the well–known result called, *the Separating Hyperplane Theorem* (see Rockafellar (1990)) that is a version of the *Hahn–Banach Theorem* (see Sheldon Lin (1996)) the following result follows (see Bingham and Kiesel (1998), Duffie (1992), Pliska (1997) and Sheldon Lin (1996)).

Theorem 4. *There are no arbitrage opportunities if and only if there exists* $\boldsymbol{\Psi} > \mathbf{0}$ *such that*

$$\mathbf{S}_{\Delta t} \boldsymbol{\Psi} = \mathbf{S}_0. \tag{21}$$

We will say that a vector $\boldsymbol{\Psi} > \mathbf{0}$ satisfying (21) is a state price vector. Moreover, we can state that *the Separating Hyperplane Theorem implies the existence of a state price vector in the proof Theorem 4.*

Now, we can give some preliminary definitions and results about basic Linear Algebra. Let A be a $m \times n$–matrix, then we define the column space of A, that we denote by $\operatorname{col} A$, as

$$\operatorname{col} A = \operatorname{span} \{A\mathbf{e}_1, A\mathbf{e}_2, \ldots, A\mathbf{e}_n\},$$

where $\mathbf{e}_i$ denotes the i–th column of the $n \times n$ identity matrix. In particular, if we set

$$\mathbf{S}^i = \mathbf{S}_{\Delta t} \mathbf{e}_i$$

for $i = 1, 2, \ldots, k$, then

$$\operatorname{col} \mathbf{S}_{\Delta t} = \operatorname{span} \left\{ \mathbf{S}^1, \mathbf{S}^2, \ldots, \mathbf{S}^k \right\}.$$

In a similar way as above we define the row space of A, denoted by $\operatorname{row} A$, by

$$\operatorname{row} A = \operatorname{col} A^T.$$

Let

$$\operatorname{nul} A = \{\mathbf{x} : A\mathbf{x} = \mathbf{0}\},$$

and for a vectorial subspace $E \subset \mathbb{R}^n$, we will denote by $E^\perp$ the orthogonal complement of E, that is,

$$E^\perp = \left\{ \mathbf{x} : \mathbf{x}^T \mathbf{y} = \mathbf{0} \text{ for all } \mathbf{y} \in E \right\}.$$

It is well–known (see Strang (1998)) that

$$E \cap E^\perp = \{\mathbf{0}\}$$

and for all $\mathbf{x} \in \mathbb{R}^n$ there exist $\mathbf{x}_1 \in E$ and $\mathbf{x}_2 \in E^{\perp}$ such that

$$\mathbf{x} = \mathbf{x}_1 + \mathbf{x}_2.$$

Moreover,

$$(\operatorname{nul} A)^{\perp} = \operatorname{row} A = \operatorname{col} A^T.$$

Finally, set $\mathbb{K} = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{x} \geq \mathbf{0}\}$, $\overset{\circ}{\mathbb{K}} = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{x} > \mathbf{0}\}$ and for $\mathbf{x} \in \mathbb{R}^n$, let $\mathcal{Z}(\mathbf{x}) = \{i : x_i = 0\}$.

The following useful result, due to Acedo, Benito, Falcó, Rubia and Torres (2001), gives a computational approach to the Fundamental Theorem of Asset Pricing.

Theorem 5. *Let $\mathbf{\Psi}^*$ be a solution of*

$$\begin{array}{l} \min \|\mathbf{S}_{\Delta t}\mathbf{\Psi} - \mathbf{S}_0\|_2^2 \\ \textit{subject to } \mathbf{\Psi} \geq \mathbf{0}, \end{array} \tag{22}$$

and take the residual vector $\boldsymbol{\theta}^* = \mathbf{S}_{\Delta t}\mathbf{\Psi}^* - \mathbf{S}_0$. If $\boldsymbol{\theta}^* \neq \mathbf{0}$ then $\boldsymbol{\theta}^*$ satisfies (Arb 2). Otherwise, if $\theta^* = \mathbf{0}$ then one and only one of the following statement holds:

1. *If $\mathbf{\Psi}^* > \mathbf{0}$ then there are no arbitrage opportunities.*
2. *If $\mathbf{\Psi}^* \geq \mathbf{0}$ and*

$$\operatorname{span}\left\{\mathbf{S}^i : i \in \mathcal{Z}(\mathbf{\Psi}^*)\right\} \subset \operatorname{span}\left\{\mathbf{S}^i : i \notin \mathcal{Z}(\mathbf{\Psi}^*)\right\},$$

then there exist $\delta > 0$ and a continuous path of state price vectors $\mathbf{\Psi}_{\varepsilon}^*$, where $\varepsilon \in (0, \delta)$, and such that

$$\lim_{\varepsilon \to 0} \mathbf{\Psi}_{\varepsilon}^* = \mathbf{\Psi}^*.$$

Moreover, there are no arbitrage opportunities.

3. *If $\mathbf{\Psi}^* \geq \mathbf{0}$ and*

$$\operatorname{span}\left\{\mathbf{S}^i : i \in \mathcal{Z}(\mathbf{\Psi}^*)\right\} \not\subseteq \operatorname{span}\left\{\mathbf{S}^i : i \notin \mathcal{Z}(\mathbf{\Psi}^*)\right\},$$

then there are arbitrage opportunities which satisfy (Arb1) and there are no state price vectors. Moreover, let $\mathbf{y}^*$ be a solution of

$$\begin{array}{l} \min \|[\mathbf{S}_{\Delta t}, -\mathbf{S}_0]\,\mathbf{y} + [\mathbf{S}_{\Delta t}, -\mathbf{S}_0]\,\mathbf{e}\|_2^2 \\[1em] \textit{subject to } \mathbf{y} \geq \mathbf{0}, \end{array} \tag{23}$$

where $\mathbf{e} = [1, 1, \ldots, 1]^T$, then

$$\boldsymbol{\theta}^* = [\mathbf{S}_{\Delta t}, -\mathbf{S}_0]\,\mathbf{y}^* + [\mathbf{S}_{\Delta t}, -\mathbf{S}_0]\,\mathbf{e}$$

is an arbitrage opportunity.

From the above theorem we obtain the following result.

Corollary 6. *Theorem 5 implies Theorem 4.*

Now we would to construct an approximate solution of (22)

$$\mathbf{\Psi} \approx \widehat{\mathbf{\Psi}} = \sum_{i=1}^{r} \boldsymbol{\psi}_1^{(i)} \otimes \cdots \otimes \boldsymbol{\psi}_d^{(i)},$$

where $\boldsymbol{\psi}_j^{(i)} \in \mathbb{R}^{M_i}$ with $\boldsymbol{\psi}_j^{(i)} \geq \mathbf{0}$ for $1 \leq i \leq d$ and $1 \leq j \leq r$. To this end we introduce the following sets. Fix $\mathcal{S}_0^+ = \mathcal{S}_0$. Let $\mathbf{x} \in \mathbb{R}^{N_1 \cdots N_d}$ we say that $\mathbf{x} \in \mathcal{S}_1^+ = \mathcal{S}_1^+(N_1, N_2, \ldots, N_d)$ if

Algorithm 5 Non-negative ALS Algorithm

```
1:  procedure NALS($\mathbf{b}, A, r,$ iter_max, tol)
2:      Initialize $\mathbf{x}_i^j$ for $i = 1, 2, 3 \ldots, d$ and for $j = 1, 2, \ldots, r$.
3:      iter = 1
4:      while iter < iter_max do
5:          $\hat{\mathbf{x}}_i^j \leftarrow \mathbf{x}_i^j, i = 1, \ldots, d$ and $j = 1, 2, \ldots, r$.
6:          for $k = 1, 2, \ldots, d$ do
7:              $Z =$ zeros$(N_1 \cdots N_d, rN_k)$;
8:              for $j = 1, 2, \ldots, r$ do
9:                  $Z(:, (j-1)N_k + 1 : jN_k) = A(\mathbf{x}_1^j \otimes \cdots \otimes \mathbf{x}_{k-1}^j \otimes I_{N_k} \otimes \hat{\mathbf{x}}_{k+1}^j \otimes \cdots \otimes \hat{\mathbf{x}}_d^j)$
10:             end for
11:             $\left[(\mathbf{x}_k^1)^T \cdots (\mathbf{x}_k^r)^T\right]^T =$ lsqnonneg$(Z, \mathbf{b})$
12:         end for
13:         if $\prod_{j=1}^r \prod_{k=1}^d \|\mathbf{x}_k^j - \hat{\mathbf{x}}_k^j\|_2 <$ tol then goto 14
14:         end if
15:         iter = iter + 1
16:     end while
17:     return $\mathbf{x}^0 = (\mathbf{x}_1^0, \ldots, \mathbf{x}_d^0)$
18: end procedure
```

$\mathbf{x} = \mathbf{x}_1 \otimes_K \mathbf{x}_2 \otimes_K \cdots \otimes_K \mathbf{x}_d$, where $\mathbf{x}_i \in \mathbb{R}^{N_i}$, $\mathbf{x}_i \geq \mathbf{0}$, for $i = 1, \ldots, d$. Then $\mathcal{S}_r^+ = \mathcal{S}_{r-1}^+ + \mathcal{S}_1^+$ for $r \geq 1$.

Recently, Lim and Comon (2009), has been show that $\mathcal{S}_r^+$ is a closed set in any norm topology in $\mathbb{R}^{N_1 \cdots N_d}$. Thus, if $\mathbf{S}_{\Delta t}$ has full rank, then the problem

$$\min_{\mathbf{\Psi} \in \mathcal{S}_n^+} \|\mathbf{S}_{\Delta t}\mathbf{\Psi} - \mathbf{S}_0\|_2^2 \,, \tag{24}$$

is well posed for all $n \geq 1$. This allow to us to propose the Algorithm 5.

In Figure 1 we show the evolution of the logarithm of residual where we perform $d \times$ `iterations` of the NALS Algorithm 5 with $r = 20$, In this example we consider a square random generated matrix $\mathbf{S}_{\Delta t} \in \mathbb{R}^{585 \times 585}$ Here we consider that $N_1 = 5, N_2 = 9$ and $N_3 = 13$. The relative error was equal to $2.763592401388982e - 11$. A similar numerical experiment is show in Figure 2. In this case we consider a matrix $\mathbf{S}_{\Delta t} \in \mathbb{R}^{150 \times 585}$. In both examples $\mathbf{S}_0$ was randomly generated.

4. A CASE STUDY II : THE FIRST PASSAGE TIME AND THE POISSON EQUATION IN $(0,1)^D$

Our first case to study is based on the well-known Feynman-Kac representation of the solution to the Dirichlet problem for Poisson's equation. Recall that the Dirichlet problem for Poisson's equation is

$$\left\{ \begin{array}{c} -\Delta u = f \text{ in } \Omega \subset \mathbb{R}^d \\ u|_{\partial\Omega} = 0. \end{array} \right. \tag{25}$$

where $f = f(x_1, x_2, \ldots, x_d)$ is a given function and $\Delta = \sum_{i=1}^d \frac{\partial^2}{\partial x_i^2}$ is the Laplace operator. The solution of this problem at $\mathbf{x}_0 \in \mathbb{R}^d$, given in the form of the path-integral with respect to standard d-dimensional Brownian motion $\mathbf{W}_t$ is as follows

$$u(\mathbf{x}_0) = \mathbb{E}\left[\int_0^{\tau_{\partial\Omega}} 2f(\mathbf{W}_t)dt\right] \tag{26}$$

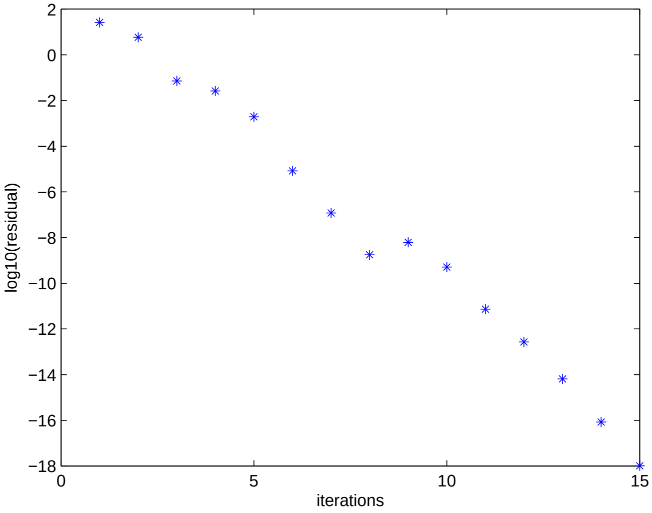

Figure 1. The convergence of the NALS Algorithm 5 with $r = 20$ for a $\mathbf{S}_{\Delta t} \in \mathbb{R}^{585\times 585}$ nonnegative square matrix.

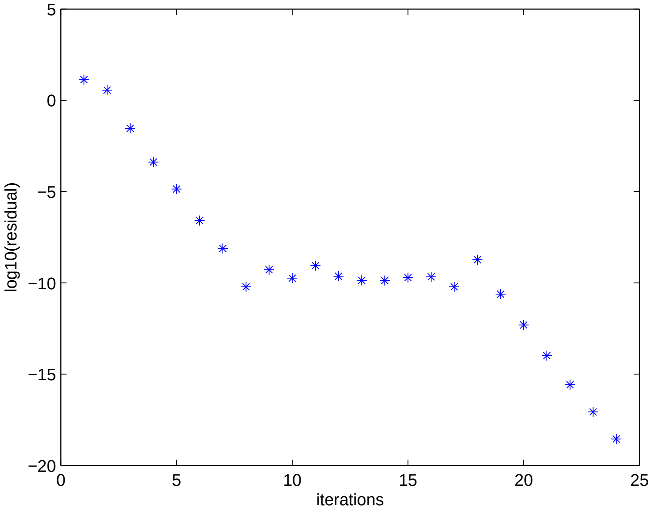

Figure 2. The convergence of the NALS Algorithm 5 with $r = 20$ for a $\mathbf{S}_{\Delta t} \in \mathbb{R}^{150\times 585}$ nonnegative matrix.

ISSN: 1697-9761

Here

$$\tau_{\partial\Omega} = \inf\{t : \mathbf{W}_t \in \partial\Omega\}$$

is the first-passage time and $\mathbf{W}_{\tau_{\partial\Omega}}$ is the first-passage location on the boundary, $\partial\Omega$. We assume that $\mathbb{E}[\tau_{\partial\Omega}] < \infty$ for all $\mathbf{x}_0 \in \Omega$ and f and u are continuous and bounded in Ω, and that the boundary, $\partial\Omega$, is sufficiently smooth so as to ensure the existence of a unique solution, $u(\mathbf{x})$, that has bounded and continuous first- and second-order partial derivatives in any interior subdomain

In order to find its variational formulation, we recall the following Green formula for the Laplacian:

$$-\int_\Omega \Delta u \; v \; d\mathbf{x} = \int_\Omega \nabla u \cdot \nabla v \; d\mathbf{x} - \int_{\partial\Omega} \frac{\partial u}{\partial n} \; v \; d\gamma. \tag{27}$$

Assume that $\Omega = (0,1)^d$. and for $d = 1, 2, \ldots$ let be the bilinear form

$$a_d(u; v) = \int_\Omega \nabla u \cdot \nabla v \; d\mathbf{x}.$$

Take $d = 3$ in (25) and then we can easily deduce that u satisfies the following problem: Find $u \in H_0^1(\Omega)$ such that

$$a_3(u; v) = \int_\Omega f \, v d\mathbf{x} \text{ for all } v \in H_0^1(\Omega). \tag{28}$$

The Galerkin approximation to (28) reads:

$$\text{find } u_h \in V_h : a_3(u_h; v_h) = \int_\Omega f \, v_h d\mathbf{x} \; \forall v_h \in V_h. \tag{29}$$

Assume $V_h = P \otimes P \otimes P$ where $P = \text{span}\{\omega^1, \ldots, \omega^N\}$ and $\omega^1, \ldots, \omega^N$ in $H_0^1(0,1)$ are the following N–linearly independent maps. First, we partitioned the interval $[0,1]$ into N-parts

$$0 = \widehat{x}_1 < \widehat{x}_2 < \ldots < \widehat{x}_{N+1} = 1.$$

Denote $h_i = \widehat{x}_{i+1} - \widehat{x}_i$ and $h = \max_{1 \le i \le N} h_i$. For $i = 1, 2, \ldots, N-1$, let

$$\omega^{(i)}(x) = \begin{cases} (x - \widehat{x}_i)/h_i & \widehat{x}_i \le x \le \widehat{x}_{i+1} \\ (\widehat{x}_{i+2} - x)/h_{i+1} & \widehat{x}_{i+1} \le x \le \widehat{x}_{i+2}, \\ 0 & \text{otherwise} \end{cases}$$

These functions are continuous and piecewise linear. It is easy to see that they are linearly independent. The first order weak derivatives of the basis functions are piecewise constant. Indeed for $i = 1, 2, \ldots, N-1$

$$\frac{d}{dx}\,\omega^{(i)}(x) = \begin{cases} 1/h_i & \widehat{x}_i \le x \le \widehat{x}_{i+1} \\ -1/h_{i+1} & \widehat{x}_{i+1} \le x \le \widehat{x}_{i+2}, \\ 0 & \text{otherwise} \end{cases}$$

We assume that we have an uniform partition, that is, $h_i = h$ for $i = 1, 2, \ldots, N$. Then the following formulas are useful

$$\int_0^1 \frac{d}{dx}\,\omega^{(i)}(x) \frac{d}{dx}\,\omega^{(i-1)}(x) dx = -\frac{1}{h} \text{ for } 2 \le i \le N-1, \tag{30}$$

$$\int_0^1 \left(\frac{d}{dx}\,\omega^{(i)}(x)\right)^2 dx = \frac{2}{h}, \text{ for } 1 \le i \le N-1, \tag{31}$$

$$\int_0^1 \omega^{(i)}(x)\,\omega^{(i-1)}(x) dx = \frac{h}{6} \text{ for } 2 \le i \le N-1, \tag{32}$$

$$\int_0^1 \left(\omega^{(i)}(x)\right)^2 dx = \frac{2h}{3}, \text{ for } 1 \le i \le N-1, \tag{33}$$

ISSN: 1697-9761

Now, let the stiffness multilinear matrix $\mathbb{A}_3$ defined by

$$\begin{aligned}
(\mathbb{A}_3)_{j_1,j_2,j_3;i_1,i_2,i_3} &= a_3(\omega^{(i_1)} \otimes \omega^{(i_2)} \otimes \omega^{(i_3)}; \omega^{(j_1)} \otimes \omega^{(j_2)} \otimes \omega^{(j_3)}) \\
&= \int_\Omega \left(\frac{\partial \omega^{(i_1)}}{\partial x_1} \frac{\partial \omega^{(j_1)}}{\partial x_1} \omega^{(i_2)} \omega^{(j_2)} \omega^{(i_3)} \omega^{(j_3)} \right. \\
&\quad + \frac{\partial \omega^{(i_2)}}{\partial x_2} \frac{\partial \omega^{(j_2)}}{\partial x_2} \omega^{(i_1)} \omega^{(j_1)} \omega^{(i_3)} \omega^{(j_3)} \\
&\quad \left. \frac{\partial \omega^{(i_3)}}{\partial x_2} \frac{\partial \omega^{(j_3)}}{\partial x_2} \omega^{(i_1)} \omega^{(j_1)} \omega^{(i_2)} \omega^{(j_2)} \right) d\mathbf{x} \\
&= A_{j_1;i_1} B_{j_2;i_2} B_{j_3;i_3} + B_{j_1;i_1} A_{j_2;i_2} B_{j_1;i_1} + B_{j_1;i_1} B_{j_2;i_2} A_{j_3;i_3},
\end{aligned}$$

where

$$A_{j,i} = a_1(\omega^{(i)}; \omega^{(j)}) \text{ and } B_{j;i} = \int_0^1 \omega^{(i)}(x) \omega^{(j)}(x) dx.$$

Then, by using (30)-(33), we have that

$$A = \begin{bmatrix} \frac{2}{h} & -\frac{1}{h} & & & \\ -\frac{1}{h} & \frac{2}{h} & -\frac{1}{h} & & \\ \ddots & \ddots & \ddots & & \\ & & -\frac{1}{h} & \frac{2}{h} & -\frac{1}{h} \\ & & & -\frac{1}{h} & \frac{2}{h} \end{bmatrix} \text{ and } B = \begin{bmatrix} \frac{2h}{3} & \frac{h}{6} & & & \\ \frac{h}{6} & \frac{2h}{3} & \frac{h}{6} & & \\ \ddots & \ddots & \ddots & & \\ & & \frac{h}{6} & \frac{2h}{3} & \frac{h}{6} \\ & & & \frac{h}{6} & \frac{2h}{3} \end{bmatrix}.$$

Finally, we obtain that

$$(\mathbb{A}_3)_{j_1,j_2,j_3;i_1,i_2,i_3} = (A \otimes B \otimes B + B \otimes A \otimes B + B \otimes B \otimes A)_{s,t},$$

by using

$$s = j_3 + \sum_{l=1}^{2} \left[(j_l - 1) N^{3-l} \right] \text{ and } t = i_3 + \sum_{l=1}^{2} \left[(i_l - 1) N^{3-l} \right]. \tag{34}$$

On the other hand we have that

$$\int_\Omega f\, (\omega^{(j_1)} \otimes \omega^{(j_2)} \otimes \omega^{(j_3)}) d\mathbf{x} \quad = \quad \mathcal{F}_{j_1,j_2,j_3}.$$

In order to solve (29) we consider a map in V_h given by

$$u_h = \sum_{i_1,i_2,i_3} \mathcal{U}_{i_1,i_2,i_3}\, \omega^{(i_1)} \otimes \omega^{(i_2)} \otimes \omega^{(j_3)},$$

where $\mathcal{U}_{i_1,i_2,i_3} \approx u(\widehat{x}_{i_1+1}, \widehat{x}_{i_2+1}, \widehat{x}_{i_3+1})$. Thus,

$$\mathcal{U}_{k,i_2,i_3} = \mathcal{U}_{i_1,k,i_3} = \mathcal{U}_{i_1,i_2,k} = 0$$

for $k \in \{0, N\}$. Substitute u_h in (29) with $v_h = \omega^{(j_1)} \otimes \omega^{(j_2)} \otimes \omega^{(j_3)}$. Then we obtain that

$$\sum_{i_1,i_2,i_3=1}^{N-1} (\mathbb{A}_3)_{j_1,j_2,j_3;i_1,i_2.i_3} \mathcal{U}_{i_1,i_2,i_3} = \mathcal{F}_{j_1,j_2,j_3} \tag{35}$$

must be hold for all $j_1, j_2, j_3 \in \{1, \ldots, N-1\}$. This implies that (29) is equivalent to solve the following linear system

$$(A \otimes B \otimes B + B \otimes A \otimes B + B \otimes B \otimes A)\mathbf{u} = \mathbf{f}, \tag{36}$$

ISSN: 1697-9761

here we consider that a general multi-index tensor can be represented by a standard vector by using that

$$\mathcal{V}_{j_1,j_2,\ldots,j_d} = \mathbf{v}_s$$

if and only if

$$s = j_d + \sum_{l=1}^{d-1} \left[(j_l - 1) N^{d-l} \right].$$

For a general $d \geq 2$ it can be shown that in order to solve numerically (25) we need to solve the following linear system:

$$\left(\sum_{j=1}^{d} A_1^{(j)} \otimes \cdots \otimes A_d^{(j)} \right) \mathbf{u} = \mathbf{f}. \tag{37}$$

where

$$A_k^{(j)} = \begin{cases} A & \text{if} \quad k = j, \\ B & \text{if} \quad k \neq j. \end{cases}$$

Finally, it is not difficult to show that A and B are symmetric and definite positive matrices. Then, from the properties 7-8 listed at the end of Section 1, $\sum_{j=1}^{d} A_1^{(j)} \otimes \cdots \otimes A_d^{(j)}$ is a symmetric and definite positive matrix. In consequence, it is invertible.

Example 7. *Firstly, we consider the following problem in 3D: Solve for*

$$(x_1, x_2, x_3) \in \Omega = (0,1)^3 :$$

$$-\Delta u = (2\pi)^2 \cdot 3 \cdot \sin(2\pi x_1 - \pi) \sin(2\pi x_2 - \pi) \sin(2\pi x_3 - \pi), \tag{38}$$

$$u|_{\partial\Omega} = 0, \tag{39}$$

which has as closed form solution

$$u(x_1, x_2, x_3) = \sin(2\pi x_1 - \pi) \sin(2\pi x_2 - \pi) \sin(2\pi x_3 - \pi).$$

We used the separable representation Algorithm 1) with parameter values `iter_max` $= 5$, `rank_max` $= 1000$ *and* $\varepsilon = 0.001$. *The algorithm give us an approximated solution* $\mathbf{u}_1 \in \mathcal{S}_1$. *In Figure 3 we represent the relative error of the solution computed using the separable representation algorithm, using logarithmic scale, as a function of the number of nodes used in the discretization of the Poisson equation. All the computations were performed using the GNU software* OCTAVE *in a AMD 64 Athlon K8 with 2Gib of RAM.*

In Figure 4 we represent the CPU time, in logarithmic scale, used in solving the linear system (37) against the separable representation algorithm. In both cases all the linear systems involved were solved using the standard linear system solver (`A\b`*) of* OCTAVE.

Example 8. *Finally we are addressing some highly multidimensional models. To this end we solve numerically (25) for* $(x_1, \ldots, x_d) \in \Omega = (0,\pi)^d$ *where*

$$f = \sum_{k=1}^{d} -(1+k) \sin^{(-1+k)}(x_k) \left(-k \cos^2(x_k) + \sin^2(x_k) \right) \prod_{k'=1, k' \neq k}^{d} \sin^{(1+k')}(x_{k'}),$$

which has as closed form solution

$$u(x_1, \ldots, x_d) = \prod_{k=1}^{d} \sin^{(k+1)}(x_k).$$

Here we consider the true solution $\mathbf{u}$ *given by* $\mathcal{U}_{i_1,\ldots,i_d} = u(\widehat{x}_{i_1+1}, \ldots, \widehat{x}_{i_d+1})$. *For* $d = 10$ *we use the parameter values* `iter_max` $= 2$, `rank_max` $= 10$ *and* $\varepsilon = 0.001$. *In a similar way as above the algorithm give us an approximated solution* $\widehat{\mathbf{u}} \in \mathcal{S}_1$. *In Figure 5 we represent the absolute error* $\|\widehat{\mathbf{u}} - \mathbf{u}\|_2$ *as a function of* $h = \pi/N$ *for* $N = 5, 10, 20, \ldots, 160$ *in* $\log_{10}$*-scale. By using similar parameters values the problem has been solved for* $d = 100$ *in about 20 minutes.*

ISSN: 1697-9761

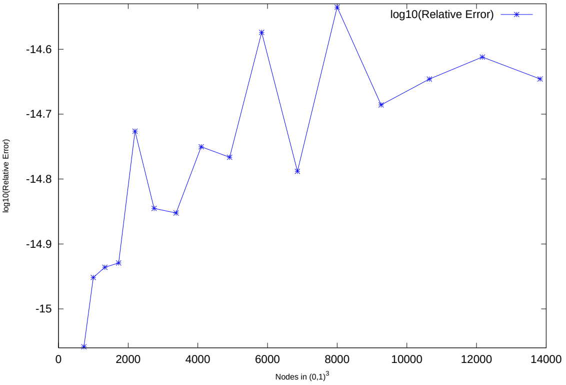

Figure 3. The relative error $\|\mathbf{u}_1 - A^{-1}\mathbf{f}\|_2 / \|A^{-1}\mathbf{f}\|_2$ in logarithmic scale.

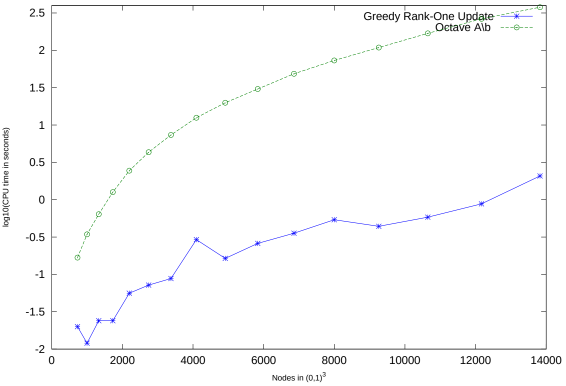

Figure 4. The CPU time, in seconds, used in solving the linear system as a function of the number of nodes employed in the discretization of the Poisson Equation.

5. FINAL COMMENTS AND REMARKS

This work has allowed defining a general form of multilinear systems in a separable representation form. In particular, we use the fact that tensors of order 3 or higher have best rank-1 approximation. This fact allows us to propose an iterative method based on the so-called enrichment-projection technique. As we can show the method runs under very weak conditions, recall that we only use the assumption that the linear system has a an invertible matrix. However, its efficiency depends strongly on the matrix form (symmetric, tridiagonal, full, sparse, ...). Also, an important issue is related to the stopping criterion in the enrichment process.

ACKNOWLEDGEMENTS

The author would thank to Alejandro Balbás his support and helpful comments along these last years, and to Paco Chinesta because with him I established many collaborations with highly

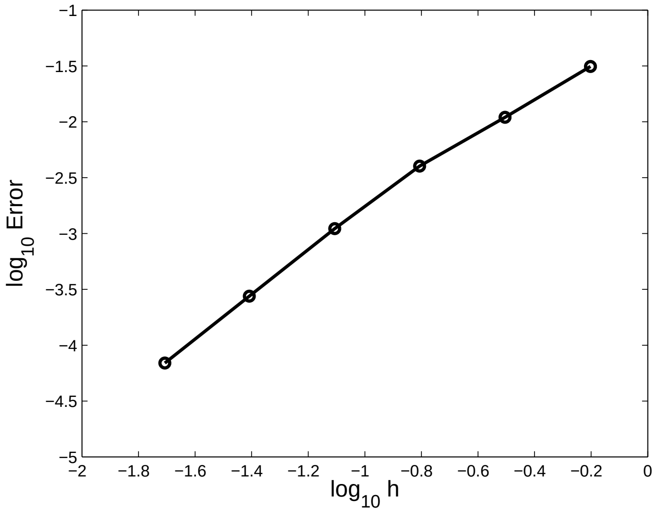

Figure 5. The absolute error $\|\widehat{\mathbf{u}}-\mathbf{u}\|_2$ as a function of $h = \pi/N$ for $N = 5, 10, 20, \ldots, 160$ in $\log_{10}$-scale.

recognized scientists, allowing the proposal of new numerical strategies. Among them I would mention Elias Cueto, Amine Ammar, Anthony Nouy, David Ryckelynck, Alexei Lozinski and many others that deserve my acknowledgments.

REFERENCES

Acedo F., Benito F., Falcó A., Rubia A. and Torres J, 2001. *A computational Approach to the fundamental Theorem of Asset Pricing in a Single Period Market*, Computational Economics 18:233-249.

Ammar A., Mokdad B., Chinesta F., and Keunings R, 2006. A new family of solvers for some classes of multidimensional partial differential equations encountered in Kinetic Theory modelling Complex Fluids, Journal of Non–Newtonian Fluid Mechanics, 139, pp. 153-176.

Ammar A., Mokdad B., Chinesta F., and Keunings R, 2007. A new family of solvers for some classes of multidimensional partial differential equations encountered in Kinetic Theory modelling Complex Fluids. Part II: Transient Simulations using Space-Time Separated Representations, Journal of Non-Newtonian Fluid Mechanics, 144, pp. 98-121

Ammar A., Chinesta F., and Falcó A, 2009. On the convergence of a Greedy Rank-One Update Algorithm for a class of linear systems. To appear in Archives of Comput. Meth. In Engrg.

Bellman R, 1961. Adaptive Control Processes: A Guided Tour, Princeton University Press, Princeton, NJ.

G. Beylkin and M. J. Mohlenkamp, 2005. Algorithms for Numerical Analysis in High Dimensions. SIAM J. Sci. Comput. Vol 26, No. 6. pp. 2133–2159.

Bingham N.H., Kiesel, R, 1998. Pricing and hedging Financial Derivatives, Springer Finance, Springer–Verlag.

Bungartz H.J., Griebel M, 2004. Sparse grids, Acta Numerica, 13, 1-123.

Chinesta F., Ammar A., Joyot P, 2008. The nanometric and micrometric scales of the structure and mechanics of materials revisited: An introduction to the challenges of fully deterministic numerical descriptions. International Journal for Multiscale Computational Engineering, 6/3, 191-213.

ISSN: 1697-9761

Chinesta F., Ammar A., Falcó., and Laso M, 2007. On the reduction of stochastic kinetic theory models of complex fluids. Modelling and Simulation in Materials Science and Engineering, 15 pp. 639-652.

de Silva V., and Lim L.-H, 2008. Tensor Rank and Ill-posedness of the best low-rank approximation problem, SIAM Journal of Matrix Analysis & Appl. Vol. 30, no 3, pp. 1084–1127.

Duffie, D, 1992. Dynamic Asset Pricing Theory, Princeton University Press.

A. Graham, 1991. Kronecker Products and Matrix Calculus with Applications. John Wiley.

Ladeveze P, 1999. Non linear computational structural mechanics, Springer, N.Y.

Ladeveze P., Passieux J.Ch., Neron D, 2009. The LATIN multiscale computa- tional method and the proper orthogonal decomposition, Computer Methods in Applied Mechanics and Engineering, In Press.

Le Bris C., Lelièvre T., and Maday Y, 2009. Results and Questions on a Nonlinear Approximation Approach for Solving High-dimensional Partial Differential Equations. Constructive Approximation, Volume 30, Number 3 pp. 621-651.

Lim L.-H. and Comon P, 2009. Nonnegative approximations of nonnegative tensors, Journal of Chemometrics, 23, no. 7-8, pp. 432-441.

Nouy A, 2009.Recent developments in spectral stochastic methods for the solution of Stochastic Partial Differential Equations, Archives of Comput. Meth. In Engrg., Volume 16, Number 3, pp. 251-285.

Pliska, S. R, 1997.Introduction to Mathematical Finance: Discrete Time Models, Blackwell Publishers, Oxford.

Rockafellar, Rm, 1990.Convex Analysis, Princeton University Press.

Sheldon Lin, X, 1996.Lectures Notes in Mathematical Finance, Departament of Statistics & Actuarial Science, University of Iowa. http://www.stat.uiowa.edu/˜shlin/teaching.html

Sreenath S.N., Cho K-H, Wellstead P, 2009. Modelling the dynamics of signalling pathways, Essays in biochemistry, 45, 128.

Strang G, 1998. Introduction to Linear Algebra, Wellesley–Cambridge Press.

C. F. Van Loan, 2000. The ubiquitous Kronecker product, J. Comput. Appl. Math. 123, pp. 85-100.

ISSN: 1697-9761