Recent advances on the use of separated representations — Source link

David González, Amine Ammar, Francisco Chinesta, Elías Cueto

Institutions: University of Zaragoza, Joseph Fourier University, École centrale de Nantes

Published on: 31 Jul 2009 - International Journal for Numerical Methods in Engineering (Wiley)

Topics: Degrees of freedom

Recent Advances in the Use of Separated Representations

David Gonzalez, Amine Ammar, Francisco Chinesta, Elías Cueto

Recent advances on the use of separated representations

David Gonzalez[1], Amine Ammar[2], Francisco Chinesta[3] and Elıas Cueto[1]

[1]*Group of Structural Mechanics and Materials Modelling, Aragón Institute of Engineering Research (I3A), University of Zaragoza, María de Luna, 5, Campus Rio Ebro, E-50018 Zaragoza, Spain*
[2]*Laboratoire de Rhéologie, UMR 5520, Université Joseph Fourier, 1301 Rue de la piscine, Domaine universitaire, 38041 Grenoble Cedex, France*
[3]*EADS Corporate International Chair, GEM Ecole Centrale de Nantes, BP 92101, F-44321 Nantes cedex 3, France*

Separated representations based on finite sum decompositions constitute an appealing strategy for reducing the computer resources and the calculation costs by reducing drastically the number of degrees of freedom that the functional approximations involve (the number of degrees of freedom scale linearly with the dimension of the space in which the model is defined instead of the exponential growing characteristic of mesh-based discretization strategies). In our knowledge the use of separated representations is the only possibility for circumventing the terrific curse of dimensionality related to some highly multidimensional models involving hundreds of dimensions, as we proved in some of our former works. Its application is not restricted to multidimensional models, obviously separated representation can also be applied in standard 2D or 3D models, allowing for high resolution computations. Because its early life numerous issues persist, many of them attracting the curiosity of many research groups within the computational mechanics community. In this paper we are focusing in two issues never until now addressed: (i) the imposition of non-homogenous essential boundary conditions and (ii) the consideration of complex geometries.

KEY WORDS: curse of dimensionality; separation of variables; singular value decomposition; essential boundary conditions

1. INTRODUCTION

Some models encountered in science and engineering are sometimes defined in multidimensional spaces (as the ones involved in quantum mechanics or kinetic theory descriptions of materials, including complex fluids) that exhibit the terrific *curse of dimensionality* when usual mesh-based discretization techniques are applied.

In those models the difficulty is quite natural and their solution needs for new strategies. One possibility lies in the use of sparse grids [1] but its use is restricted to models defined in spaces of moderate dimensions. Another technique able to circumvent, or at least alleviate, the curse of dimensionality consists of using a separated representation of the unknown field as we proposed in our former works [2, 3] and then applied in numerous contexts: (i) quantum chemistry [4]; (ii) Brownian dynamics [5]; (iii) kinetic theory description of polymers solutions and melts [6]; (iv) kinetic theory descriptions of rods suspensions [7]; among many others.

Basically, the separated representation of a generic function $u(x_1,\ldots,x_D)$ (also known as finite sums decomposition) writes:

$$u(x_1,\ldots,x_D) \approx \sum_{i=1}^{i=N} F_1^i(x_1) \times \cdots \times F_D^i(x_D) \tag{1}$$

This kind of representation is not new, it was widely employed in the last decades in the framework of quantum chemistry. In particular the Hartree–Fock (that involves a single product of functions) and post-Hartree–Fock approaches (as the MCSCF that involves a finite number of sums) made use of a separated representation of the wavefunction [8]. In the context of computational mechanics a similar decomposition was proposed, that was called radial approximation and that was applied for separating the space and time coordinates in thermomechanical models [9].

This kind of approximation only needs a technique able to construct, in a way completely transparent for the user, the separated representation of the unknown field involved in a partial differential equation (PDE).

The technique that we proposed for computing the different functions involved in Equation (1) consists of an alternating directions linearization strategy that we summarize here. For the sake of clarity, and without any loss of generality, we restrict our discussion to the D-dimensional Poisson's equation

$$\Delta u = -f(x_1, x_2, \ldots, x_D) \tag{2}$$

where u is a scalar function of $(x_1, x_2, \ldots, x_D)$. Problem (2) is defined in the domain $(x_1, x_2, \ldots, x_D) \in \Omega = (-L, +L)^D$ with vanishing essential boundary conditions.

The problem solution can be written in the form

$$u(x_1, x_2, \ldots, x_D) = \sum_{j=1}^{\infty} \alpha_j \prod_{k=1}^{D} F_{kj}(x_k) \tag{3}$$

where F_{kj} is the jth basis function, with unit norm, which only depends on the kth coordinate.

It is well known that the solution of numerous problems can be accurately approximated using a finite (sometimes very reduced) number (N) of approximation functions, i.e.:

$$u(x_1, x_2, \ldots, x_D) \approx \sum_{j=1}^{N} \alpha_j \prod_{k=1}^{D} F_{kj}(x_k) \tag{4}$$

The previous expression implies the same number of approximation functions in each dimension, but each one of these functions could be expressed in a discrete form using different number of parameters (nodes of the 1D grids).

Now, an appropriate numerical procedure is needed for computing the coefficients α_j as well as the N approximations functions in each dimension.

The proposed numerical scheme consists of an iteration procedure that solves at each iteration n the following three steps:

Step 1: *Projection of the solution in a discrete basis*. If we assume the functions $F_{kj}(\forall j \in [1, \ldots, n]; \forall k \in [1, \ldots, D])$ known (verifying the boundary conditions), the coefficients α_j can be computed by introducing the approximation of u into the Galerkin variational formulation associated with Equation (2):

$$\int_\Omega \nabla u^* \cdot \nabla u \, \mathrm{d}\Omega = \int_\Omega u^* f \, \mathrm{d}\Omega \tag{5}$$

Introducing the approximation of u and u^*

$$u(x_1, x_2, \ldots, x_D) = \sum_{j=1}^{n} \alpha_j \prod_{k=1}^{D} F_{kj}(x_k) \tag{6}$$

and

$$u^*(x_1, x_2, \ldots, x_D) = \sum_{j=1}^{n} \alpha_j^* \prod_{k=1}^{D} F_{kj}(x_k) \tag{7}$$

we have

$$\int_\Omega \nabla \left(\sum_{j=1}^{n} \alpha_j^* \prod_{k=1}^{D} F_{kj}(x_k) \right) \cdot \nabla \left(\sum_{j=1}^{n} \alpha_j \prod_{k=1}^{D} F_{kj}(x_k) \right) \mathrm{d}\Omega = \int_\Omega \left(\sum_{j=1}^{n} \alpha_j^* \prod_{k=1}^{D} F_{kj}(x_k) \right) f \, \mathrm{d}\Omega \tag{8}$$

Now, we assume that $f(x_1, \ldots, x_D)$ can be written in the form

$$f(x_1, \ldots, x_D) \approx \sum_{h=1}^{m} \prod_{k=1}^{D} f_{kh}(x_k) \tag{9}$$

Equation (8) involves integrals of a product of D functions each one defined in a different coordinate. Let $\prod_{k=1}^{D} g_k(x_k)$ be one of these functions to be integrated. The integral over Ω can be performed by integrating each function in its definition interval and then multiplying the D computed integrals according to

$$\int_\Omega \prod_{k=1}^{D} g_k(x_k) \, \mathrm{d}\Omega = \prod_{k=1}^{D} \int_{-L}^{L} g_k(x_k) \, \mathrm{d}x_k \tag{10}$$

which makes possible the numerical integration in highly dimensional spaces.

Now, due to the arbitrariness of the coefficients α_j^*, Equation (8) allows to compute the n-approximation coefficients α_j, solving the resulting linear system of size $n \times n$. This problem is linear and moreover rarely exceeds the order of tens of degrees of freedom. Thus, even if the resulting coefficient matrix is densely populated, the time required for its solution is negligible with respect to the one required for performing the approximation basis enrichment (step 3).

Step 2: *Checking convergence*. From the solution of u at iteration n given by Equation (6) we compute the residual Re related to Equation (2):

$$\mathrm{Re} = \frac{\sqrt{\int_{\Omega}(\Delta u + f(x_1, \ldots, x_D))^2}}{\|u\|} \tag{11}$$

If Re$<\varepsilon$ (epsilon is a small enough parameter) the iteration process stops, yielding the solution $u(x_1, \ldots, x_D)$ given by Equation (6). Otherwise, the iteration procedure continues.

The integral in Equation (11) can be written as the product of one-dimensional integrals by performing a separated representation of the square of the residual.

Step 3: *Enrichment of the approximation basis*. From the coefficients α_j just computed the approximation basis can be enriched by adding the new function $\prod_{k=1}^{D} F_{k(n+1)}(x_k)$. For this purpose we solve the non-linear Galerkin variational formulation related to Equation (2)

$$\int_{\Omega} \nabla u^* \cdot \nabla u \, \mathrm{d}\Omega = \int_{\Omega} u^* f \, \mathrm{d}\Omega \tag{12}$$

using the approximation of u given by:

$$u(x_1, x_2, \ldots, x_D) = \sum_{j=1}^{n} \alpha_j \prod_{k=1}^{D} F_{kj}(x_k) + \prod_{k=1}^{D} R_k(x_k) \tag{13}$$

The weighting function can be expressed as:

$$u^*(x_1, x_2, \ldots, x_D) = R_1^*(x_1) \times R_2(x_2) \times \cdots \times R_D(x_D) + \cdots + R_1(x_1) \times R_2(x_2) \times \cdots \times R_D^*(x_D) \tag{14}$$

This leads to a non-linear variational problem, whose solution allows to compute the D functions $R_k(x_k)$. Functions $F_{k(n+1)}(x_k)$ are finally obtained by normalizing, after convergence of the non-linear solver, the functions $R_1, R_2, \ldots, R_D$.

To solve this problem we introduce a discretization of those functions $R_k(x_k)$. Each one of these functions is approximated using a 1D finite element (FE) description. If we assume than p_k nodes are used to construct the interpolation of function $R_k(x_k)$ in the interval $[-L, L]$, then the size of the resulting discrete non-linear problem is $\sum_{k=1}^{k=D} p_k$. The price to pay for avoiding a whole mesh in the multidimensional domain is the solution of a non-linear problem. However, even in high dimensions the size of the non-linear problems remains moderate and no particular difficulties have been found in its solution up to hundreds dimensions. Concerning the computation time, even when the non-linear solver converges quickly, this step consumes the main part of the global computing time.

Different non-linear solvers have been analyzed: fixed-point, Newton or one based on an alternating directions scheme. In this work the last strategy was retained. Thus, in the enrichment step, function $R_1^{s+1}(x_1)$ is updated by assuming known all the others functions (given at the previous iteration of the non-linear solver $R_2^s(x_2), \ldots, R_D^s(x_D)$). Then, functions $R_1^{s+1}(x_1), R_3^s(x_3), \ldots, R_D^s(x_D)$ are assumed known for updating function $R_2^{s+1}(x_2)$, and so on until updating the last function $R_D^{s+1}(x_D)$. Now the convergence is checked by calculating $\sum_{i=1}^{i=D} \|R_i^{s+1}(x_i) - R_i^s(x_i)\|^2$. If this norm is small enough we can define the functions $F_{k(n+1)}(x_k)$ by normalizing the functions $R_1, R_2, \ldots, R_D$ and come back to step 1. On the contrary, if this norm is not small enough, a new iteration of the non-linear solver should be performed by updating functions $R_i^{s+2}(x_i),\ i = 1, \ldots, D$

and then checking again the convergence. Despite its simplicity, our experience proves that this strategy is in fact very robust.

We must recall that the technique that we proposed in the papers just referred, is not a universal strategy able to solve any kind of multidimensional PDE. Thus, the efficient application of the technique that we just described requires the separability of all the fields involved in the model. Obviously, this separability is not always possible because some functions need a tremendous number of sums. On the other hand, even when the field is separable (one could perform this separation by invoking, for example, the singular value decomposition (SVD) or the multidimensional SVD) the finite sums decomposition of general multidimensional functions is not realistic because the amount of memory needed for storing the discrete form of such functions before applying the multidimensional SVD.

In many physical models (see for example [2, 4]), a fully separation (consisting of a sum of products of one-dimensional functions) could not be envisaged from a practical point of view. Thus, a better approximation lies in writing

$$u(\mathbf{x}_1,\ldots,\mathbf{x}_d)\approx\sum_{i=1}^{i=N} F_1^i(\mathbf{x}_1)\times\cdots\times F_D^i(\mathbf{x}_d) \tag{15}$$

where the different functions taking part in the finite sums decomposition are defined in spaces of moderate dimensions, that is $\mathbf{x}_i\in\Omega_i\subset\mathbb{R}^{q_i}$, where usually $q_i=1$, 2 or 3.

When the model is separable, as it is the case in many physical models coming from computational science and engineering, the unknown field involved in the model can be expressed as a tensor product of approximations defined in lower-dimensional domains, eventually one-dimensional. However in that case, two major difficulties appear: (i) the first one is related to the enforcement of non-homogeneous essential boundary conditions; and (ii) the representation of geometrically complex domains other than cubes or hyper-cubes. This paper concerns these two difficulties that never until now, up to our knowledge, have been addressed. For the sake of clarity (and mainly for the ease of representation of the results), and without loss of generality, in what follows we restrict our analysis to 2D or 3D models.

The simplest way to enforce non-homogeneous essential boundary conditions (i.e. $u=u_g$ in $\partial\Omega$) consists in defining a function $\hat{u}(\mathbf{x})$, $\mathbf{x}\in\Omega\subset\mathbb{R}^D$, with the requested regularity, verifying the boundary conditions, i.e. $\hat{u}(\mathbf{x}\in\partial\Omega)=u_g(\mathbf{x}\in\partial\Omega)$. Now, one could perform the separation of $\hat{u}$, by invoking the SVD or the multidimensional SVD, leading to

$$\hat{u}(x_1,\ldots,x_D)\approx\sum_{i=1}^{i=M} G_1^i(x_1)\times\cdots\times G_D^i(x_D) \tag{16}$$

that allows defining the approximation of the unknown field as

$$u(x_1,\ldots,x_D)\approx\sum_{i=1}^{i=M} G_1^i(x_1)\times\cdots\times G_D^i(x_D)+\sum_{i=1}^{i=N} F_1^i(x_1)\times\cdots\times F_D^i(x_D) \tag{17}$$

where the second part of the right member $\sum_{i=1}^{i=N} F_1^i(x_1)\times\cdots\times F_D^i(x_D)$ is subjected to homogenous boundary conditions.

Finally, in some cases, even when the separability is assured, the geometrical complexity of the domain needs for special treatments. Different possibilities could be envisaged. This paper explores a very simple strategy based on the use of R-functions.

Sections 2 and 3 focus in the treatment of non-homogeneous essential boundary conditions and complex geometries, respectively, within the separated representation framework. Finally, Section 4 presents some numerical examples proving the efficiency of the strategies proposed.

2. IMPOSITION OF NON-HOMOGENEOUS ESSENTIAL BOUNDARY CONDITIONS

Consider, for simplicity and without loss of generality, the following problem:

$$-\Delta u = f(x_1, x_2, \ldots, x_D) \quad \text{in } \Omega = (0, L)^D \tag{18}$$

subjected to the boundary conditions:

$$u = u_g \neq 0 \quad \text{on } \Gamma \equiv \partial\Omega \tag{19}$$

Let us assume that we are able to find a function ψ, continuous in $\overline{\Omega}$, such that $-\Delta\psi \in L_2(\Omega)$ verifying Equation (19) [10]. Then, the solution of the problem given by Equations (18) and (19) can be obtained straightforwardly by

$$u = \psi + z \tag{20}$$

where we thus face a problem in the z variable

$$-\Delta z = f + \Delta\psi \quad \text{in } \Omega \tag{21}$$

$$z = 0 \quad \text{on } \Gamma \tag{22}$$

solvable by the method of separation of variables before presented.

2.1. *Construction of the function ψ*

We thus take an extension of the essential boundary conditions, u_g, ψ, within the domain Ω. The solution will be constructed in the form

$$u \approx \psi + \sum_{j=1}^{Q} \alpha_j \prod_{k=1}^{D} F_{kj}(x_k) \tag{23}$$

thus achieving the exact imposition of essential boundary conditions. In the last equation, $F_{kj}(x_k)$ is a function of the approximation space of the solution with homogeneous boundary conditions.

In practice, the function u_g can be defined at different portions of the boundary, through different functions ψ_i. Assuming the domain where the model is defined given by the hypercube $(0, L)^D$, essential boundary conditions must be enforced on all the $(D-1)$-dimensional faces of the hypercube: the two limit points of a 1D segment, the four faces of a 2D square, the six faces of a 3D cube, and so on. Thus, the number of faces of a D-dimensional hypercube results $P = 2D$ on which different essential boundary conditions ψ_i are assumed prescribed.

$$\begin{aligned} u(\Gamma_1) &= \psi_1 \\ u(\Gamma_2) &= \psi_2 \\ &\vdots \\ u(\Gamma_P) &= \psi_P \end{aligned} \tag{24}$$

where Γ_i represent the different $(D-1)$-dimensional faces.

Each one of these functions ψ_i can be grouped to form a single, global, function ψ utilizing, for instance, the transfinite interpolation method proposed in [11]. Each portion of the essential boundary $\partial\Omega_i$ can be described by an implicit function $w_i(x_1, x_2, \ldots, x_{i-1}, x_{i+1}, \ldots, x_D) = 0$, where w_i is a function vanishing on $\partial\Omega_i$ and positive elsewhere (the distance function, for instance, can be considered as a suitable w_i function). Thus,

$$\psi = \frac{\sum_{i=1}^{P} \psi_i \prod_{j=1;\, j\neq i}^{P} w_j}{\sum_{i=1}^{P} \prod_{j=1;\, j\neq i}^{P} w_j} \tag{25}$$

constitutes a generalization of Lagrangian interpolation. The resulting function ψ interpolates each ψ_i along each w_i.

Following this procedure, the weak form of the problem will be: Find $z \in H^1(\Omega)$ such that for every $z^* \in H_0^1(\Omega)$

$$\int_\Omega \nabla z^* \cdot \nabla z \, \mathrm{d}\Omega = \int_\Omega z^* f \, \mathrm{d}\Omega + \int_\Omega \nabla z^* \cdot \nabla \psi \, \mathrm{d}\Omega \tag{26}$$

holds.

Remark 1
The proposed form for the function ψ, given by Equation (25) is singular at some points of the boundary. Provided that numerical integration has been performed in this work through standard, first order, Gauss quadrature rules, no integration point will be placed in these singular positions. There exist, however, some alternative expressions for the function ψ that posses singular points at different positions. The interested reader can consult their form in reference [12] and references therein.

Finally, the obtained function ψ, which is generally not product or sum separable, according to [13], is separated approximately by employing the SVD technique. To this end, we employ singular values up to a given precision, which is normally fixed to $10^{-5}\|\psi\|$.

3. PROBLEMS DEFINED OVER GENERAL DOMAINS

The proposed solution strategy is based on the method of the characteristic function [14]. Let us consider a function $\Phi : \Omega^{\square} \to \mathbb{R}$, with $\Omega^{\square}$ a hypercube (usually the smallest one) containing the arbitrarily-shaped domain Ω, and with continuous derivatives such that:

$$\begin{aligned} \Phi(\mathbf{x}) &> 0, \quad \mathbf{x} \in \Omega \\ \Phi(\mathbf{x}) &= 0, \quad \mathbf{x} \in \partial\Omega \\ \Phi(\mathbf{x}) &= 0, \quad \mathbf{x} \notin \overline{\Omega} \\ |\nabla\Phi(\mathbf{x})| &\geqslant \alpha > 0, \quad \mathbf{x} \in \partial\Omega \end{aligned} \tag{27}$$

The solution space will then be $\mathscr{S}_h^{\Phi} = \{u \mid u = \Phi v, \ \ v \in H^1(\Omega^{\square})\} \subset H_0^1(\Omega^{\square})$.

The approximate solution to the problem given by Equations (18)–(19) will be $u_h \in \mathscr{S}_h^{\Phi}$, where u_h is the solution of the problem:

Find $u \in \mathscr{S}_h^{\Phi}$ such that for every $w \in H_0^1(\Omega^{\square})$

$$\int_{\Omega^{\square}} \nabla w \cdot \nabla u \, \mathrm{d}\Omega = \int_{\Omega^{\square}} w f \, \mathrm{d}\Omega \tag{28}$$

and $\mathscr{S}_h^{\Phi} = \Phi \mathscr{S}_h$, with $\mathscr{S}_h$ the finite-dimensional trial space, that depends on the particular method employed.

A study of the error and convergence associated with this method can be found in [14] and references therein, whereas the demonstration of the completeness of the method is accomplished in [12].

The final form of the solution in the context of a separated representation will then be:

$$u \approx \Phi \cdot \left[\sum_{j=1}^{Q} \alpha_j \prod_{j=1}^{D} F_{kj}(x_k) \right] \tag{29}$$

which will verify straightforwardly the homogeneous essential boundary conditions enforced along the boundary.

Remark 2
If the problem under consideration possesses non-homogeneous essential boundary conditions and is defined in a domain other than an hypercube at the same time, the obvious choice for the approximation combines the techniques developed in Sections 2 and 3 above:

$$u \approx \psi + \Phi \cdot \left[\sum_{j=1}^{Q} \alpha_j \prod_{j=1}^{D} F_{kj}(x_k) \right] \tag{30}$$

3.1. Construction of the function Φ

It is clear that this approach is not very useful unless a method to easily construct such function Φ in a general case is established. In this case we employ R-functions [11].

An R-function is a real-valued function whose sign is completely defined by the sign of its arguments. Such functions encode boolean operations that help to construct the combinations of simple, basis, functions. For instance, consider the following functions [11], that behave like the logical operators *and* and *or*:

$$x \wedge y \equiv (x + y - \sqrt{x^2 + y^2}) \tag{31}$$

$$x \vee y \equiv (x + y + \sqrt{x^2 + y^2}) \tag{32}$$

Consider, for instance, the domain shown in Figure 1. The domain is defined by means of a set of six inequalities of the general form

$$\begin{aligned} f_1 &= y \geqslant y_1 \\ f_2 &= x \geqslant x_2 \\ f_3 &= y \geqslant y_3 \\ &\cdots \end{aligned} \tag{33}$$

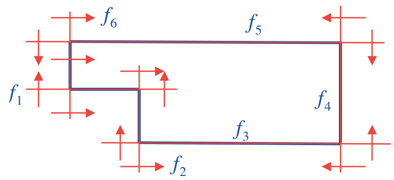
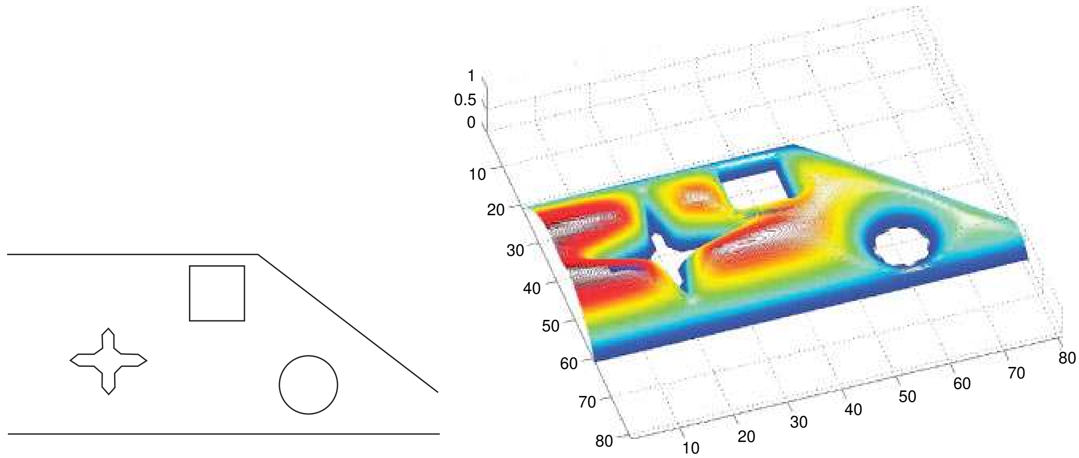


Figure 1. Definition of the boundary of a given domain by a set of six inequalities.



(a) (b)

Figure 2. Essential boundary of a given domain (a) and associated R-function (b).

These inequalities can be combined to render a definition of the whole domain as:

$$\Omega = (f_1 \vee f_2) \wedge f_3 \wedge f_4 \wedge f_5 \wedge f_6 \tag{34}$$

An example of the performance of this technique is shown in Figure 2, where the essential boundary of the domain is defined (Figure 2(a)) by black lines and the resulting R-function is depicted (Figure 2(b)).

The question now lies in establishing a separated representation for the function Φ. By employing standard SVD we can employ up to m singular values of the decomposition, $\Phi_{11} \times \cdots \times \Phi_{1D}, \ldots, \Phi_{m1} \times \cdots \times \Phi_{mD}$, necessary to obtain a prescribed precision. Thus, the final expression for the approximate solution of the problem will be

$$u \approx [\Phi_{11}(x_1) \times \cdots \times \Phi_{1D}(x_D) + \cdots + \Phi_{m1}(x_1) \times \cdots \times \Phi_{mD}(x_D)] \times \left[\sum_{j=1}^{N} \alpha_j \prod_{j=1}^{D} F_{kj}(x_k) \right] \tag{35}$$

The difficulty in this case is that a large number of terms in the SVD decomposition of the function Φ would lead to a complex expression for the approximate solution with a large number of terms in the sum. Each term would be integrated in its own 1D space through standard Gauss quadratures. Thus, finally, the above difficulty can be easily circumvented.

4. NUMERICAL EXAMPLES

In this section we describe some simple numerical experiments for illustrating the potential of the discretization technique based on the use of separated representations.

4.1. *A first example*

Let us consider the problem

$$\Delta u = 0 \quad \text{in} \ (0, 1)^3 \tag{36}$$

with

$$u = x \quad \text{on} \ \partial\Gamma \equiv \Omega$$

In this case, essential boundary conditions can be expressed in the form given by Equation (24) as

$$\psi_1 = 0 \quad \text{on} \ \Gamma_1 = (0, y, z)$$

$$\psi_6 = 1 \quad \text{on} \ \Gamma_6 = (1, y, z)$$

$$\psi_i = x \quad \text{on the other four faces of the cube}$$

With the help of the technique developed before, the obtained ψ function is shown in Figure 3 that corresponds to the exact solution $\psi = x$.

The weak form of the problem looks:

$$\int_\Omega \nabla z^* \cdot \nabla z \, \mathrm{d}\Omega = \int_\Omega z^* f \, \mathrm{d}\Omega + \int_\Omega \nabla z^* \cdot \nabla \psi \, \mathrm{d}\Omega = 0 \tag{37}$$

Solving the problem in variable z gives the trivial solution $z = 0$ and thus we have

$$u = \psi + z = x + 0 = x \tag{38}$$

which coincides with the analytical solution of the problem, as expected.

4.2. *A second, more elaborated, problem*

Let us consider now the problem

$$\Delta u = 0 \quad \text{in} \ (-1, 1)^2 \tag{39}$$

subjected to the boundary condition

$$u = x^2 + y^2 \quad \text{on} \ \partial\Omega$$

that in this case do not verify the PDE (39).

In this case, with the help of the approximation proposed in Equation (25), we obtain the following function verifying essential boundary conditions:

$$\psi = \frac{-2 + x^4 + y^4}{-2 + x^2 + y^2} \tag{40}$$

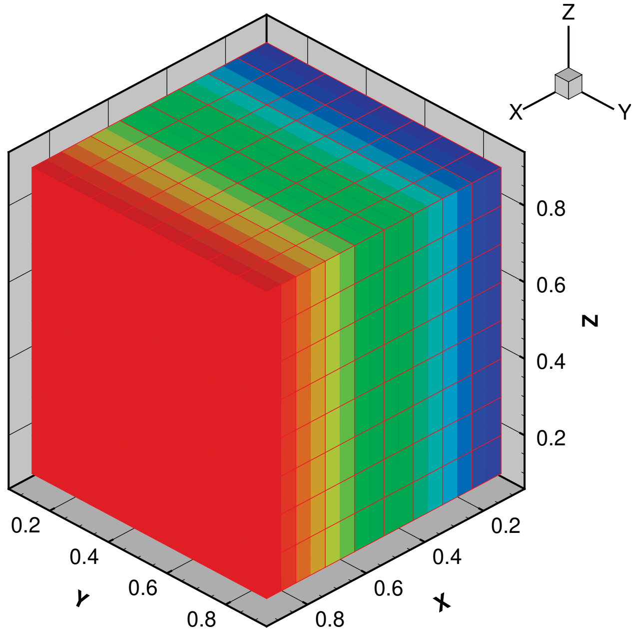


Figure 3. Function ψ obtained.

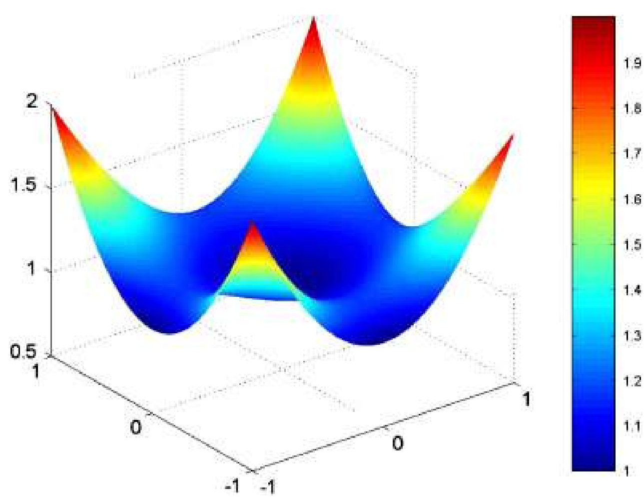


Figure 4. Function ψ for the second problem, Equation (39).

which is represented in Figure 4. This function cannot be written as a closed sums decomposition according to [13]. Applying SVD, we arrive to the one-dimensional functions depicted in Figure 5, that approximate the function ψ to the chosen level of precision, $10^{-5}\|\psi\|$ in this case.

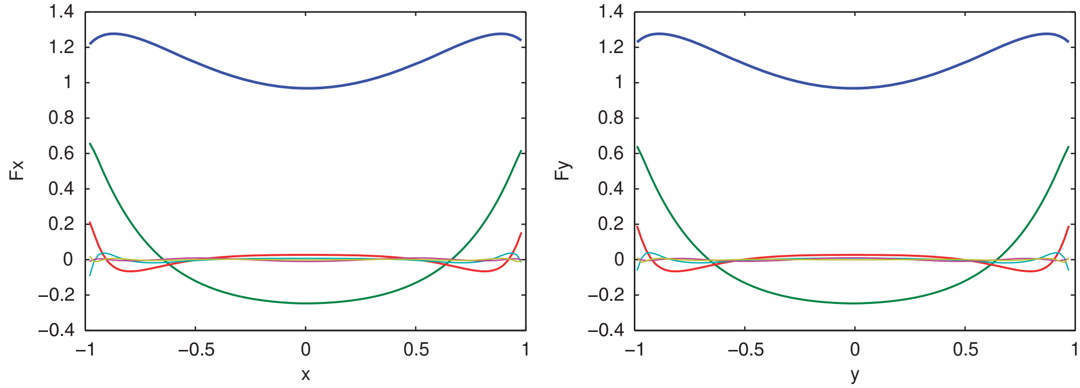


Figure 5. Singular value decomposition of ψ up to the precision $10^{-5}\|\psi\|$.

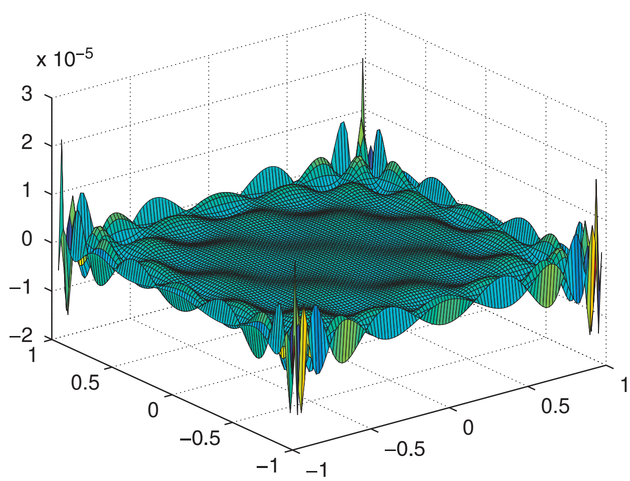


Figure 6. Error in the approximation of ψ by employing the first six SVD values.

The error generated in this separation, by employing the first six singular values of the decomposition, is depicted in Figure 6.

The method before described was applied in meshes composed by 15, 20, 30, 50, 80 and 130 nodes along each spatial direction. The solution was compared with one approximate solution obtained by standard FEs in a mesh composed by 40 000 nodes. The obtained solution, that will be considered as the reference one, is depicted in Figure 7.

The solution strategy can be resumed as follows. For each discretization, ranging from 15 nodes to 130 nodes at each spatial direction, we have employed the same SVD decomposition of function ψ, constructed to approximate the function to an error of 10^{-5}. The separated representation strategy

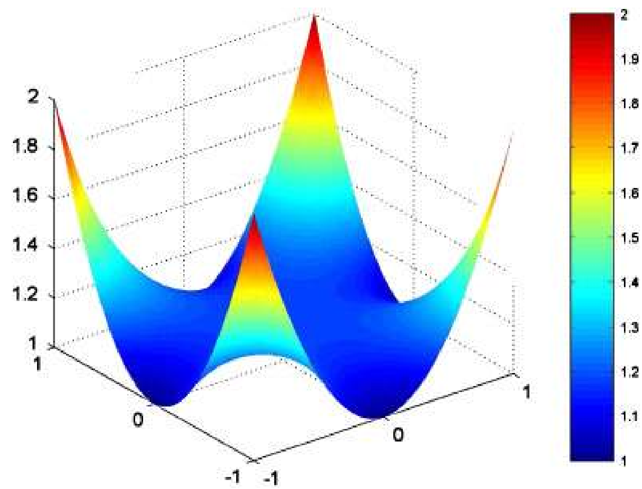


Figure 7. Solution obtained by standard FEM, assumed as the reference solution of the problem.

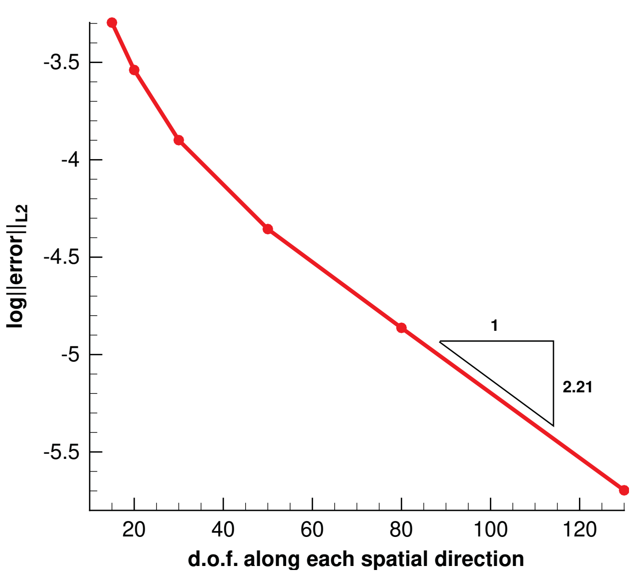


Figure 8. Solution convergence obtained by the proposed technique.

presented in Sections 1 and 2 has been employed until reaching a small enough residual. The obtained error for each discretization along each spatial direction is depicted in Figure 8.

The final solution for the separation of variables technique is depicted in Figure 9. The homogeneous part of the solution, i.e. the solution for the problem in z variable, is shown in Figure 10.

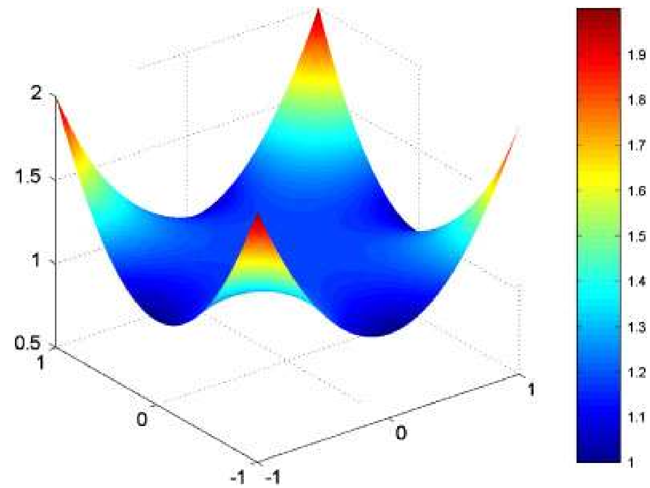


Figure 9. Solution obtained by the proposed technique. Mesh consisting of 80 nodes in each direction.

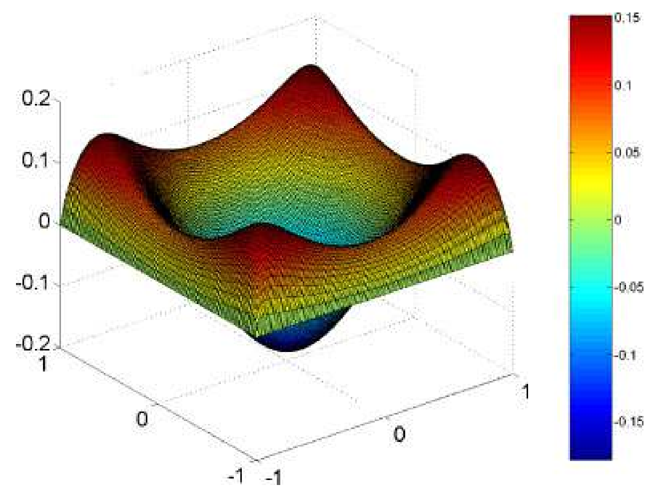


Figure 10. Solution obtained by the proposed technique for the homogeneous problem in z variable. Mesh of 80 nodes.

4.3. Extension of the proposed technique to higher-dimensional problems

The main issue associated with the straightforward extension of the proposed technique to higher-dimensional problems lies in obtaining the appropriate counterpart of the SVD decomposition in high dimensions. To this end, we have employed the technique implemented by Kolda and Bader [15], in conjunction with the MATLAB tensor toolbox [16]. In general, we seek for

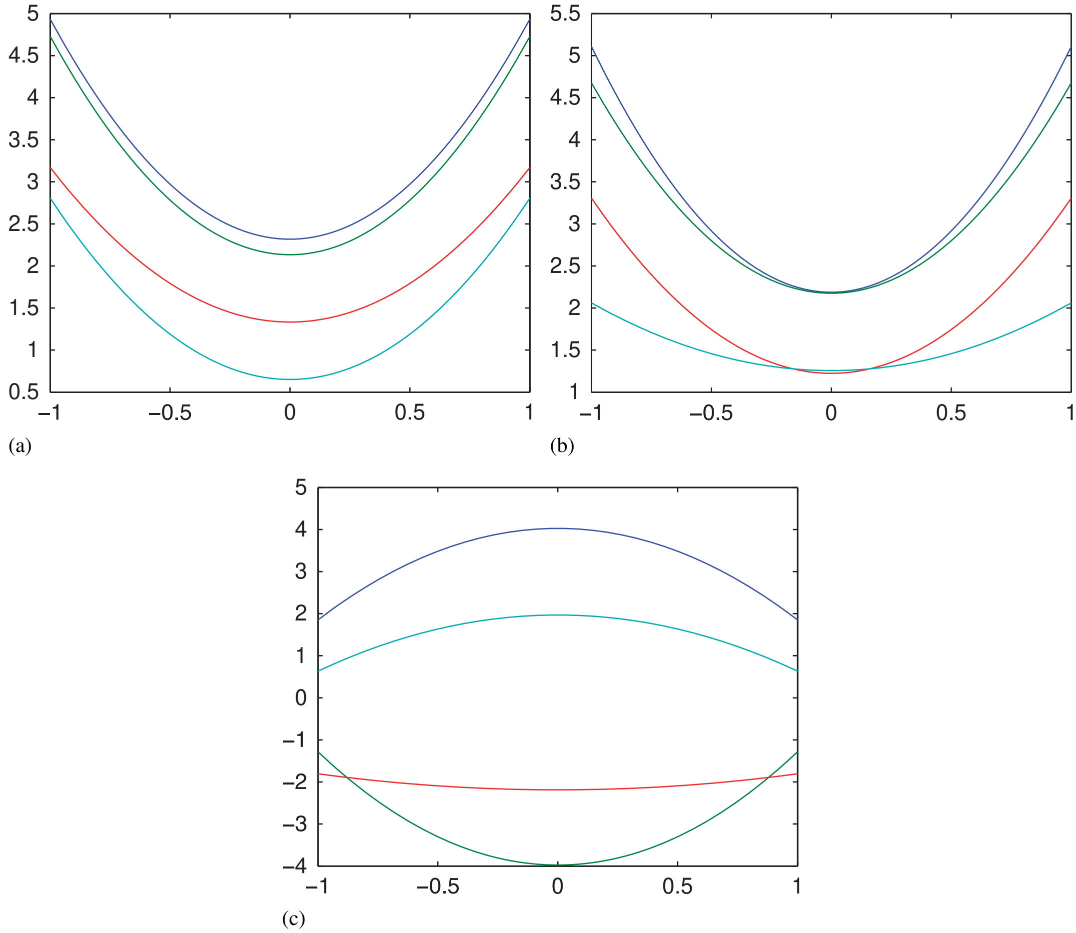


Figure 11. Functions obtained along x (a), y (b) and z (c) spatial directions for the boundary conditions of the problem stated in Equation (43).

decompositions of our function ψ, whose nodal values are stored in an order n ($n=3$ in the example here considered, for simplicity of graphical representation) tensor $\mathbf{A}$, in the form

$$\mathbf{A}=\mathbf{\Sigma}\times_1\mathbf{U}\times_2\mathbf{V}\times_3\mathbf{W}\times_4\ldots \tag{41}$$

where $\mathbf{\Sigma}$ is a diagonal tensor and $\mathbf{U}$, $\mathbf{V}$, $\mathbf{W}$, ... are the components or factors of the decomposition. Columns of the factors are orthonormal in standard, two-dimensional SVD, but this is not possible, in general, for tensors of order three or higher. The products involved in the previous decomposition are defined as follows. Mode-n multiplication, denoted by $\times_n$, means to multiply the matrix times the tensor in the nth mode. In other words, the n-mode product of a tensor $\mathbf{C}\in\mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$ by a matrix $\mathbf{U}\in\mathbb{R}^{J_n\times I_n}$ is an $I_1\times I_2\times\ldots I_{n-1}\times J_n\times I_{n+1}\times\cdots\times I_N$-tensor whose components

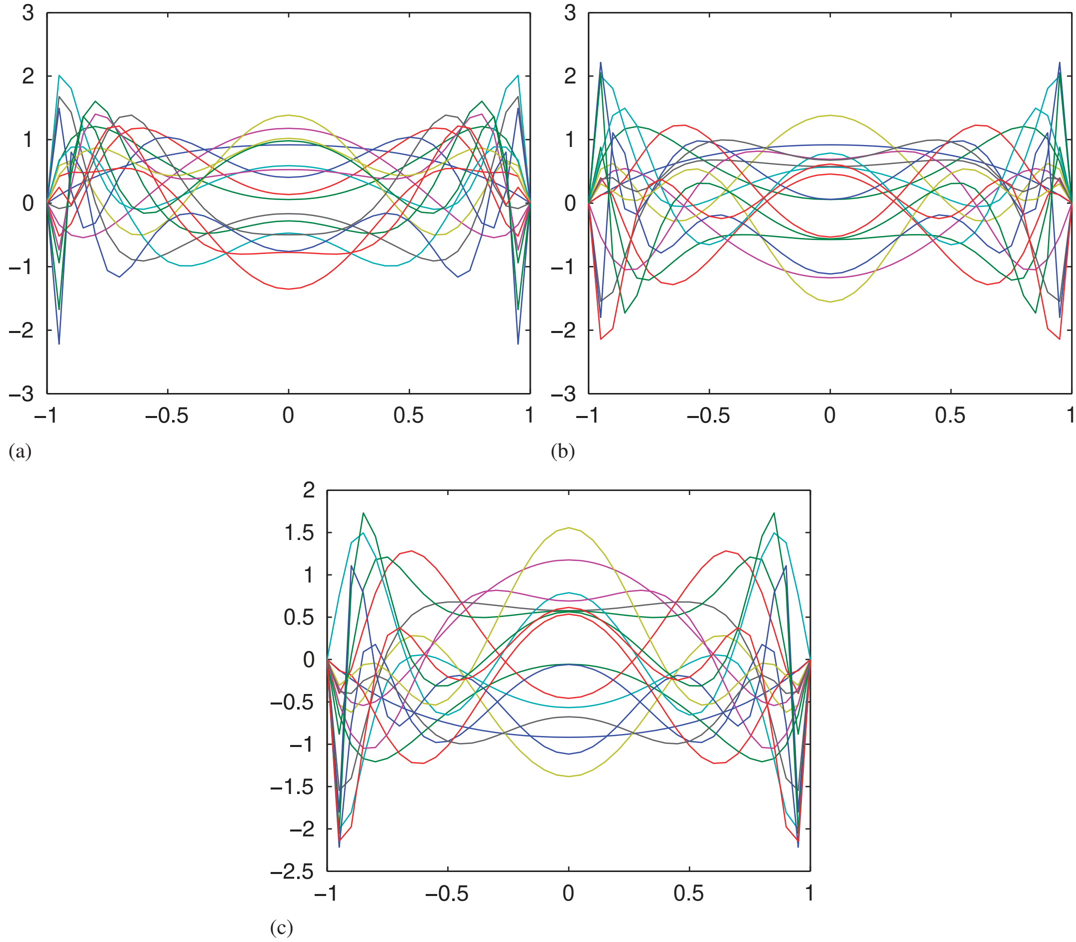


Figure 12. Functions obtained along x (a), y (b) and z (c) spatial directions for the problem stated in Equation (43).

are given by

$$(\mathbf{C} \times_n \mathbf{U})_{i_1 i_2 \ldots i_{n-1} j_n i_{n+1} \ldots i_N} = \sum_{i_n} c_{i_1 i_2 \ldots i_{n-1} i_n i_{n+1} \ldots i_N} u_{j_n i_n} \tag{42}$$

In this work we have employed the PARAFAC (Parallel Factors) decomposition, also known as CANDECOMP or Canonical Decomposition, provided by the MATLAB tensor toolbox [16]. This technique provides a diagonal tensor $\mathbf{\Sigma}$, suitable for the purpose of this work. The r most relevant entries of this tensor are employed in the decomposition, in order to accomplish with a user-predefined tolerance in the representation of the function ψ.

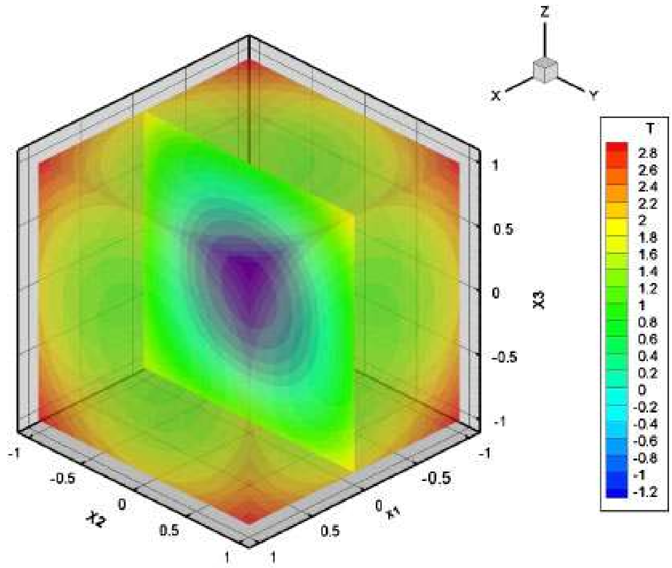


Figure 13. Solution obtained by the proposed technique for the problem with non-homogeneous boundary conditions in three dimensions. Mesh of 40 elements along each direction.

To analyze the behavior of the technique, we have extended the previous problem to three-dimensional settings

$$\Delta u = 0 \quad \text{in} \ \ (-1,1)^3 \tag{43}$$

subjected to the boundary condition

$$u = x^2 + y^2 + z^2 \quad \text{on} \ \partial\Omega$$

The higher-order SVD decomposition of the function ψ up to the tolerance $10^{-6}\|\psi\|$ provides the functions depicted in Figure 11(a)–(c). Only four functions along each spatial direction were necessary.

With these functions thus calculated, we let the method run to obtain the solution to the problem as depicted along each direction in Figure 12. The global, three-dimensional, solution to the problem is depicted in Figure 13. In this case, 18 functions were necessary along each spatial direction, that is a sum decompositions consisting of 18 sums, to reach the before-mentioned accuracy. This figure also depicts the trace of the solution at the plane $x = 0$.

4.4. A numerical model involving 10^9 *degrees of freedom*

In order to show the potential of the proposed method, we have analyzed the problem

$$\Delta u = 2(1-y^4)(1-z^6) + 12(1-x^2)y^2(1-z^6) + 30(1-x^2)(1-y^4)z^4 \quad \text{in} \ \ (-1,1)^3 \tag{44}$$

(the same analyzed in [2]), subjected to the boundary conditions

$$u = 0 \quad \text{on} \ \partial\Omega$$

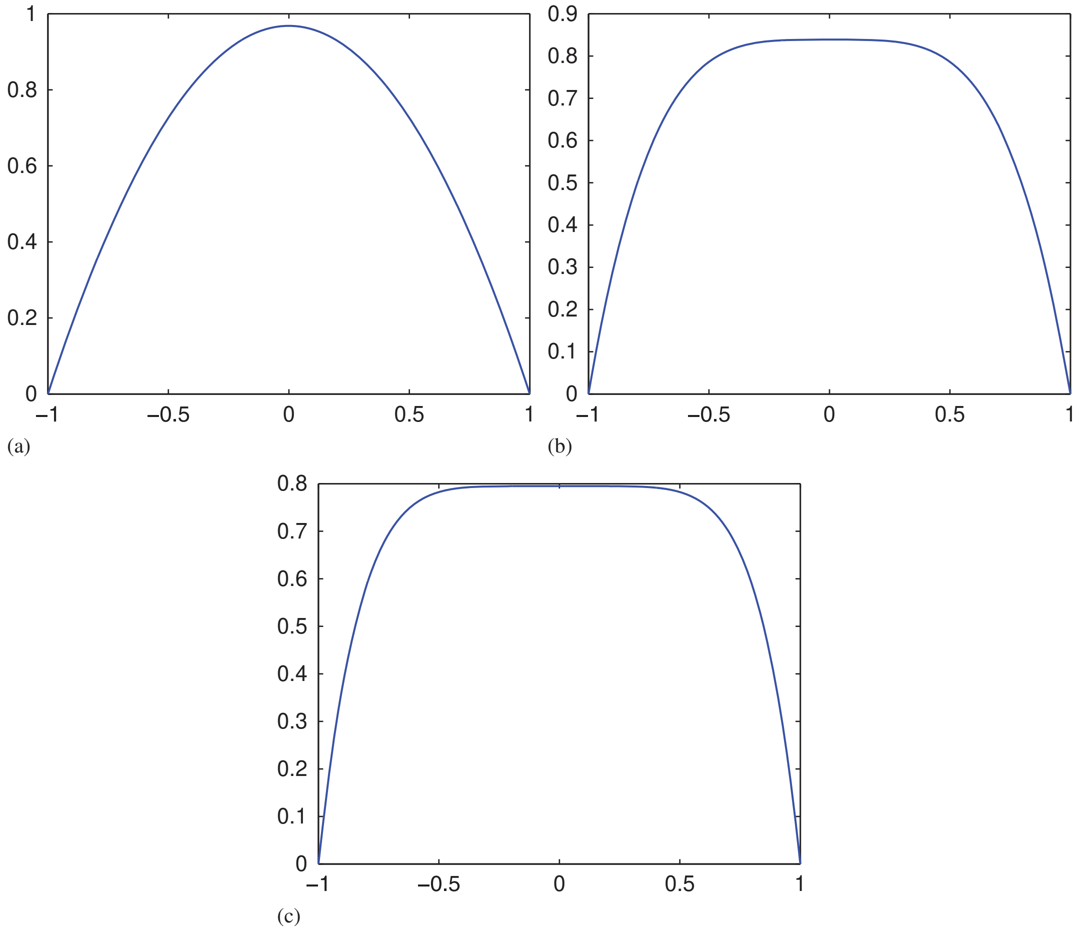


Figure 14. Functions obtained along x (a), y (b) and z (c) spatial directions for the problem stated in Equation (44).

The analytical solution to this problem is given by the function $u=(1-x^2)(1-y^4)(1-z^6)$. The solution has been approximated by using a separated representation where each one-dimensional function along each coordinate was approximated by using standard piecewise linear FE shape functions on a grid consisting of 1001 nodes on each direction.

The different computed one-dimensional functions are depicted in Figure 14(a)–(c). As expected an accurate enough solution was obtained after only one iteration, as was the case in [2], because the particular form of the exact solution that consists of a single functional product.

The solution obtained is represented in Figure 15, where derefinement has been applied to the results by employing a coarser mesh (for obvious reasons, it is difficult to depict the results on a mesh of 10^9 nodes). The error obtained is depicted in Figure 16, where a slice on an arbitrary section of the domain has been performed for clarity.

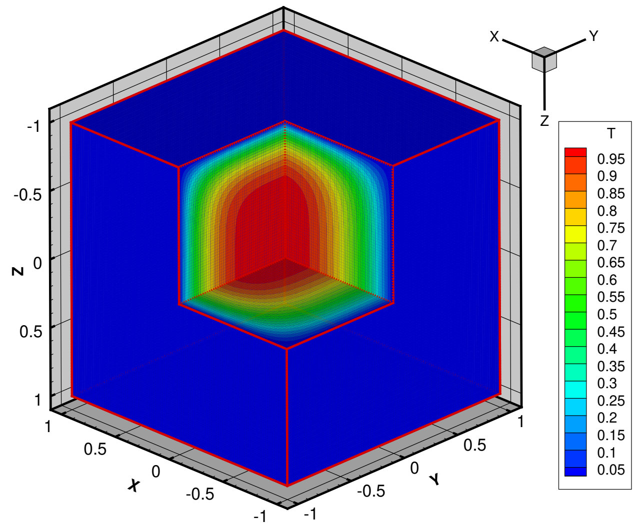


Figure 15. Solution of the problem stated in Equation (14).

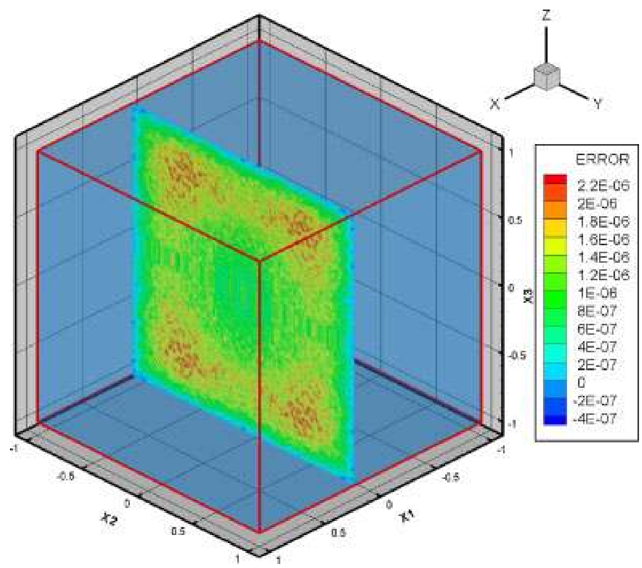


Figure 16. Slice on the error map of the problem stated in Equation (14).

In this example we do not pursue accuracy, but testing the possibility of employing a huge number of degrees of freedom. What is noticeable about this solution, which approaches the limits of traditional FEs, is that, despite the extremely high number of degrees of freedom involved, the Matlab code ran in around 10 s on a PC with four processors (only one was employed, no

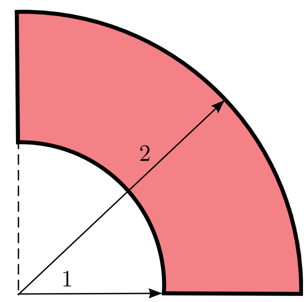


Figure 17. Geometry of the problem in Section 4.5.

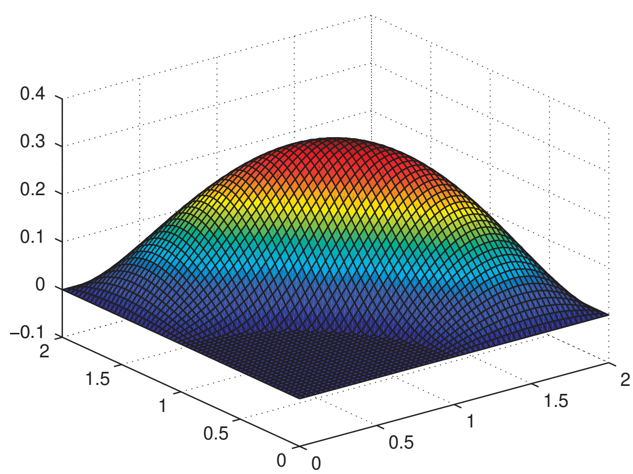


Figure 18. Characteristic function for the problem in Section 4.5.

parallel calculation was allowed, even if parallelization is an easy matter in the context of separated representations) with 1.96 Gb of RAM memory each, running under Linux.

4.5. Models defined in domains other than hypercubes

To show that the proposed method is applicable to problems defined over irregularly shaped domains, other than hypercubes, we consider the following problem defined over a cylindrical domain whose geometry is depicted in Figure 17. The problem is defined as $\Delta u = 1.0$, with $u = 0$ on $\partial\Omega$.

Since the analytical solution is not known, we have compared the results with those from a finite element simulation over a fine enough mesh, using standard linear triangular elements over the actual geometry of the domain. By applying the technique explained in Section 3.1, we computed the characteristic function depicted in Figure 18.

In order to apply the separated representation strategy, we first embed the geometry in a squared in which each axis contains 60×60 nodes. The solution outside the domain of interest depicted in Figure 17 is obviously discarded.

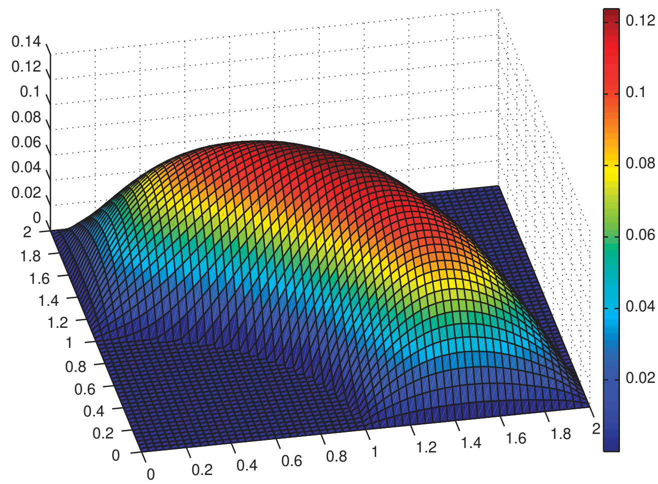


Figure 19. Solution for the problem in Section 4.5.

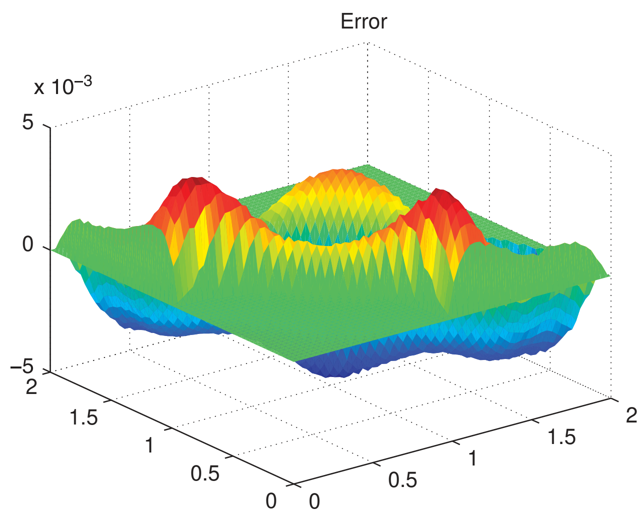


Figure 20. Error in the approximation of the problem in Section 4.5.

By comparing the separated representation solution to that obtained by a dense enough FE mesh, and by using the norm $\|e\|_2 = (1/n)\sqrt{\sum_i^n e_i^2}$ (with n the number of points in which the error is evaluated and e_i the pointwise error at that points), we have obtained after the first iteration an error of 2.97×10^{-5}, with no significant improvement by adding other sums of the decomposition. The computed solution is depicted in Figure 19. If more accurate solutions are desired, higher nodal densities along each direction should be employed. It is important to note that the separated representation solution cannot produce solution more accurate than the ones related to a finite element FE discretization on the mesh resulting from the one-dimensional nodal distributions tensor product (Figure 20).

5. CONCLUSIONS

Problems defined in high-dimensional spaces are very common in Engineering and Applied Sciences. The common feature of all these models relies in the exponentially growing size of the mesh with the number of dimensions of the problem. In previous works by the authors, a technique based on the use of a separated representation consisting of a finite sums decomposition has proven to be very useful for some of these problems.

However, the theoretical establishment of these techniques still deserves further developments. In this paper we have dealt with two particular features of the method, never until now, in our knowledge, addressed. The first one is the imposition of non-homogeneous essential boundary conditions and the second one is related to the simulation in domains other than hypercubes.

The proposed technique is based upon the construction of a change of variable for the first case. This change of variable must be compatible, of course, with the separated representation strategy needing the use of multidimensional SVD techniques for that purpose. The second issue related to the treatment of complex geometries has been accomplished through the method of the characteristic function, a very classical method, recently recovered within the framework of meshless methods. Using both strategies we have solved some numerical models proving the usefulness of the proposed techniques and, in turn, enlarge its field of application. We believe that the proposed techniques have an enormous potential in a wide range of applications, and this constitutes our current effort of the research.

The extension of these strategies to some physical models defined in high dimensions and the consideration of general boundary conditions including general natural conditions are some works in progress.

REFERENCES

1. Bungartz HJ, Griebel M. Sparse grids. *Acta Numerica* 2004; **13**:1–123.
2. Ammar A, Mokdad B, Chinesta F, Keunings R. A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids. *Journal of Non-Newtonian Fluid Mechanics* 2006; **139**:153–176.
3. Ammar A, Mokdad B, Chinesta F, Keunings R. A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids. Part II: transient simulation using space-time separated representations. *Journal of Non-Newtonian Fluid Mechanics* 2007; **144**:98–121.
4. Chinesta F, Ammar A, Joyot P. 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* 2008; **6/3**:191–213.
5. Chinesta F, Ammar A, Falco A, Laso M. On the reduction of stochastic kinetic theory models of complex fluids. *Modeling and Simulation in Materials Science and Engineering* 2007; **15**:639–652.
6. Mokdad B, Pruliere E, Ammar A, Chinesta F. On the simulation of kinetic theory models of complex fluids using the Fokker–Planck approach. *Applied Rheology* 2007; **17/2**(26494):1–14.
7. Pruliere E, Ammar A, El Kissi N, Chinesta F. Multiscale modelling of flows involving short fibersuspensions. *Archives of Computational Methods in Engineering*, *State of the Art Reviews* 2009; **16**:1–30.
8. Cancès E, Defranceschi M, Kutzelnigg W, Le Bris C, Maday Y. *Computational Quantum Chemistry*: *A Primer*, *Handbook of Numerical Analysis*, vol. X. Elsevier: Amsterdam, 2003; 3–270.
9. Ladeveze P. *Nonlinear Computational Structural Mechanics*. Springer: NY, 1999.
10. Rektorys K. *Variational Methods in Mathematics*, *Science and Engineering*. D. Reidel publishing Company: Dordrecht, 1980.
11. Rvachev VL, Sheiko TI. R-functions in boundary value problems in mechanics. *Applied Mechanics Reviews* 1995; **48**:151–188.

12. Rvachev VL, Sheiko TI, Shapiro V, Tsukanov I. On completeness of RFM solution structures. *Computational Mechanics* 2000; **25**:305–316.
13. Messiter M, Shamash Y. Product and sum separable functions. *IEEE Transactions on Automatic Control* 1985; **30**(7):694–697.
14. Babuska I, Banerjee U, Osborn E. Survey of meshless and generalized finite element methods: a unified approach. *Acta Numerica* 2003; **12**:1–125.
15. Kolda TG, Bader BW. Tensor decompositions and applications. *Technical Report SAND2007-6702*, SANDIA National Laboratories, November 2007.
16. Bader BW, Kolda TG. Efficient MATLAB computations with sparse and factored tensors. *SIAM Journal on Scientific Computing* 2007; **30**(1):205–231.