

Streamline upwind/Petrov–Galerkin-based stabilization of proper generalized decompositions for high-dimensional advection–diffusion equations

David González¹, Elías Cueto^{1,*},†, Francisco Chinesta^{2,3}, Pedro Díez⁴ and Antonio Huerta⁴

¹*Aragón Institute of Engineering Research (I3A). Universidad de Zaragoza. Edificio Betancourt. María de Luna, s.n. 50018 Zaragoza, Spain*

²*EADS Corporate Foundation International Chair, École Centrale de Nantes. 1, Rue de la Noë. 44300 Nantes, France*

³*Institut Universitaire de France*

⁴*Laboratori de Calcul Numeric, Universitat Politècnica de Catalunya, Barcelona, Spain*

SUMMARY

This work is a first attempt to address efficient stabilizations of high dimensional advection–diffusion models encountered in computational physics. When addressing multidimensional models, the use of mesh-based discretization fails because the exponential increase of the number of degrees of freedom related to a multidimensional mesh or grid, and alternative discretization strategies are needed. Separated representations involved in the so-called proper generalized decomposition method are an efficient alternative as proven in our former works; however, the issue related to efficient stabilizations of multidimensional advection–diffusion equations has never been addressed to our knowledge. Thus, this work is aimed at extending some well-experienced stabilization strategies widely used in the solution of 1D, 2D, or 3D advection–diffusion models to models defined in high-dimensional spaces, sometimes involving tens of coordinates. Copyright © 2013 John Wiley & Sons, Ltd.

Received 25 October 2012; Revised 28 January 2013; Accepted 15 February 2013

KEY WORDS: convection–diffusion equation; SUPG; proper generalized decomposition; separated representations; finite sum decomposition; Fokker–Planck equation

1. INTRODUCTION

Despite the continuous growth of computer performance, some models remain nowadays intractable for current, state-of-the-art simulation techniques. Models defined in highly dimensional spaces, for instance, constitute a clear example of such problems. The interested reader can consult, for instance, [1] for an excellent survey of the state of the art on numerical simulation of micro–macro models for complex fluids. These micro–macro models are very often based on the Fokker–Planck equation, an advection–diffusion equation that describes the evolution of the probability distribution function of the phase–space variables, namely

$$\frac{D\psi}{Dt} = -\frac{\partial}{\partial \mathbf{X}} \cdot \{\mathbf{A}\psi\} + \frac{1}{2} \frac{\partial}{\partial \mathbf{X}} \frac{\partial}{\partial \mathbf{X}} : \mathbf{D}\psi$$

*Correspondence to: Elías Cueto, Universidad de Zaragoza. Edificio Betancourt. María de Luna, s.n. E-50018 Zaragoza, Spain.

†E-mail: ecueto@unizar.es

where $\psi(x, X, t)$ represents the probability distribution function of the phase–space variables X . Whereas A is an N_d -dimensional deterministic vector responsible for the drift of these variables by the macroscopic flow, and D represents an $N_d \times N_d$ -dimensional stochastic matrix responsible for Brownian effects on the flow. For Rouse models, for instance, N_d can easily take values on the order of 50 or more [2].

Recently, the authors have presented and applied to different classes of problems an a priori model reduction technique based upon the approximation of the essential field by a finite sum of separable functions that are generated ‘on the fly’ by the method itself, see, for instance [3–9]. This method, coined as proper generalized decomposition (PGD, as opposed to a posteriori methods as the proper orthogonal decomposition [10–12]), also allows for an easy treatment of problems defined in spaces of a high number of dimensions. Mesh-based techniques lead to the well-known *curse of dimensionality*, because the number of degrees of freedom grows exponentially with the number of dimensions. The use of separated representations allows to overcome this difficulty and has been used in many fields of Science and Engineering, such as 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 multi-configurational self-consistent field that involves a finite number of sums) make use of a separated representation of the wave function [13]. In the context of Computational Mechanics, a similar decomposition was proposed, which was called radial approximation, and that was applied for separating the space and time coordinates in thermomechanical models [14].

A detailed analysis of the convergence properties of different approaches of the PGD technique can be found at [15]. But this analysis has been restricted, for the moment, to symmetric operators. It remains unknown, to the best of the authors’ knowledge, how the PGD behaves, for instance, when dealing with convection–diffusion equations.

It is well-known that standard finite element (Galerkin) methods do not work well for convection–diffusion or convection–diffusion–reaction equations, because they lead to unstable, oscillating solutions [16]. The first stabilization methods in finite elements including upwinding of convective terms that eventually lead to stable solutions were published in [17, 18].

Among the very numerous methods that have been proposed for the stabilization of convection–diffusion equations, the streamline upwind/Petrov–Galerkin (SUPG) method [19] is one of the most extended. When a reaction term is important, subgrid scale (SGS) techniques have been advocated to reduce oscillations, see [20]. An inherent difficulty of these methods is the choice of the stabilization parameter. In fact, algebraic or asymptotic analyses have been developed in one-dimensional problems. Optimal stabilization parameters in higher dimensions are not easily obtained.

This paper has neither the aim of addressing a state of the art on stabilization techniques for advection–diffusion equations and the consequent evaluation and choice of the optimal stabilization technique nor the proposal of new stabilization strategies. This work is a first attempt to address efficient stabilizations of high dimensional advection–diffusion models encountered in computational physics. When addressing multidimensional models, the use of mesh-based discretization fails because the exponential increase of the number of degrees of freedom related to a multidimensional mesh or grid and alternative discretization strategies are needed. Separated representations involved in the so-called PGD method is an efficient alternative as proven in our former works, some of them already cited; however, the issue related to efficient stabilizations of multidimensional advection–diffusion equations has never been addressed. Thus, this work has, as main aim, the extension of some well-experienced stabilization strategies widely used in the solution of 1D, 2D, or 3D advection–diffusion models to models defined in high dimensional spaces, sometimes involving tens of coordinates (the space and the time but also a number of configurational coordinates describing some extra model features). Thus, in what follows, we will focus on the extension of standard SUPG or subgrid scale stabilizations to solutions expressed in a separated approximation format.

Here, the use of finite sums of separable functions will be used to reduce a high-dimensional problem to a sequence of one-dimensional ones where optimal stabilization could be employed. The outline of the paper is as follows. In Section 2, we review the technique of separation of variables developed by the authors in the framework of convection–diffusion problems. In Section 3, we describe the framework of the class of convection–diffusion–reaction problems we deal with. We

then provide two alternative formulations and discuss the advantages of each one in Sections 4 and 5. In Section 6, we show some examples illustrating the performance and capabilities of the technique so as to verify their stabilizing properties. These techniques are then applied in Section 7 to a 5-dimensional convection–diffusion problem.

2. THE PROPER GENERALIZED DECOMPOSITION

The use of a separated representation for solving complex models is not new. It has been employed in a variety of settings, including, as mentioned before, quantum mechanics in the Hartree–Fock approach or continuum mechanics, see the pioneering work by Ladeveze and coworkers [14]. It is based on the establishment of an approximation of the unknown field, u , in the form:

$$u(\mathbf{x}) \approx u^n(x_1, x_2, \dots, x_{n_{\text{sd}}}) = \sum_{i=1}^n F_1^i(x_1) \cdot F_2^i(x_2) \cdot \dots \cdot F_{n_{\text{sd}}}^i(x_{n_{\text{sd}}}), \quad (1)$$

with n_{sd} as the number of spatial dimensions of the problem ($n_{\text{sd}} \gg 1$ within the scope of this work, although the results are general for any number of dimensions). The term proper generalized decomposition has been coined for methods that use this kind of separated representations, making use of an appropriate algorithm to determine the functions F_j^i . This a priori character of the technique is opposed to the a posteriori character of other model reduction techniques as proper orthogonal decomposition or Karhunen–Loeve method [10, 11, 21].

For the sake of simplicity in the description of the technique and without loss of generality, two-dimensional problems are considered (they will be generalized later on, see Section 6); thus, the unknown field can be written as

$$u(x, y) \approx u^n(x, y) = \sum_{i=1}^n F^i(x) \cdot G^i(y). \quad (2)$$

To construct this separated representation, an iterative, greedy algorithm is proposed. Assuming that the first n functions in the sum have been already computed, see (2), the method proceeds by finding the best functions $R(x)$ and $S(y)$ such that the updated representation given by

$$\begin{aligned} u^{n+1}(x, y) &= \sum_{i=1}^n F^i(x) \cdot G^i(y) + R(x) \cdot S(y) \\ &= u^n(x, y) + R(x) \cdot S(y), \end{aligned}$$

satisfies the weak formulation for a particular choice of the test functions. These test functions are

$$w(x, y) = R^*(x) \cdot S(y) + R(x) \cdot S^*(y). \quad (3)$$

Once these test functions are introduced in a weak form, such as (7) or (8), the resulting nonlinear problem of finding functions R and S is solved by using an iterative, fixed-point method. First, it is assumed that R is known and a new approximation for S is computed. Then assuming S is known, a new approximation for R is sought. This process is repeated until convergence.

For more details on the separated representation constructor, the interested reader can refer to [5, 8] and the references therein.

3. PROBLEM SETTING

For simplicity and without loss of generality, we consider the steady-state convection–diffusion–reaction equation. This equation is given by

$$\mathbf{a} \cdot \nabla u - \nabla \cdot (\nu \nabla u) + \sigma u = s \text{ in } \Omega \subset \mathbb{R}^{n_{\text{sd}}}, \quad (4)$$

with n_{sd} the number of spatial dimensions—in the applications we are interested in much larger than one—and with boundary conditions

$$u = u_D \text{ on } \Gamma_D, \tag{5a}$$

$$\mathbf{n} \cdot \nu \nabla u = v \frac{\partial u}{\partial \mathbf{n}} = t \text{ on } \Gamma_N, \tag{5b}$$

where u is the scalar unknown field, \mathbf{a} is the advective velocity, $\nu > 0$ the diffusivity, assumed (without loss of generality) constant, σ the reaction term and $s(\mathbf{x})$ a volumetric source term. The function u_D denotes the prescribed value of u on the Dirichlet portion of the boundary given by Γ_D , and t denotes the value of the normal diffusive flux on the Neumann boundary Γ_N .

The weak form of the problem defined in Equations (4) and (5) is find $u(\mathbf{x}) \in \mathcal{S} = \{u \in \mathcal{H}^1(\Omega) | u = u_D \text{ on } \Gamma_D\}$ such that for all $w \in \mathcal{V} = \{w \in \mathcal{H}^1(\Omega) | w = 0 \text{ on } \Gamma_D\}$

$$\int_{\Omega} w(\mathbf{a} \cdot \nabla u) d\Omega + \int_{\Omega} \nabla w \cdot (\nu \nabla u) d\Omega + \int_{\Omega} w \sigma u d\Omega = \int_{\Omega} w s d\Omega + \int_{\Gamma_N} w t d\Gamma. \tag{6}$$

This is usually expressed compactly as

$$a(w, u) + c(\mathbf{a}; w, u) + (w, \sigma u) = (w, s) + (w, t)_{\Gamma_N}, \tag{7}$$

where the following definitions are employed:

$$\begin{aligned} a(w, u) &= \int_{\Omega} \nabla w \cdot (\nu \nabla u) d\Omega, & c(\mathbf{a}; w, u) &= \int_{\Omega} w(\mathbf{a} \cdot \nabla u) d\Omega, \\ (w, u) &= \int_{\Omega} w u d\Omega, & (w, t)_{\Gamma_N} &= \int_{\Gamma_N} w t d\Gamma. \end{aligned}$$

The general form of a *consistent stabilization* technique is [16]

$$a(w, u) + c(\mathbf{a}; w, u) + (w, \sigma u) + \sum_e \int_{\Omega^e} \mathcal{P}(w) \tau \mathcal{R}(u) d\Omega = (w, s) + (w, t)_{\Gamma_N}, \tag{8}$$

where $\mathcal{P}(w)$ is some operator applied to the test functions, $\mathcal{R}(u) = \mathcal{L}(u) - s$ is the residual of the equation, and τ is the stabilization parameter. Note that the differential operator \mathcal{L} is the one associated to the strong form, Equation (4), namely

$$\mathcal{L}(u) = \mathbf{a} \cdot \nabla u - \nabla \cdot (\nu \nabla u) + \sigma u.$$

In the SUPG method, $\mathcal{P}(w) = \mathbf{a} \cdot \nabla w$, whereas in SGS, $\mathcal{P}(w) = -\mathcal{L}^*(w)$, where \mathcal{L}^* is the adjoint operator, that is,

$$\mathcal{L}^*(u) = -\mathbf{a} \cdot \nabla u - \nabla \cdot (\nu \nabla u) + \sigma u.$$

Nodally exact results are obtained for linear elements in 1D for the convection–diffusion if

$$\tau = \frac{h}{2a} \left(\coth Pe - \frac{1}{Pe} \right), \tag{9}$$

where Pe is the mesh Péclet number, defined as $Pe = ah/2\nu$; h represents the mesh size parameter and a the modulus of the convective velocity. For convection–diffusion–reaction a fourth-order accurate formula reads

$$\tau = \frac{h}{2a} \left(1 + \frac{9}{Pe^2} + \left(\frac{h}{2a} \sigma \right)^2 \right)^{-1/2}. \tag{10}$$

As noted previously, generalizations to higher dimensions are not readily available [16].

The purpose of the remainder of this paper is to introduce a stabilized technique based on the use of the PGD introduced in [3,4] for high-dimensional problems. This kind of approximation makes use of a representation of the essential field in terms of a finite sum of separable functions.

In Sections 4 and 5, two alternative approaches based on PGD schemes are developed. A discussion with numerical examples follows. The first approach does the separation in infinite dimensional spaces, then when the finite element discretization is needed, the optimal stabilization parameter is chosen. The second one proceeds conversely, first the high-dimensional convection–diffusion equations is discretized, taking into account that stabilization is needed, and then separation following the PGD rationale is imposed.

4. STABILIZATION OF THE CONTINUOUS PGD SEPARATION

The PGD method is applied to the infinite dimensional problem defined by the weak form, Equation (7). In what follows, for the sake of simplicity, the reaction term of the equation is neglected, $\sigma = 0$, because it does not imply any special difficulty for the proposed methodology. Thus, the weak problem (7) becomes find $R(x)$ and $S(x)$ for all w or more precisely from (3) for all $R^*(x)$ and $S^*(x)$ in the proper infinite dimensional spaces, such that

$$a(w, RS) + c(\mathbf{a}; w, RS) = (w, s) + (w, t)_{\Gamma_N} - a(w, u^n) - c(\mathbf{a}; w, u^n).$$

As noted earlier, the PGD method proceeds by a sort of alternating direction strategy, assuming iteratively that either R or S is known. For instance, in the previous equation, assuming that R is given and that S must be found, the weak form for all S^* reads

$$a(RS^*, RS) + c(\mathbf{a}; RS^*, RS) = (RS^*, s) + (RS^*, t)_{\Gamma_N} - a(RS^*, u^n) - c(\mathbf{a}; RS^*, u^n),$$

which is, in fact, a 1D *convection–diffusion–reaction* problem (recall that is standard to assume in PGD that $\Omega = \Omega_x \otimes \Omega_y$), namely

$$(S^{*'} , \gamma_y S')_{\Omega_y} + (S^* , c_y S')_{\Omega_y} + (S^* , \kappa_y S)_{\Omega_y} = (S^* , b_y)_{\Omega_y} + S^* p_y |_{\Upsilon_{N_y}^y}, \quad (11)$$

where the following definitions are used:

$$\begin{aligned} \gamma_y &= \int_{\Omega_x} v R^2 dx, & c_y &= \int_{\Omega_x} a_y R^2 dx, & \kappa_y &= \int_{\Omega_x} (v R'^2 + a_x R R') dx, \\ b_y &= \int_{\Omega_x} \left(s R - v R' \frac{\partial u^n}{\partial x} - R \mathbf{a} \cdot \nabla u^n \right) dx + \frac{d}{dy} \left[\int_{\Omega_x} v R \frac{\partial u^n}{\partial y} dx \right], & \text{and} & \\ p_y &= \int_{\Upsilon_{N_y}^x} t R dx - \int_{\Omega_x} v R \frac{\partial u^n}{\partial y} dx. \end{aligned} \quad (12)$$

It is important to remark that given the usual PGD assumption $\Omega = \Omega_x \otimes \Omega_y$, Neumann boundary conditions are also separable in the x and y problems. Here, because the unknown is $S(y)$, if $\Gamma_N \cap \partial\Omega_y = \Gamma_{N_y} \neq \emptyset$, then Γ_{N_y} must be separable as $\Gamma_{N_y} = \Upsilon_{N_y}^y \otimes \Upsilon_{N_y}^x$. Thus, p_y defined in (12) requires to integrate for a fixed y (corresponding to the Neumann boundary, denoted symbolically as $\Upsilon_{N_y}^y$) along the x direction, which is described as $\Upsilon_{N_y}^x$.

Note that proceeding identically with ‘known’ S and test function $R^* S$, an equivalent problem to determine R can be stated. This would induce a problem similar to Equations (11) and (12) but along the x direction.

In any case, Equation (11) can be identified as a convection–diffusion–reaction weak form, which is stabilized as follows:

$$\begin{aligned} & (S^{*'}, \gamma_y S')_{\Omega_y} + (S^*, c_y S)_{\Omega_y} + (S^*, \kappa_y S)_{\Omega_y} \\ & + \sum_e \int_{\Omega_y^e} \mathcal{P}_y(S^*) \tau_y (c_y S' - (\gamma_y S')' + \kappa_y S) dy \\ & = (S^*, b_y)_{\Omega_y} + S^* p_y|_{\Gamma_{N_y}^y} + \sum_e \int_{\Omega_y^e} \mathcal{P}_y(S^*) \tau_y b_y dy, \end{aligned} \tag{13}$$

where the operator applied to the test functions is $\mathcal{P}_y(w) = c_y w'$ in SUPG and $\mathcal{P}_y(w) = c_y w' + (\gamma_y w')' - \kappa_y w$ in SGS. Moreover, optimal values of the stabilization parameter can be employed substituting diffusion, γ_y , convection, c_y , and reaction, κ_y , in Equations (9) or (10). For linear elements, the second derivatives in the stabilization terms are zero but recall that for high-order elements, consistent stabilization is important [22].

5. SEPARATION OF THE DISCRETE STABILIZED EQUATION

The other natural possibility arising from the PGD method is to perform a separated approximation of the already stabilized Equation (8). The method proceeds in a very similar way, but now, only one stabilization parameter must be determined (as in standard finite element techniques). The same stabilization parameter will be used in each one-dimensional problem. Moreover, although in each 1D direction is a different convection–diffusion–reaction problem, τ will be unique and defined a priori. Thus, in principle, for a convection–diffusion problem ($\sigma = 0$) such as the one studied here, reaction will not be accounted for in the stabilization.

In the stabilized weak problem (8), once the PGD method proceeds assuming iteratively that either R or S is known, becomes, when for instance, R is given and that S must be found,

$$\begin{aligned} & a(RS^*, RS) + c(\mathbf{a}; RS^*, RS) \\ & + \sum_e \int_{\Omega^e} \mathcal{P}(RS^*) \tau (\mathbf{a} \cdot \nabla(RS) - \nabla \cdot (v \nabla(RS))) d\Omega \\ & = (RS^*, s) + (RS^*, t)_{\Gamma_N} - a(RS^*, u^n) - c(\mathbf{a}; RS^*, u^n) \\ & - \sum_e \int_{\Omega^e} \mathcal{P}(RS^*) \tau (\mathbf{a} \cdot \nabla u^n - \nabla \cdot (v \nabla u^n) - s) d\Omega. \end{aligned}$$

To compare with the previous formulation, the 1D problem in y for SUPG is presented. However, such comparison is only possible if further assumptions are imposed, namely, that diffusion and convection are independent of x and y , which is not always the case. Under these assumptions, the previous equation can be rewritten as

$$\begin{aligned} & (S^{*'}, \gamma_y S')_{\Omega_y} + (S^*, c_y S)_{\Omega_y} + (S^*, \kappa_y S)_{\Omega_y} \\ & + \sum_e \int_{\Omega_y^e} a_y S^{*'} \tau (c_y S' - \gamma_y S'' + \tilde{\kappa}_y S) dy + \sum_e \int_{\Omega_y^e} a_x S^* \tau (\hat{c}_y S' - \hat{\gamma}_y S'' + \hat{\kappa}_y S) dy \\ & = (S^*, b_y)_{\Omega_y} + S^* p_y|_{\Gamma_{N_y}^y} \\ & - \sum_e \int_{\Omega^e} (a_x R' S^* + a_y R S^{*'}) \tau (\mathbf{a} \cdot \nabla u^n - \nabla \cdot (v \nabla u^n) - s) d\Omega, \end{aligned} \tag{14}$$

where the definitions of (12) are used and the following ones are needed

$$\begin{aligned} \tilde{\kappa}_y &= \int_{\Omega_x} (-v R R'' + a_x R R') dx, \\ \hat{\gamma}_y &= \int_{\Omega_x} v R R' dx, \quad \hat{c}_y = \int_{\Omega_x} a_y R R' dx, \quad \text{and} \quad \hat{\kappa}_y = \int_{\Omega_x} (-v R' R'' + a_x R'^2) dx. \end{aligned}$$

This clearly evidences that both formulations are not, even in the simplified case, nearly equivalent. Moreover, it is obvious that the stabilization of this 1D problem is not optimal. Nevertheless, as shown in the examples, stabilized results are obtained.

6. VERIFYING THE STABILIZATION PROCEDURES

Before addressing the solution of a high-dimensional advection–diffusion equation, we are verifying the previous extensions of standard stabilization towards the separated representation format work. For this, we are focusing in some well-illustrated 2D benchmark problems, to which a reference solution can be easily computed to conclude on the performances of the considered stabilizations within the PGD framework. After proving the ability of these procedures for addressing the stabilization within the PGD framework, we will address a problem defined in a high-dimensional space to which no reference solution is available.

6.1. A first example

In this section, we consider an example of convection–diffusion skew to the mesh, proposed in [16], among other references, see Figure 1.

The flow is unidirectional, with $\|a\| = 1$, with the convective velocity not aligned to any of the axes, forming 30° with the x -direction. Thus, the results will be entirely comparable with those in reference [16]. Following this same reference, the diffusivity coefficient is taken to be $5 \cdot 10^{-6}$, corresponding to a mesh Péclet number of 10^4 . The inlet boundary conditions are discontinuous and of two different types at the outlet:

- Downwind homogeneous natural boundary conditions
- Downwind homogeneous essential boundary conditions

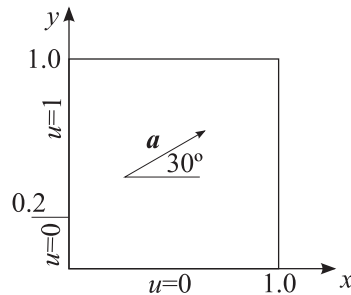


Figure 1. Statement of the problem used for the validation of the proposed technique.

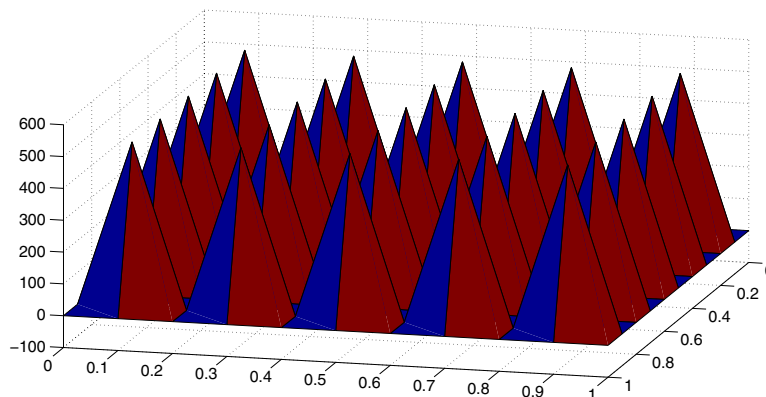


Figure 2. Solution obtained by applying the separated representation approach to standard, non-stabilized Galerkin procedures.

We compare the results obtained with formulations 1 (introduced in section 4) and 2 (section 5) explained before in the framework of proper generalized decompositions. In Figure 2, results for the Dirichlet problem solved with PGD and standard Galerkin approach are shown. No stable results were obtained for a mesh of 2000 finite elements along each spatial direction.

To compare with existing techniques, the problem has been solved by SUPG-stabilized standard finite elements. A refined mesh of 100×100 finite elements has been employed to obtain a reference solution to compare with. In addition, and for the purpose of comparison, a 10×10 mesh was employed. These results are depicted in Figures 3 and 4 for Dirichlet and Neumann boundary conditions, respectively.

Results obtained with the PGD formulation 1 and 2 are depicted in Figure 5 for the Dirichlet problem and Figure 6 for the Neumann problem. As it can be noticed, results are practically indistinguishable for this problem.

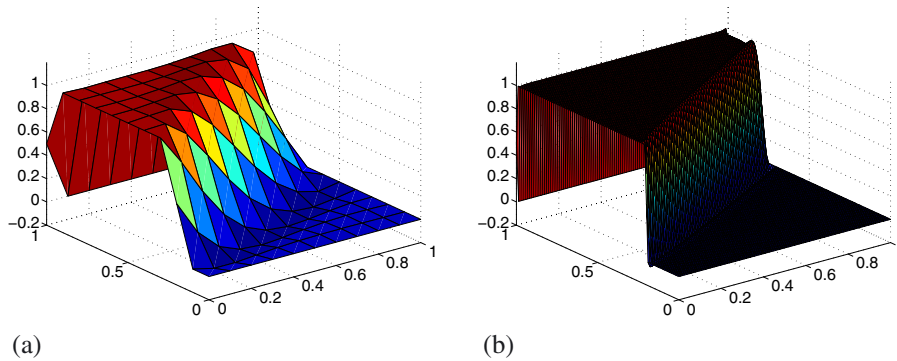


Figure 3. Dirichlet problem: FEM-SUPG result with a (a) 10×10 mesh and (b) 100×100 mesh.

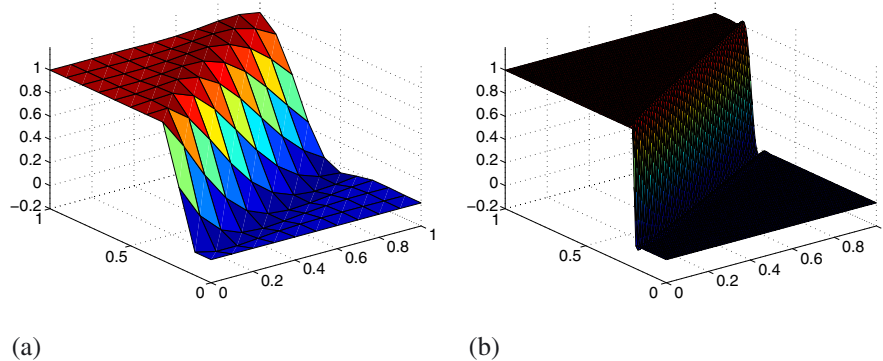


Figure 4. Neumann problem: FEM-SUPG result with a (a) 10×10 mesh and (b) 100×100 mesh.

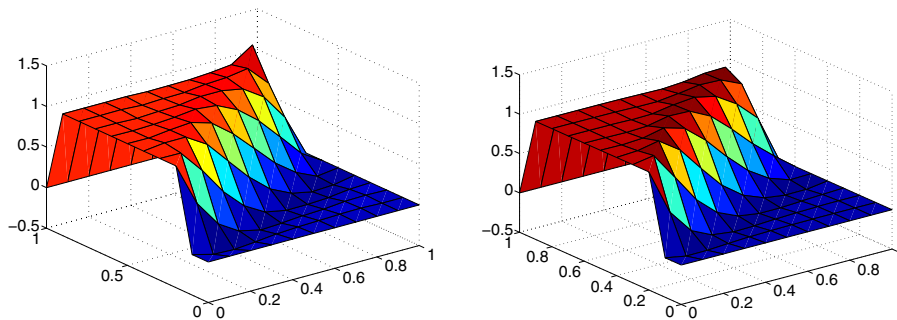


Figure 5. Dirichlet problem: results of formulations 1 and 2.

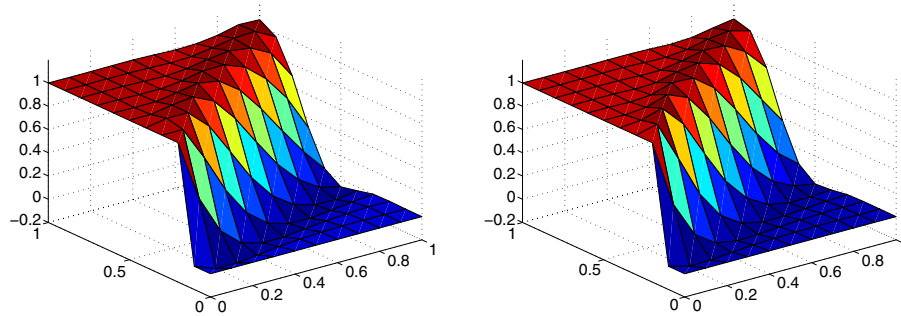


Figure 6. Neumann problem: results of formulations 1 and 2.

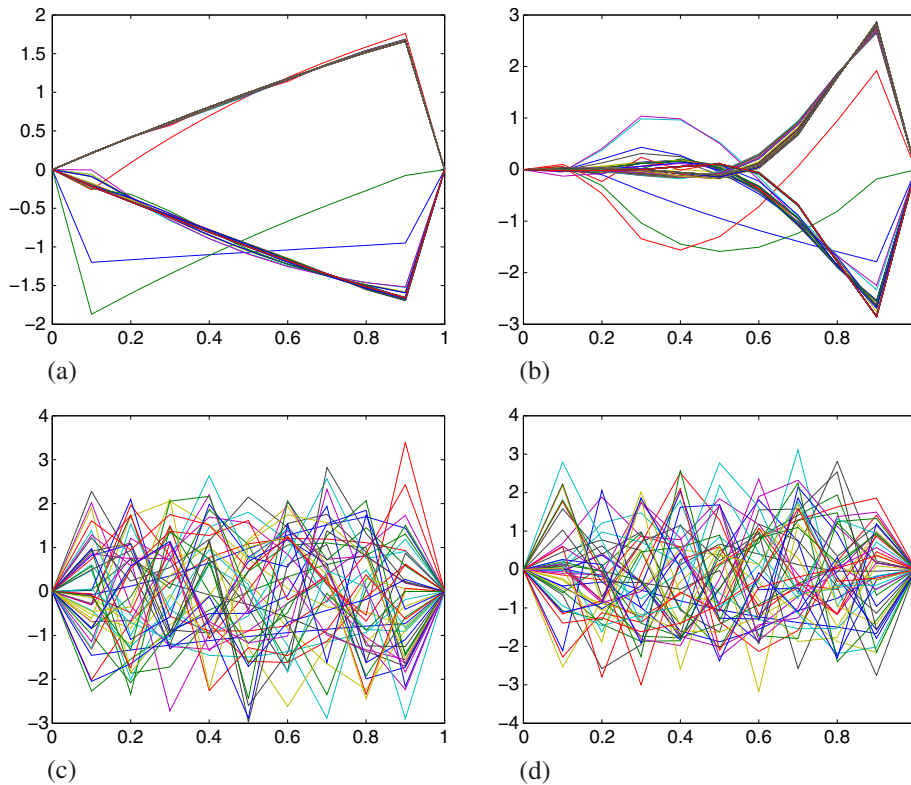


Figure 7. Functions employed in the approximation of problem 1 with Dirichlet boundary conditions, (a) along x -coordinate and (b) y -coordinate. Formulations 1 (top row) and 2 (bottom row) are depicted.

These results were obtained with the product of 85 functions for version 1 and 20 functions for version 2 along each direction. These functions are depicted in Figure 7 for the Dirichlet problem and formulations 1 and 2. The great number of very similar functions obtained by the PGD method and version 1 can be noticed. Note that in the aforementioned Figure 7, the approximating functions have been normalized. Although 85 versus 20 functions seems to be a great difference for both approaches to the problem, by looking at the convergence plots in Figure 8, it can be noticed that both formulations, for a fixed number of summands, give very similar levels of accuracy. It seems, however, that the functions incorporated by the second approach are closer to be mutually orthogonal in comparison with those of version 1. Namely, their cosines, computed as their scalar product divided by the product of their norms, is lower for version 2 than for version 1.

To clearly compare the results, a cut along $x = 0.5$ has been carried out. These are plotted in Figures 9 and 10 for the Dirichlet and Neumann problems, respectively.

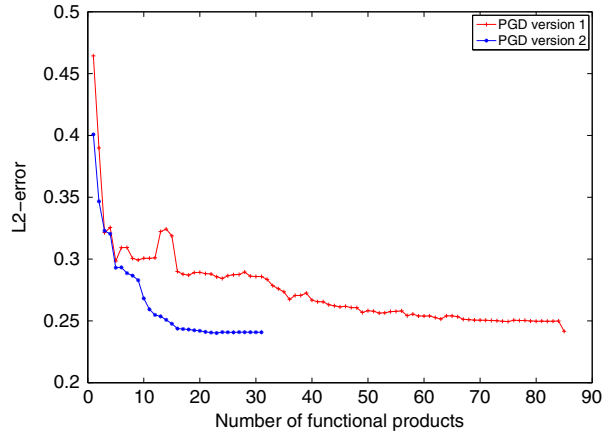


Figure 8. Convergence for the Dirichlet problem and formulations 1 and 2. Error in \mathcal{L}_2 -norm versus number of functions incorporated to the approximation.

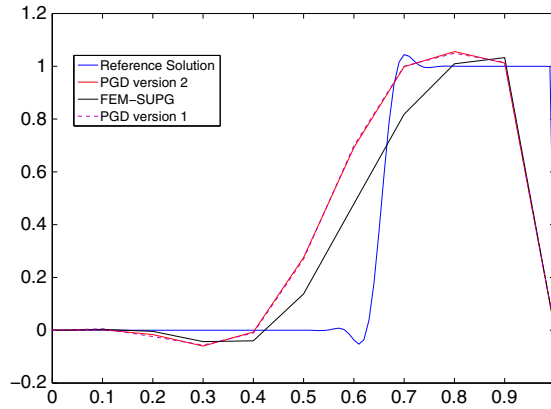


Figure 9. Comparison between standard, finite element stabilized (SUPG) approximations and the proposed PGD formulations for the Dirichlet problem.

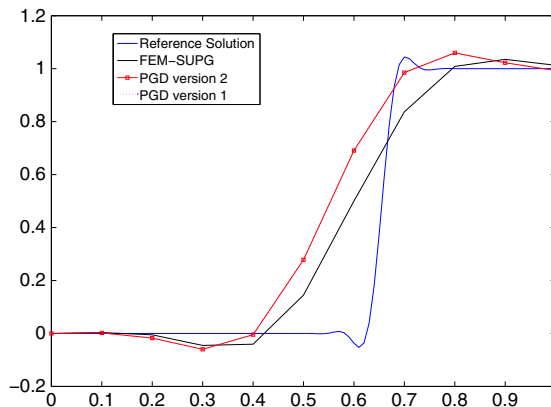


Figure 10. Comparison between standard, finite element stabilized (SUPG) approximations and the proposed PGD formulations for the Neumann problem.

Table I. Comparison of ‘errors’ with respect to the reference solution of a 100×100 SUPG-stabilized finite element mesh. Dirichlet problem.

Formulation	1	2	FEM–SUPG
$\ e\ _{L_2}$	0.249436	0.240834	0.273473
$\ e\ _{\mathcal{H}^1}$	1.249441	1.242036	1.273479

Table II. Comparison of ‘errors’ with respect to the reference solution of a 100×100 SUPG-stabilized finite element mesh. Neumann problem.

Formulation	1	2	FEM–SUPG
$\ e\ _{L_2}$	0.251757	0.241130	0.259762
$\ e\ _{\mathcal{H}^1}$	1.251761	1.242258	1.259767

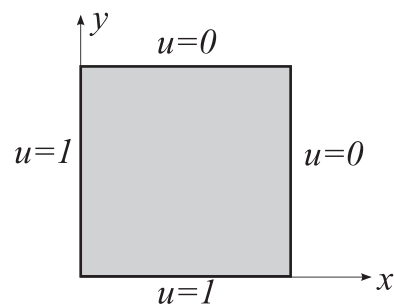


Figure 11. Geometry of the problem analyzed in section 6.2 .

Taking the 100×100 SUPG-stabilized finite element mesh results as a reference, we can compute the norm of the ‘error’ with respect to this reference. These values are reported in Tables I and II for the Dirichlet and Neumann boundary conditions, respectively.

As can be noticed, the proposed formulation (both, in fact) provides a somewhat more steep slope in the jump while there appears to be slightly high oscillations. To better understand the approximation provided by the proposed method, we include the next example.

6.2. A second example

We consider the problem

$$\cos\left(\frac{\pi}{3}\right) \frac{\partial u}{\partial x} + \sin\left(\frac{\pi}{3}\right) \frac{\partial u}{\partial y} = 10^{-4} \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) + 1,$$

solved on the unit square, with boundary conditions (see Figure 11)

- $u = 1$ on $x = 0$,
- $u = 1$ on $x = 1$,
- $u = 0$ on $y = 0$,
- $u = 0$ on $y = 1$.

In the absence of analytical solution, a FEM–SUPG solution on a mesh composed by 100×100 linear elements is employed as a reference, see Figure 12.

Results are compared with the reference solution and that of a 10×10 standard linear finite elements with SUPG stabilization. As a reference value, for the formulation 2, 20 functions along each spatial direction were employed, see Figure 13. Similar values are obtained for formulation 1.

Results are better analyzed by comparing a cut of the obtained results at abscissa $x = 0.8$. These are shown in Figure 14 below.

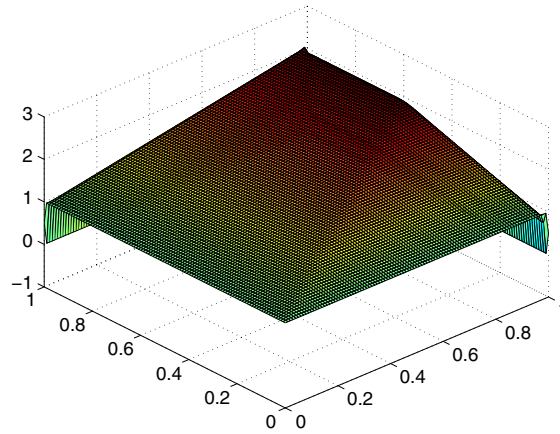


Figure 12. Reference solution for the problem in section 6.2, obtained by FEM-SUPG and a 100×100 linear finite element mesh.

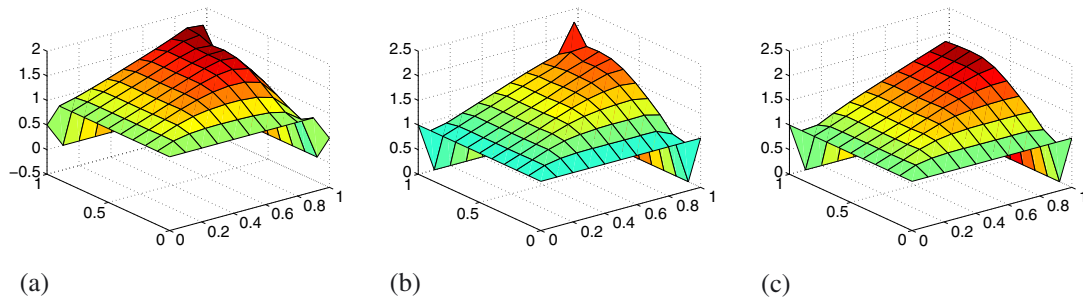


Figure 13. Comparison between the obtained results by (a) FEM-SUPG (10×10 elements), (b) formulation 1, and (c) formulation 2.

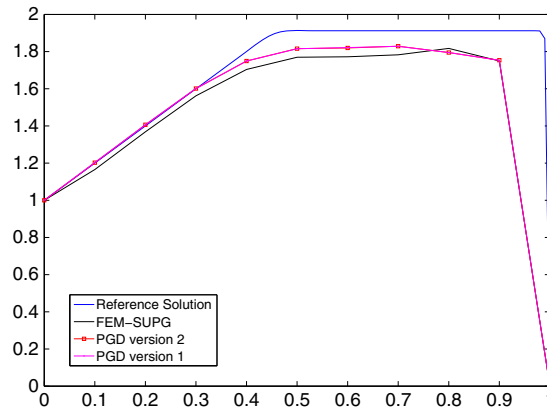


Figure 14. Cut along $x = 0.8$ for the problem in section 6.2.

If we compare the obtained errors with respect to the reference solution, we obtain the results summarized in Table III.

6.3. A third example

This time we solved a problem in the unit square with $\|a\| = 5$, oriented at 30° with the horizontal, and we took $\nu = 1$, see Figure 15. Homogeneous Dirichlet boundary conditions were assumed on the whole boundary.

Table III. Obtained ‘errors’ with respect to the 100×100 SUPG-stabilized finite element solution.

Formulation	1	2	FEM–SUPG
$\ e\ _{L_2}$	0.016660	0.0101530	0.085855
$\ e\ _{\mathcal{H}^1}$	1.016674	0.999251	1.09428

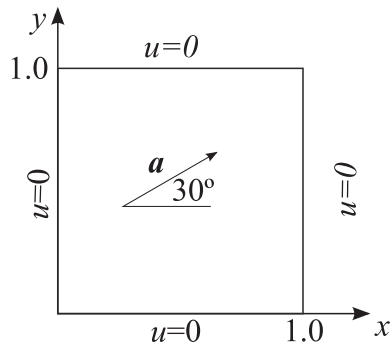


Figure 15. Geometry of problem 3.

Table IV. Obtained ‘errors’ with respect to the 100×100 SUPG-stabilized finite element solution for problem 3.

Formulation	1	2	FEM–SUPG
$\ e\ _{L_2}$	0.009163	0.009871	0.1596
$\ e\ _{\mathcal{H}^1}$	1.159664	1.009154	1.159664

Again, a 100×100 finite element mesh, with standard SUPG stabilization, was considered as a reference solution. In this example, the differences between the proposed method and the equivalent in finite elements are more evident. Table IV resumes these differences. The reference solution and the obtained approximations for formulations 1 and 2 are depicted in Figure 16. Note that the PGD approaches to the problem given in this example has a remarkable difference in accuracy with the standard, SUPG-stabilized finite element approach. Note also that the first version of section 4 gives more accurate results in the \mathcal{L}_2 -norm than that in section 5.

Also noticeable is the number of functions needed for attaining such level of error. The algorithm coined as ‘version 2’ needed only one pair of functions to give the error shown in Table IV. For one pair of functions, version 1 of the method gave an \mathcal{L}_2 -norm error of 0.01645, still one order of magnitude less than the stabilized FEM.

7. A HIGH DIMENSIONAL EXAMPLE

To fully justify not only the behavior of the proposed technique, which has been thoroughly addressed in the previous sections, but also its true advantages for high-dimensional problems, we consider here a problem defined in \mathbb{R}^5 .

The problem is defined in the hypercube $(x, y, z, t, s) \in [0, 1]^5$, with $\mathbf{a} = (1, 1, 1, 1, 1)$. All boundary hyperplanes were subjected to Dirichlet boundary conditions of the same type: $u = 0$ at $x, z, t, s = 0$, $u = 1$ at $x, z, t, s = 1$, whereas $u = 1$ at $y = 0$ and $u = 0$ at $y = 1$. A finite element mesh of ten elements per edge of the hypercube was employed. The restricted number of elements along each direction is kept here for the purpose of highlighting the stabilization characteristics of the proposed techniques rather than its ability to cope with the curse of dimensionality, which has been addressed in previous works, see [9], for instance.

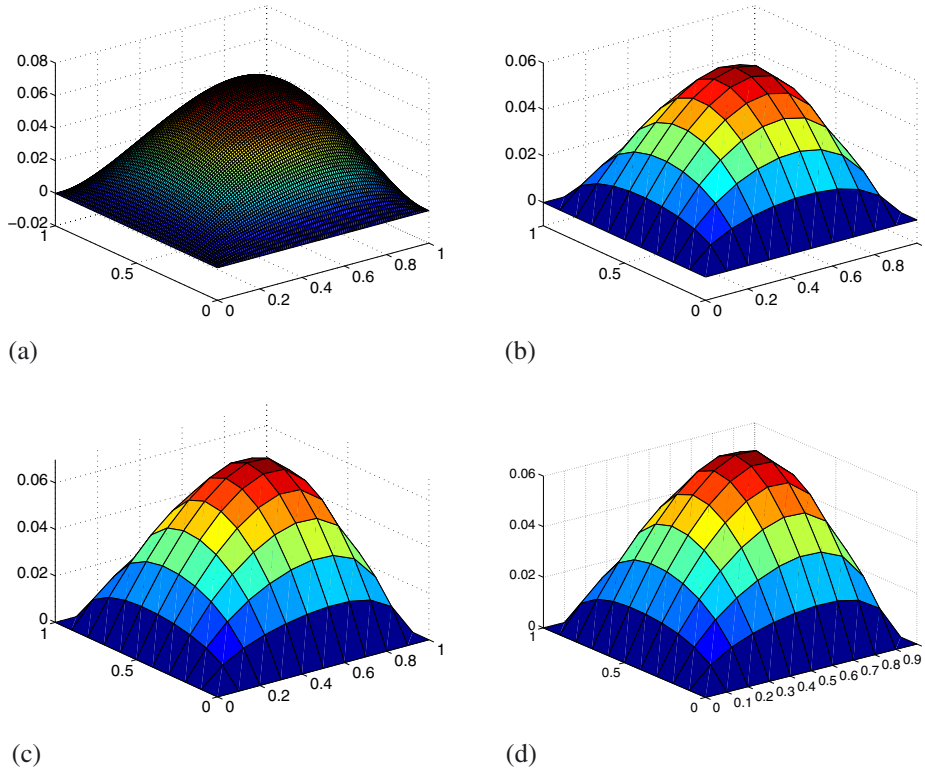


Figure 16. Comparison between the obtained results by (a) FEM-SUPG (100×100 elements, reference solution), (b) FEM-SUPG (10×10 elements) (c) formulation 1, and (d) formulation 2.

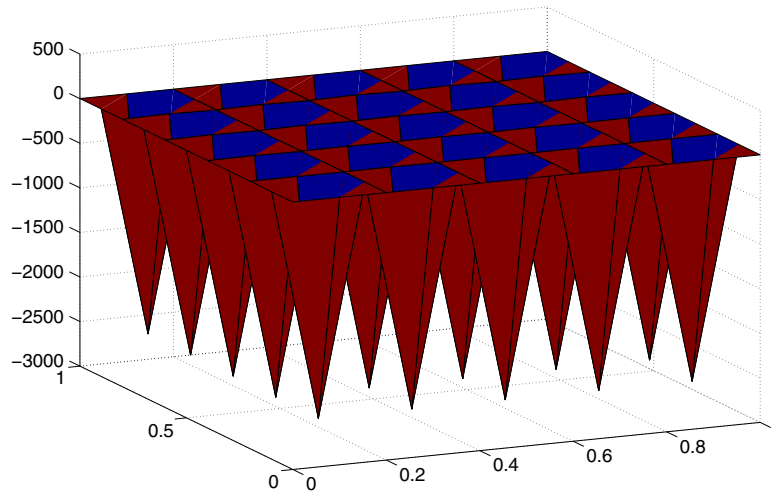


Figure 17. Non-stabilized results for the 5D problem.

Obviously, the representation of results in a 5D domain is not easy. We have chosen to represent the results at the plane (x, y) located at $z, t, s = 0.5$. Obviously, the solution without stabilization is highly oscillating, as expected, see Figure 17.

On the contrary, results for the two different stabilization techniques presented in Sections 4 and 5, respectively, are shown in Figure 18. Functions employed to approximate the solution along each spatial direction (for the formulation in Section 4 in this case) are depicted in Figure 19.

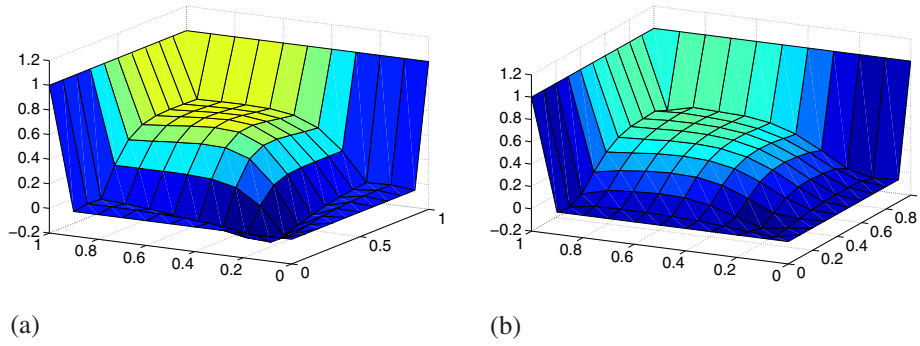


Figure 18. Comparison between the obtained results by (a) formulation 1 and (b) formulation 2.

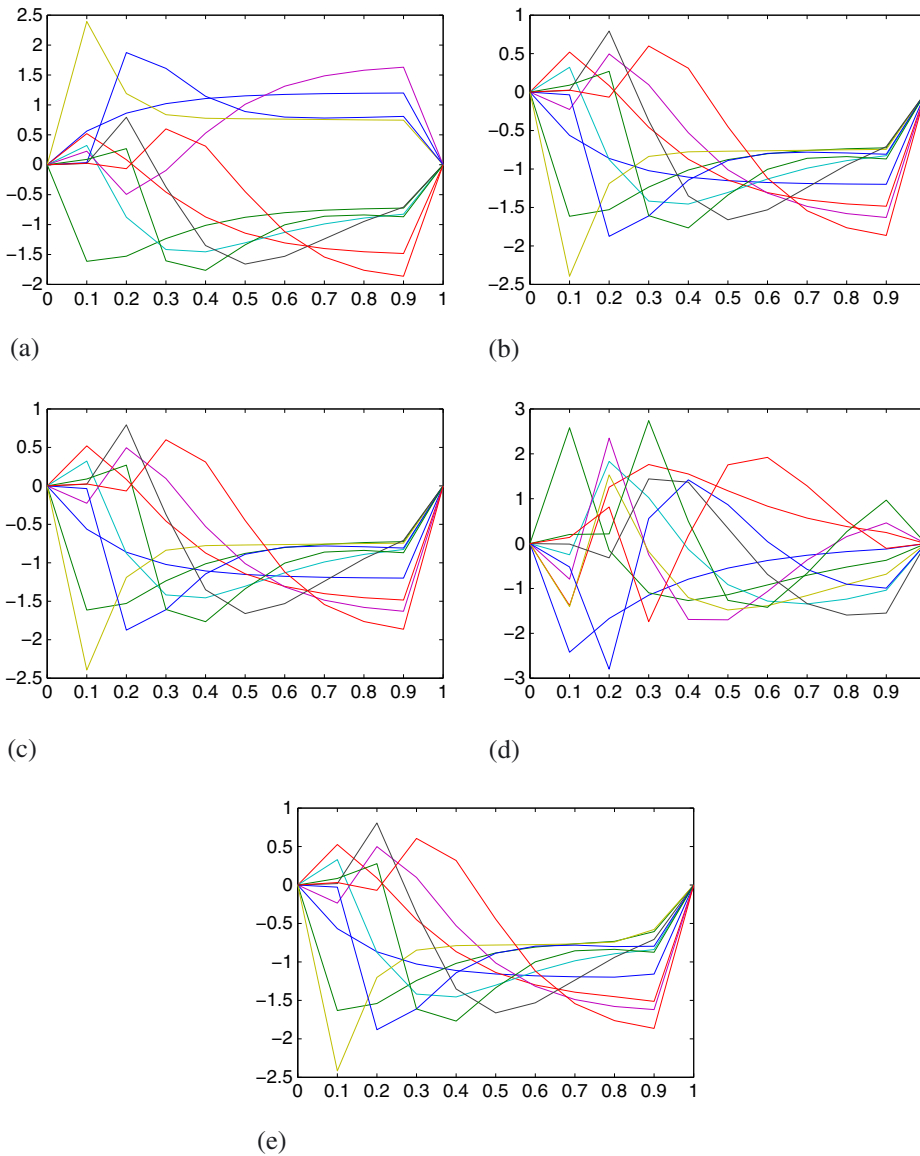


Figure 19. Separated functions employed to approximate the solution of the problem defined in 5D. In this case, these functions correspond to the formulation developed in Section 4.

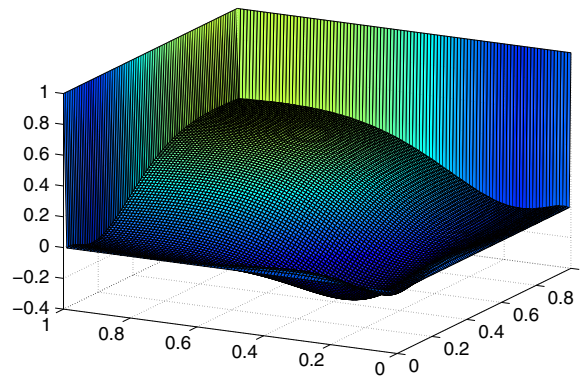


Figure 20. Results at $z = t = s = 0.5$ for the problem defined in 5D. Formulation developed in Section 5.

These same results can be now compared with those obtained by employing a mesh of 100 elements along each edge of the hypercube. This would make a total of $100^5 = 10^{10}$ elements if traditional finite element methods would have been employed. Obviously, this is out of reach for a standard personal computer (this is a practical consequence of the curse of dimensionality mentioned in the introduction). However, the result in Figure 20 has been obtained by means of PGD approximations in less than 2 minutes in a laptop running MATLAB. Mathworks, Inc. 3 Apple Hill Drive Natick, MA 01760-2098 UNITED STATES.

8. CONCLUSIONS

The PGD techniques allow for an efficient means of a priori model reduction. While constructing alternatively a reduced approximation to the problem by means of products of separable functions, PGD techniques have been devised as an efficient method to deal with the so-called *curse of dimensionality*, that is, the exponential growth of the number of degrees of freedom for mesh-based discretization techniques applied to problems defined in highly-dimensional spaces.

Fokker–Planck equations, for instance, constitute a typical example of such difficulty, because they are usually defined in highly dimensional spaces. Fokker–Planck equations are a class of conservation (convection–diffusion) equations for the probability density function associated with the time evolution of the *micro*-state variables in multi-scale models of complex fluids.

In this work, we have addressed precisely an analysis of the behavior of PGD techniques in the context of convection–diffusion equations. Two main possibilities arose. The first one, more appealing at a first sight, consists in establishing a separated representation of the infinite-dimensional convection–diffusion equation. This gave rise to a sequence of one-dimensional convection–diffusion–reaction problems that could eventually be stabilized by SUPG techniques or whatever state-of-the-art stabilization technique suitable for this class of problems. SUPG techniques present the advantage of exact stabilization (at the nodes) for one-dimensional convection–reaction problems solved with linear finite elements. Thus, PGD techniques seemed to be specially well-suited for this problem, not only providing a reduced modeling technique for the problem but also attaining exact stabilization in any number of dimensions. This approach gave good results, as discussed in the text.

The second approach consisted in applying the PGD technique to the *already* stabilized problem and showed a similar degree of accuracy as formulation 1. The price to pay, however, is the multidimensional (and therefore not exact) stabilizing term to be added to the equation before constructing the approximation. This technique, however, greatly exploited its dimensional reduction capabilities and showed great promise for its extension to highly-dimensional problems. For the examples included in this paper, only 20 terms were enough for approximating the problem with a resolution that would make necessary to employ 10^{10} degrees of freedom with standard finite elements.

ACKNOWLEDGEMENTS

This work is supported by the Spanish Ministry of Economy and Innovation CICYT-DPI2011-27778-C02-01/02.

REFERENCES

1. Keunings R. Micro–macro methods for the multiscale simulation of viscoelastic flow using molecular models of kinetic theory. In *Rheology Reviews 2004*, Binding DM, Walters K (eds). British Society of Rheology, 2004; 67–98.
2. Owens RG, Phillips TN. *Computational Rheology*. Imperial College Press: London, 2002.
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. 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.
5. Chinesta F, Ammar A, Cueto E. Recent advances in the use of the proper generalized decomposition for solving multidimensional models. *Archives of Computational Methods in Engineering* 2010; **17**(4):327–350.
6. 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**:191–213.
7. Chinesta F, Ammar A, Lemarchand F, Beauchene P, Boust F. Alleviating mesh constraints: model reduction, parallel time integration and high resolution homogenization. *Computer Methods in Applied Mechanics and Engineering* 2008; **197**:400–413.
8. Chinesta F, Ladeveze P, Cueto E. A short review on model order reduction based on proper generalized decomposition. *Archives of Computational Methods in Engineering* 2011; **18**:395–404.
9. González D, Ammar A, Chinesta F, Cueto E. Recent advances on the use of separated representations. *International Journal for Numerical Methods in Engineering* 2010; **81**(5):637–659.
10. Karhunen K. Über lineare methoden in der wahrscheinlichkeitsrechnung. *Annales Academiae Scientiarum Fennicae* 1946; **37**:1–79.
11. Loève MM. *Probability Theory* (3rd edn.), The University Series in Higher Mathematics. Van Nostrand: Princeton, NJ, 1963.
12. Lorenz EN. *Empirical Orthogonal Functions and Statistical Weather Prediction*. MIT, Departement of Meteorology, Scientific Report Number 1, Statistical Forecasting Project: Cambridge, MA, USA, 1956.
13. Cancès E, Defranceschi M, Kutzelnigg W, Le Bris C, Maday Y. Computational quantum chemistry: a primer. In *Handbook of numerical analysis*, Vol. X, Ph. Ciarlet, C. Le Bris (eds): North-Holland, 2003; 3–270.
14. Ladeveze P. *Nonlinear Computational Structural Mechanics*. Springer: NY, 1999.
15. Nouy A. A priori model reduction through proper generalized decomposition for solving time-dependent partial differential equations. *Computer Methods in Applied Mechanics and Engineering* 2010; **199**(23-24):1603–1626.
16. Donea J, Huerta A. *Finite Element Methods for Flow Problems*. J. Wiley and Sons: Chichester, West Sussex, England, 2002.
17. Christie I, Griffiths DF, Mitchell AR, Zienkiewicz OC. Finite element methods for second order differential equations with significant first derivatives. *International Journal for Numerical Methods in Engineering* 1976; **10**(6):1389–1396.
18. Heinrich JC, Huyakorn PS, Zienkiewicz OC, Mitchell AR. An ‘upwind’ finite element scheme for two-dimensional convective transport equation. *International Journal for Numerical Methods in Engineering* 1977; **11**(1):131–143.
19. Hughes TJR, Brooks AN. A multidimensional upwind scheme with no crosswind diffusion. In *Finite Element Methods for Convection Dominated Flows*. AMD, Vol. 34, Hughes TJR (ed.). American Society of Mechanical Engineering: New York, 1979.
20. Hughes TJR, Feijóo GR, Mazzei L, Quincy J-B. The variational multiscale method—a paradigm for computational mechanics. *Computer Methods in Applied Mechanics and Engineering* 1998; **166**(1–2):3–24.
21. Ryckelynck D, Chinesta F, Cueto E, Ammar A. On the a priori model reduction: overview and recent developments. *Archives of Computational Methods in Engineering* 2006; **12**(1):91–128.
22. Huerta A, Fernández-Méndez S. Time accurate consistently stabilized mesh-free methods for convection-dominated problems. *International Journal for Numerical Methods in Engineering* 2003; **59**(9):1225–1242.