



HAL
open science

Advanced simulation of models defined in plate geometries: 3D solutions with 2D computational complexity

Brice Bognet, Felipe Bordeu, Francisco Chinesta, Adrien Leygue, Arnaud Poitou

► To cite this version:

Brice Bognet, Felipe Bordeu, Francisco Chinesta, Adrien Leygue, Arnaud Poitou. Advanced simulation of models defined in plate geometries: 3D solutions with 2D computational complexity. *Computer Methods in Applied Mechanics and Engineering*, 2012, 201, pp.1-12. 10.1016/j.cma.2011.08.025 . hal-01462825

HAL Id: hal-01462825

<https://hal.science/hal-01462825>

Submitted on 9 Feb 2017

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

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



Distributed under a Creative Commons Attribution 4.0 International License

Advanced simulation of models defined in plate geometries: 3D solutions with 2D computational complexity

B. Bognet, F. Bordeu, F. Chinesta, A. Leygue, A. Poitou

GEM UMR CNRS – Ecole Centrale de Nantes, EADS Corporate Foundation International Chair, 1 rue de la Noe, BP 92101, F-44321 Nantes cedex 3, France

Many models in polymer processing and composites manufacturing are defined in degenerated three-dimensional domains, involving plate or shell geometries. The reduction of models from 3D to 2D is not obvious when complex physics are involved. The hypotheses to be introduced for reaching this dimensionality reduction are sometimes unclear, and most of possible proposals will have a narrow interval of validity. The only getaway is to explore new discretization strategies able to circumvent or at least alleviate the drawbacks related to mesh-based discretizations of fully 3D models defined in plate or shell domains. An in-plane–out-of-plane separated representation of the involved fields within the context of the Proper Generalized Decomposition allows solving the fully 3D model by keeping a 2D characteristic computational complexity. Moreover the PGD features allow the introduction of many extra-coordinates, as for example the orientation of the different laminate plies, without affecting the solvability of the resulting multidimensional model.

Keywords: PGD, Separated representations, Model reduction, Plate theory

1. Introduction

Many models in polymer processing and composites manufacturing are defined in degenerated three-dimensional domains. By degenerated we understand that at least one of the characteristic dimensions of the domain is much lower than the other ones. This situation is particularly common in models defined plate or shells type geometries.

Mesh based solutions of models defined in such degenerated domains is a challenging issue because the resulting meshes usually involve too many degrees of freedom. In that case the first question concerns the possibility of reducing the model complexity. Classical beam, plate or shell theories are some examples of simplified modeling where the 3D subjacent elastic model is substituted by lower dimensional models (1D in the case of beam theory and 2D in the case of plate and shell theories).

Going from a 3D elastic problem to a 2D plate theory model usually involves some kinematical and/or mechanical hypotheses [19] on the evolution of the solution through the thickness of the plate. Despite the quality of existing plate theories, their solution close to the plate edges is usually wrong as the displacement fields are truly 3D in those regions and do not satisfy the kinematic hypothesis. Indeed, the kinematic hypothesis is a good approximation where Saint-Venant's principle is verified. However, some het-

erogeneous complex plates do not verify the Saint Venant's principle anywhere. In that case the solution of the three dimensional model is mandatory even if its computational complexity could be out of the nowadays calculation capabilities.

The Saint Venant's principle was extensively used in the Ladeveze's works for defining elegant and efficient 3D simplified models [9]. This technique was then generalized to dynamics [11].

In the case of elastic behaviors the derivation of such 2D plate theory models is quite simple and it constitutes the foundations of classical plate and shell theories. Today, most commercial codes for structural mechanics applications propose different type of plate and shell finite elements, even in the case of multilayered composites plates or shells. However, in composites manufacturing processes the physics encountered in such multilayered plate or shell domains is much richer, because it usually involves chemical reactions, crystallization and strongly coupled and non-linear thermo-mechanical behaviors [1]. The complexity of the involved physics makes impossible the introduction of pertinent hypotheses for reducing *a priori* the dimensionality of the model from 3D to 2D. In that case a fully 3D modeling is compulsory, and because the richness of the thickness description (many coupled physics and many plies with different physical states and directions of anisotropy) the approximation of the fields involved in the models needs thousands of nodes distributed along the thickness direction. Thus, fully 3D descriptions may involve millions of degrees of freedom that should be solved many times because the history dependent thermomechanical behavior. Moreover, when we are considering optimization or inverse identification, many direct problems have to be solved in order to reach the minimum of a certain cost function.

Today, the solution of such fully 3D models remains intractable despite the impressive progresses reached in mechanical modeling, numerical analysis, discretization techniques and computer science during the last decade. New numerical techniques are needed for approaching such complex scenarios, able to proceed to the solution of fully 3D multiphysics models in geometrically complex parts (e.g. a whole aircraft). The well established mesh-based discretization techniques fail because the excessive number of degrees of freedom involved in the fully 3D discretizations where very fine meshes are required in the thickness direction (despite its reduced dimension) and also in the in-plane directions to avoid too distorted meshes.

Thus, 3D solutions seem mandatory in many cases, however such solutions are not obvious because the numerical complexity that mesh based discretizations imply. Thus, new approaches able to address the efficient solutions of such models is required. In this manuscript we propose the application of the model reduction method known as Proper Generalized Decomposition – PGD – to the simulation of 3D thermomechanical models defined in plate geometries. This method is based on the use of separated representations. A space–time separated representation was originally proposed in the 80s by Pierre Ladeveze as one of the main ingredients of the LATIN (non-linear and non-incremental solver) and which was called “radial approximation” (the interested reader can refer to the Ladeveze works [10,12,13] the references therein). Then, this kind of separated representation was considered in the context of stochastic modeling by Nouy [16,17] as well as for addressing multidimensional models [2,3].

The separated representation basically consists in constructing by successive enrichment an approximation of the solution (defined in a space of dimension d) in the form of a finite sum of N functional products involving d functions of each coordinate. In contrast with the shape functions of classical discretization methods, these individual functions are unknown *a priori*. They are obtained by introducing the approximate separated representation into the weak formulation of the original problem and solving the resulting non-linear equations. If M nodes are used to discretize each coordinate, the total number of unknowns amounts to $N \times M \times d$ instead of the M^d degrees of freedom of classical mesh-based methods. Thus, the complexity of the method grows linearly with the dimension d of the space wherein the problem is defined, in vast contrast with the exponential growth of classical mesh-based techniques.

This strategy was successfully applied in our studies of the kinetic theory description of complex fluids. A multidimensional separated representation of the linear steady-state Fokker–Planck equation was introduced in the seminal work [2], further extended to transient simulations in [3] and non-linear Fokker–Planck equations in [14]. In [15,18], we considered the solution of Fokker–Planck equations in complex flows, where space, time and conformation coordinates coexist. We have also applied the same approach for solving the Schrödinger equation [5], the chemical master equation [6] or kinetic theory models formulated within the Brownian Configurations Fields framework [4]. For other applications the interested reader can refer to the review paper [7].

The fully three-dimensional solution of models defined in degenerate domains is also an appealing field of application of the PGD. Consider the unknown field $u(\mathbf{x}, t)$ defined in a plate domain \mathcal{E} . Two approaches come to mind:

- Complete decomposition:

$$u(\mathbf{x}, t) \approx \sum_{i=1}^N X_i(x) \cdot Y_i(y) \cdot Z_i(z) \cdot T_i(t). \quad (1)$$

This strategy is particularly suitable for separable domains, i.e. $\mathcal{E} = \Omega_x \times \Omega_y \times \Omega_z$. For general domains, embedding \mathcal{E} into a larger separable domain $\Omega_x \times \Omega_y \times \Omega_z$ can also be done, as described in [8].

- Plate-type decomposition:

$$u(\mathbf{x}, t) \approx \sum_{i=1}^N X_i(x, y) \cdot Z_i(z) \cdot T_i(t). \quad (2)$$

This strategy is particularly suited when $\mathcal{E} = \Omega \times \mathcal{I}$, with $\Omega \subset \mathcal{R}^2$ and $\mathcal{I} \subset \mathcal{R}$. More complex domains (e.g. plates with a varying thickness) can be treated using an appropriate change of variable. Because such decomposition involves the calculation of 2D functions $X_i(x, y)$ and 1D functions $Z_i(z)$ (these ones with a computational complexity negligible with respect to the computation of the 2D functions) we can conclude that the computational complexity of the fully 3D solution is of the same order of magnitude than the solution of 2D models, justifying the manuscript title.

We would like to emphasize that this paper does not concern a new plate theory proposal. This paper concerns the proposal of a new solution procedure able to compute efficiently fully 3D solutions of any model defined in plate domains whose numerical complexity reduces to the one characteristic of 2D solvers. It is important to notice that the results computed by applying the in-plane–out-of-plane separated representation can be only compared with the ones coming from standard 3D solutions, but not with the ones computed using classical plate or shell theories whose accuracy depends on the validity of the hypotheses introduced during the derivation of such simplified theories. The accuracy of our strategy must be evaluated by comparing the computed solution with the reference one, that could be obtained for example by using an accurate enough 3D finite element solution. The comparison of the solution computed by using our strategy and the ones obtained using different plate theories has no sense because we cannot consider these solutions as reference solutions (they are subjected to many hypotheses that fail, as argued above, in many cases). Solutions obtained by using a plate theory should be compared with fully 3D solutions to conclude on its accuracy, but in this work we are not concerned by such comparisons. In summary, we are not elaborating an alternative plate theory, but simply proposing a new algorithm for computing fully 3D solutions in degenerated plate domains, with a computational complexity characteristic of 2D solvers.

Undoubtedly, the connections between the PGD and different plate theories should be explored deeply. As we show later, in some circumstances, the first mode of the PGD solution has important resemblances with the solution obtained by applying standard plate theories, Mindlin or Kirchhoff depending on the plate thickness. Additional modes come to represent 3D effects that a simplified modeling (simple plate theories) is not able to capture. A deep analysis of all the connections existing between the PGD modes and standard and advanced plate theories constitutes a work in progress. This analysis could inspire new hypothesis to be introduced in advanced plate theories. Another interest of such a connection is defining bridges between plate models and fully 3D PGD descriptions to capture locally 3D effects in multidomain decompositions or multiscale frameworks. The first preliminary results of this study are extremely promising.

In the next section, the PGD is applied to perform an in-plane–out-of-plane separated representation for the steady state heat equation defined in a multilayered plate. The strategy is then generalized to elastic behaviors in Section 3. The accuracy and the numerical efficiency of the method are analyzed showing its full potential for the treatment of more complex composite structures. Finally, a parametric modeling case is addressed, in which the orientation of the different laminate plies is considered as extra-coordinates. As soon as the parametric solution is computed, only once and off-line, it can be particularized on-line for different values of the plies orientation on light computing platforms, as for example

smartphones. Some examples illustrating these capabilities are addressed in the last section of this paper.

2. Illustrating the in-plane–out-of-plane separated representation of a multilayered plate

In what follows we are illustrating the construction of the Proper Generalized Decomposition of a generic model defined in a plate domain $\Xi = \Omega \times \mathcal{I}$ with $\Omega \subset \mathcal{R}^2$ and $\mathcal{I} = [0, H] \subset \mathcal{R}$. For the sake of simplicity we consider the model related to the steady state heat conduction equation:

$$\nabla \cdot (\mathbf{K} \cdot \nabla u) = 0, \quad (3)$$

in a plate geometry that contains a P plies in the plate thickness. Each ply is characterized by given conductivity tensor $\mathbf{K}_i(x, y)$ which is assumed constant through the ply thickness. Moreover, without any loss of generality, we assume the same thickness h for the different layers of the laminate. Thus, we can define a characteristic function representing the position of each layer $i = 1, \dots, P$:

$$\xi_i(z) = \begin{cases} 1, & z_i \leq z \leq z_{i+1}, \\ 0, & \text{otherwise,} \end{cases} \quad (4)$$

where $z_i = (i-1) \cdot h$ defines the location of ply i in the laminate thickness. Now, the laminated conductivity can be given in the following separated form:

$$\mathbf{K}(x, y, z) = \sum_{i=1}^{i=P} \mathbf{K}_i(\mathbf{x}) \cdot \xi_i(z), \quad (5)$$

where \mathbf{x} denotes the in-plane coordinates, i.e. $\mathbf{x} = (x, y) \in \Omega$. The specific boundary conditions imposed in this problem are not relevant for illustrating the in-plane–out-of-plane separated representation. The issue related to the enforcement of boundary conditions will be addressed in the next section and was also the main topic of [8].

The weak form of Eq. (3), with appropriate boundary conditions, writes:

$$\int_{\Xi} \nabla u^* \cdot (\mathbf{K} \cdot \nabla u) d\Xi = 0, \quad (6)$$

with the test function u^* defined in an appropriate functional space. The solution $u(x, y, z)$ is searched under the separated form:

$$u(\mathbf{x}, z) \approx \sum_{j=1}^{j=N} X_j(\mathbf{x}) \cdot Z_j(z). \quad (7)$$

In what follows we are illustrating the construction of such a decomposition. For this purpose we assume that at enrichment step $n < N$ the solution $u^n(\mathbf{x}, z)$ is already known:

$$u^n(\mathbf{x}, z) = \sum_{j=1}^{j=n} X_j(\mathbf{x}) \cdot Z_j(z), \quad (8)$$

and that at the present step $n+1$ we look for the solution enrichment $R(\mathbf{x}) \cdot S(z)$:

$$u^{n+1}(\mathbf{x}, z) = u^n(\mathbf{x}, z) + R(\mathbf{x}) \cdot S(z). \quad (9)$$

The test function involved in the weak form is searched under the form:

$$u^*(\mathbf{x}, z) = R^*(\mathbf{x}) \cdot S(z) + R(\mathbf{x}) \cdot S^*(z). \quad (10)$$

By introducing Eqs. (9) and (10) into Eq. (6) it results:

$$\begin{aligned} \int_{\Xi} \left(\left(\frac{\tilde{\nabla} R^* \cdot S}{R^* \cdot \frac{dS}{dz}} \right) + \left(\frac{\tilde{\nabla} R \cdot S^*}{R \cdot \frac{dS^*}{dz}} \right) \right) \cdot \left(\mathbf{K} \cdot \left(\frac{\tilde{\nabla} R \cdot S}{R \cdot \frac{dS}{dz}} \right) \right) d\Xi \\ = - \int_{\Xi} \left(\left(\frac{\tilde{\nabla} R^* \cdot S}{R^* \cdot \frac{dS}{dz}} \right) + \left(\frac{\tilde{\nabla} R \cdot S^*}{R \cdot \frac{dS^*}{dz}} \right) \right) \cdot \mathbf{Q}^n d\Xi, \end{aligned} \quad (11)$$

where $\tilde{\nabla}$ denotes the plane component of the gradient operator, i.e. $\tilde{\nabla}^T = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y} \right)$ and \mathbf{Q}^n denotes the flux at iteration n :

$$\mathbf{Q}^n = \mathbf{K} \cdot \sum_{j=1}^{j=n} \frac{\tilde{\nabla} X_j(\mathbf{x}) \cdot Z_j(z)}{X_j(\mathbf{x}) \cdot \frac{dZ_j(z)}{dz}}. \quad (12)$$

Now, as the enrichment process is non-linear we propose to search the couple of functions $R(\mathbf{x})$ and $S(z)$ by applying an alternating direction fixed point algorithm. Thus, assuming $R(\mathbf{x})$ known, we compute $S(z)$, and then we update $R(\mathbf{x})$. The process continues until reaching convergence. The converged solutions allow defining the next term in the finite sums decomposition: $R(\mathbf{x}) \rightarrow X_{n+1}(\mathbf{x})$ and $S(z) \rightarrow Z_{n+1}(z)$.

We are illustrating each one of the just referred steps:

- Computing $R(\mathbf{x})$ from $S(z)$.

When $S(z)$ is known the test function reduces to:

$$u^*(\mathbf{x}, z) = R^*(\mathbf{x}) \cdot S(z), \quad (13)$$

and the weak form (11) reduces to:

$$\int_{\Xi} \left(\frac{\tilde{\nabla} R^* \cdot S}{R^* \cdot \frac{dS}{dz}} \right) \cdot \left(\mathbf{K} \cdot \left(\frac{\tilde{\nabla} R \cdot S}{R \cdot \frac{dS}{dz}} \right) \right) d\Xi = - \int_{\Xi} \frac{\tilde{\nabla} R^* \cdot S}{R^* \cdot \frac{dS}{dz}} \cdot \mathbf{Q}^n d\Xi. \quad (14)$$

Now, as all the functions involving the coordinate z are known, they can be integrated over $\mathcal{I} = [0, H]$. Thus, if we consider:

$$\mathbf{K} = \begin{pmatrix} \mathbb{K} & \mathbf{k} \\ \mathbf{k}^T & \kappa \end{pmatrix}, \quad (15)$$

with

$$\mathbb{K} = \begin{pmatrix} \mathbf{K}_{xx} & \mathbf{K}_{xy} \\ \mathbf{K}_{xy} & \mathbf{K}_{yy} \end{pmatrix}, \quad (16)$$

$$\mathbf{k} = \begin{pmatrix} \mathbf{K}_{xz} \\ \mathbf{K}_{yz} \end{pmatrix} \quad (17)$$

and $\kappa = \mathbf{K}_{zz}$, then we can define:

$$\mathbf{K}^{\mathbf{x}} = \begin{pmatrix} \int_{\mathcal{I}} \mathbb{K} \cdot S^2 dz & \int_{\mathcal{I}} \mathbf{k} \cdot \frac{dS}{dz} \cdot S dz \\ \int_{\mathcal{I}} \mathbf{k}^T \cdot \frac{dS}{dz} \cdot S dz & \int_{\mathcal{I}} \kappa \cdot \left(\frac{dS}{dz} \right)^2 dz \end{pmatrix} \quad (18)$$

and

$$(\mathbf{Q}^{\mathbf{x}})^n = \sum_{j=1}^{j=n} \left(\left(\int_{\mathcal{I}} \mathbb{K} \cdot S \cdot Z_j dz \quad \int_{\mathcal{I}} \mathbf{k} \cdot \frac{dZ_j}{dz} \cdot S dz \right) \cdot \frac{\tilde{\nabla} X_j(\mathbf{x})}{X_j(\mathbf{x})} \right) \quad (19)$$

that allows writing Eq. (14) into the form:

$$\int_{\Omega} \left(\frac{\tilde{\nabla} R^*}{R^*} \right) \cdot \left(\mathbf{K}^{\mathbf{x}} \cdot \left(\frac{\tilde{\nabla} R}{R} \right) \right) d\Omega = - \int_{\Omega} \left(\frac{\tilde{\nabla} R^*}{R^*} \right) \cdot (\mathbf{Q}^{\mathbf{x}})^n d\Omega, \quad (20)$$

that defines an elliptic 2D problem defined in Ω .

- Computing $S(z)$ from $R(\mathbf{x})$.

When $R(\mathbf{x})$ is known the test function writes:

$$p^*(\mathbf{x}, z) = R(\mathbf{x}) \cdot S^*(z) \quad (21)$$

and the weak form (11) reduces to:

$$\int_{\Xi} \left(\frac{\tilde{\nabla} R \cdot S^*}{R \cdot \frac{dS^*}{dz}} \right) \cdot \left(\mathbf{K} \cdot \left(\frac{\tilde{\nabla} R \cdot S}{R \cdot \frac{dS}{dz}} \right) \right) d\Xi = - \int_{\Xi} \left(\frac{\tilde{\nabla} R \cdot S^*}{R \cdot \frac{dS^*}{dz}} \right) \cdot \mathbf{Q}^n d\Xi. \quad (22)$$

Now, as all the functions involving the in-plane coordinates $\mathbf{x} = (x, y)$ are known, they can be integrated over Ω . Thus, using

the previous notation we can define:

$$\mathbf{K}^z = \begin{pmatrix} \int_{\Omega} (\tilde{\nabla} R) \cdot (\mathbb{K} \cdot \tilde{\nabla} R) d\Omega & \int_{\Omega} (\tilde{\nabla} R) \cdot \mathbf{k} \cdot R d\Omega \\ \int_{\Omega} (\tilde{\nabla} R) \cdot \mathbf{k} \cdot R d\Omega & \int_{\Omega} \kappa \cdot R^2 d\Omega \end{pmatrix} \quad (23)$$

and

$$(\mathbf{Q}^z)^n = \sum_{j=1}^{j=n} \left(\begin{pmatrix} \int_{\Omega} (\tilde{\nabla} R) \cdot (\mathbb{K} \cdot \tilde{\nabla} X_j) d\Omega & \int_{\Omega} (\tilde{\nabla} R) \cdot \mathbf{k} \cdot X_j d\Omega \\ \int_{\Omega} (\tilde{\nabla} X_j) \cdot \mathbf{k} \cdot R d\Omega & \int_{\Omega} \kappa \cdot X_j \cdot R d\Omega \end{pmatrix} \cdot \begin{pmatrix} Z_j(z) \\ \frac{dZ_j}{dz}(z) \end{pmatrix} \right), \quad (24)$$

that allows writing Eq. (22) into the form:

$$\int_{\mathcal{I}} \left(\frac{dS^*}{dz} \right) \cdot \left(\mathbf{K}^z \cdot \begin{pmatrix} S \\ \frac{dS}{dz} \end{pmatrix} \right) dz = - \int_{\mathcal{I}} \left(\frac{dS^*}{dz} \right) \cdot (\mathbf{Q}^z)^n dz \quad (25)$$

that defines a one-dimensional elliptic boundary value problem (BVP).

At each enrichment step the construction of the new functional product requires iterations. If m_i denotes the number of iterations needed at enrichment step i , the total number of iterations involved in the construction of the PGD approximation is $m = \sum_{i=1}^N m_i$. The computational cost of the entire procedure is therefore dominated by the m two-dimensional problems for the functions $X_i(\mathbf{x})$ as the m one-dimensional problems for the functions $Z_i(z)$ have a negligible cost. In general, m_i rarely exceeds 10. The number N of functional products needed to approximate the solution with enough accuracy depends on the solution's regularity. All numerical experiments carried to date reveal that N ranges between a few tens and one hundred, depending on the complexity of the problem. Thus, we can conclude that the complexity of the PGD procedure to compute the problem solution is of some tens of 2D problems only.

3. Fully 3D simulation of mechanical models defined in plate domains

In this section, we apply the PGD method to the simulation of the linear elastic behavior of plates-shaped domains. The PGD method allows us to separately search for the in-plane and the out-of-plane contributions to the fully 3D solution, allowing significant savings in computing time and memory resources. The method is first validated on a simple case and its full potential is then presented for the simulation of the behavior of multilayered composite plates and honeycomb core plates.

We assume the following separated representation for the displacement field:

$$\mathbf{u}(x, y, z) = \begin{pmatrix} u(x, y, z) \\ v(x, y, z) \\ w(x, y, z) \end{pmatrix} \approx \sum_{i=1}^N \begin{pmatrix} u_{xy}^i(x, y) \cdot u_z^i(z) \\ v_{xy}^i(x, y) \cdot v_z^i(z) \\ w_{xy}^i(x, y) \cdot w_z^i(z) \end{pmatrix}, \quad (26)$$

where $u_{xy}(x, y)$, $v_{xy}(x, y)$ and $w_{xy}(x, y)$ are functions of the in-plane coordinates whereas $u_z(z)$, $v_z(z)$ and $w_z(z)$ are functions involving the thickness coordinate.

Because neither the number of terms in the separated representation of the displacement field nor the dependence on z of functions $u_z^i(z)$, $v_z^i(z)$ and $w_z^i(z)$ are assumed *a priori*, the approximation is flexible enough for representing the fully 3D solution, being obviously more general than classical plate theories that assume particular *a priori* behaviors in the thickness direction [19].

Let us consider a linear elasticity problem on a plate domain $\Xi = \Omega \times \mathcal{I}$. The weak formulation associated to such a problem reads:

$$\int_{\Xi} \epsilon(\mathbf{u}^*) \cdot \mathbf{K} \cdot \epsilon(\mathbf{u}) d\Xi = \int_{\Xi} \mathbf{u}^* \cdot \mathbf{f}_d d\Xi + \int_{\Gamma_N} \mathbf{u}^* \cdot \mathbf{F}_d d\Gamma, \quad (27)$$

with \mathbf{K} the generalized 6×6 Hooke tensor, \mathbf{f}_d represents the volumetric body forces while \mathbf{F}_d represents the forces applied on the boundary Γ_N . The separation of variables introduced in Eq. (26) yields the following expression for the strain:

$$\epsilon(\mathbf{u}(x, y, z)) \approx \sum_{i=1}^N \begin{pmatrix} u_{xy,x}^i \cdot u_z^i \\ v_{xy,y}^i \cdot v_z^i \\ w_{xy}^i \cdot w_{z,z}^i \\ u_{xy,y}^i \cdot u_z^i + v_{xy,x}^i \cdot v_z^i \\ u_{xy}^i \cdot u_{z,z}^i + w_{xy,x}^i \cdot w_z^i \\ v_{xy}^i \cdot v_{z,z}^i + w_{xy,y}^i \cdot w_z^i \end{pmatrix}. \quad (28)$$

Depending of the number of non-zero elements in the \mathbf{K} matrix, the development of $\epsilon(\mathbf{u}^*) \cdot \mathbf{K} \cdot \epsilon(\mathbf{u})$ involves many terms, 21 in the case of an isotropic material and 41 in the case of anisotropic behaviors.

Imposing complex boundary conditions with the PGD is not straightforward, especially in the case of Dirichlet boundary conditions. Neumann-type boundary conditions are easily enforced since they appear naturally in the weak formulation of the elasticity problem as an additional term on the right hand side. The only difficulty might reside in expressing the appropriate boundary flux integrals in the separated form used by the method. This step is most of the time quite straightforward. In the more complex cases one can advantageously make use of the Singular Value Decomposition to express the boundary integrals in the desired form. In the case of Dirichlet boundary conditions, one solution is to make use of the incremental nature of the PGD. When computing the problem solution we *a priori* represent the solution with a set of initial modes that have arbitrary values on the interior of the domain but that are such that their sum satisfies all of the Dirichlet conditions. Later, in the enrichment phase, we impose for the new modes only homogeneous Dirichlet boundary conditions that will not modify the previously satisfied conditions. In [8] we deeply analyzed this question.

Assuming that the first n modes of the solution were already computed, we focus on the solution enrichment related to the computation of the next functional couple, according to:

$$\mathbf{u}^{n+1}(x, y, z) = \mathbf{u}^n(x, y, z) + \begin{pmatrix} R_u(x, y) \cdot S_u(z) \\ R_v(x, y) \cdot S_v(z) \\ R_w(x, y) \cdot S_w(z) \end{pmatrix}. \quad (29)$$

As previously, we consider the test function given by:

$$\mathbf{u}^*(x, y, z) = \begin{pmatrix} R_u^*(x, y) \cdot S_u(z) + R_u(x, y) \cdot S_u^*(z) \\ R_v^*(x, y) \cdot S_v(z) + R_v(x, y) \cdot S_v^*(z) \\ R_w^*(x, y) \cdot S_w(z) + R_w(x, y) \cdot S_w^*(z) \end{pmatrix}. \quad (30)$$

Introducing the trial and test functions given by Eqs. (29) and (30) respectively, into the weak form (27) we obtain:

$$\begin{aligned} & \int_{xy} \int_z [\epsilon(\mathbf{u}^*(x, y, z)) \cdot \mathbf{K} \cdot \epsilon(\mathbf{u}^{n+1}(x, y, z))] dx dy dz \\ &= \int_{xy} \int_z \mathbf{u}^* \cdot \mathbf{f}_d dx dy dz + \int_{\Gamma_N} \mathbf{u}^* \cdot \mathbf{F}_d d\Gamma. \end{aligned} \quad (31)$$

Because the simultaneous calculation of $\mathbf{R}(x, y)$ and $\mathbf{S}(z)$ defines a non-linear problem, a linearization strategy is compulsory. We use here the simple alternating direction fixed point previously described.

Given an initial value $\mathbf{S}^{(0)}(z)$ of $\mathbf{S}(z)$ arbitrarily chosen, all z dependent functions are known. Eq. (31) therefore reduces to a 2D (x and y dependent) problem where the three components of $\mathbf{R}(x, y)$ are the unknown fields. Its solution yields $\mathbf{R}^{(1)}(x, y)$, a first approximation of $\mathbf{R}(x, y)$. Then using the just computed $\mathbf{R}^{(1)}(x, y)$ in Eq. (31) we similarly obtain a 1D problem (z -dependent) which

allows computing the three components of $\mathbf{S}^{(1)}(z)$ that constitutes the next approximation of $\mathbf{S}(z)$. This fixed point loop keeps running until reaching convergence, i.e.:

$$\int_{\Xi} \sum_{i=1}^{i=3} \left(\mathbf{R}_i^{(j)}(x, y) \cdot \mathbf{S}_i^{(j)}(z) - \mathbf{R}_i^{(j-1)}(x, y) \cdot \mathbf{S}_i^{(j-1)}(z) \right)^2 dx dy dz \leq \epsilon, \quad (32)$$

where ϵ is a small enough parameter. Other norms could of course be considered for the stopping criterion. This issue together with the choice of ϵ is open and is still being investigated.

The solution is enriched with new modes until the residual norm becomes less than a fraction of the initial residual norm. The converged separated representation represents the fully 3D solution of the linear elasticity problem defined in the plate domain, whose solution only involves the solution of some 2D problems involving the in-plane coordinates and some 1D problems related to the functions involving the thickness coordinate. Thus, 3D solutions can be computed by keeping a 2D computational complexity.

4. Numerical results

In this section we consider the separated representation based discretization technique – PGD – widely described in the previous sections for solving a variety of test cases in order to validate its accuracy and demonstrate its ability to efficiently handle complex scenarios that would be difficult to solve using classical finite element based 3D discretizations. As explained in the previous sections, the PGD method requires the solution of several two-dimensional and one-dimensional elliptic BVP. For all the numerical results presented below these problems have been solved using a standard Galerkin method.

4.1. Verification test

To validate the proposed technique we consider a square homogeneous plate depicted in Fig. 1 and we compare the classical 3D linear elastic finite element solution and the one obtained by using the PGD with an equivalent discretization, that is, the 2D functions involving the in-plane coordinates in the PGD are approximated using the same mesh that the finite element considered on the plate surface, and the 1D functions involving the thickness coordinate when using the PGD were approximated by using the same number of nodes that was considered in the thickness finite element approximation.

The applied load consists of a uniform pressure applied on the upper face of the plate. The finite element solution was performed by considering a uniform mesh composed of $100 \times 100 \times 50$ 8-nodes hexahedral elements. The PGD solution was performed by using the uniform mesh composed of 100×100 4-nodes elements for approximating the functions involving the in-plane coordinates, whereas a uniform one dimensional mesh composed of 50 2-nodes 1D linear elements were used for approximating the functions involving the thickness coordinate.

The solution computed by using the PGD is depicted in Fig. 1. Nine modes were needed for approximating the solution when using the PGD, most of them to describe the 3D effects that appear in the neighborhood of the boundaries where the displacement was prescribed. In order to compute these 9 terms involved in the separated representation 165 2D and 1D problems were solved. Fig. 2 shows the relative energy density error and the relative error in von Mises stress, considering as reference solution the one computed by using the FEM. The energy density error is everywhere lower than 0.3% except in the vicinity of the plate corners where it reaches a value of 0.57%. As depicted in Fig. 3, this error decreases when considering more terms in the separated representation, before reaching a plateau.

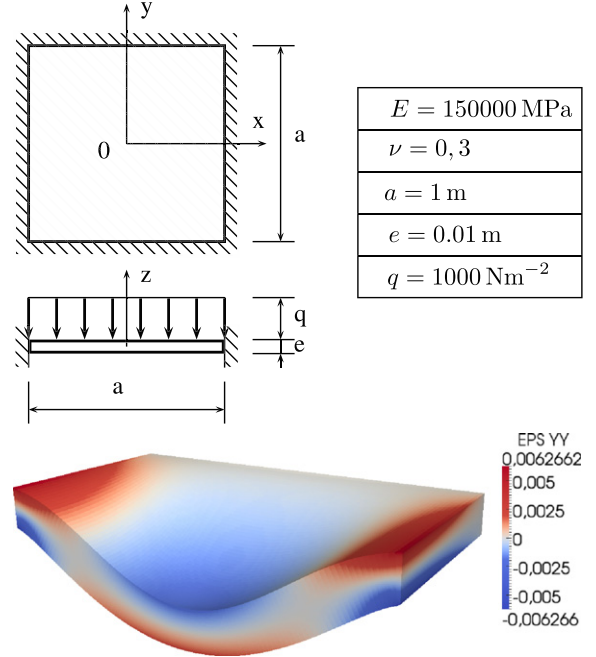


Fig. 1. Problem geometry and computed solution by applying the PGD. Only the solution in a half of the domain is depicted for the sake of clarity.

Fig. 4 compares the CPU time of both the PGD and the FEM based discretizations for solving the linear elasticity problem previously described as a function of the number of in-plane degrees of freedom, $N_x \times N_y$, and of the number of degrees of freedom in the thickness, N_z . We can notice the mild evolution of the computational complexity with the number of in-plane or out-of-plane degrees of freedom when using the PGD instead the exponential growth when using the finite element discretization.

In this simple problem, the edge effects are already present and confirm the necessity of several modes to correctly describe the solution in the boundary neighborhood. In Figs. 5 and 6 we depict respectively the first and the second mode of the FEM solution. The first mode seems to represent classical plate theory solutions because the associated displacements $u_z^1(z)$, $v_z^1(z)$ and $w_z^1(z)$ shows a linear evolution in the thickness direction. Thus the first mode of the solution predicts that vertical sections of the undeformed

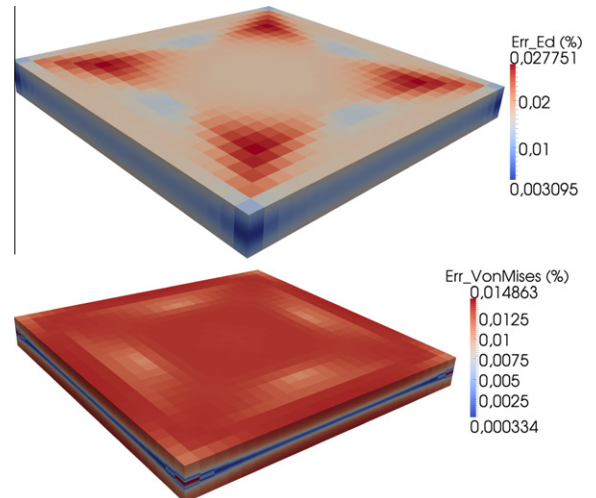


Fig. 2. Error with respect to the FEM solution considering the usual energy norm (top) and relative error in von Mises stress.

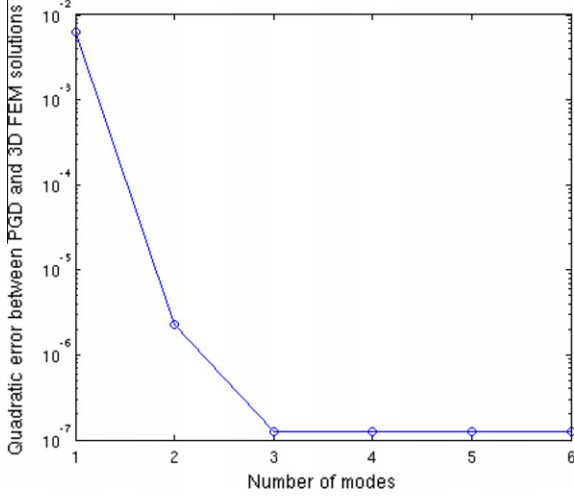


Fig. 3. Illustration of the very fast convergence of the solution with respect to the number of terms in the finite sums decomposition.

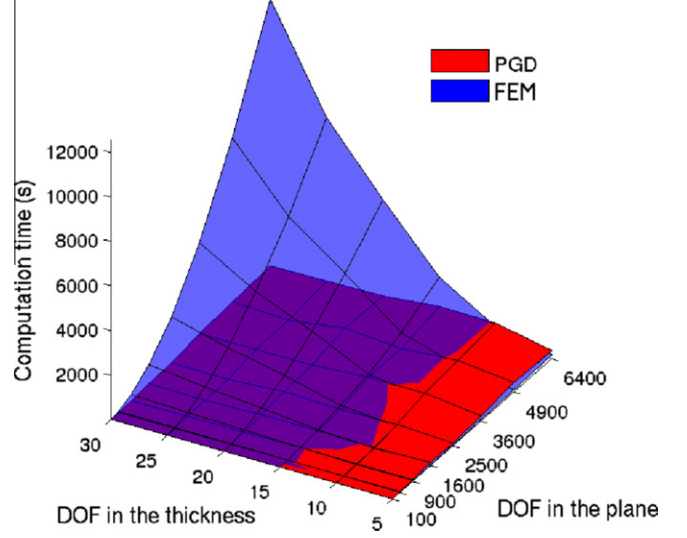


Fig. 4. Comparison of the PGD and FEM based 3D discretizations.

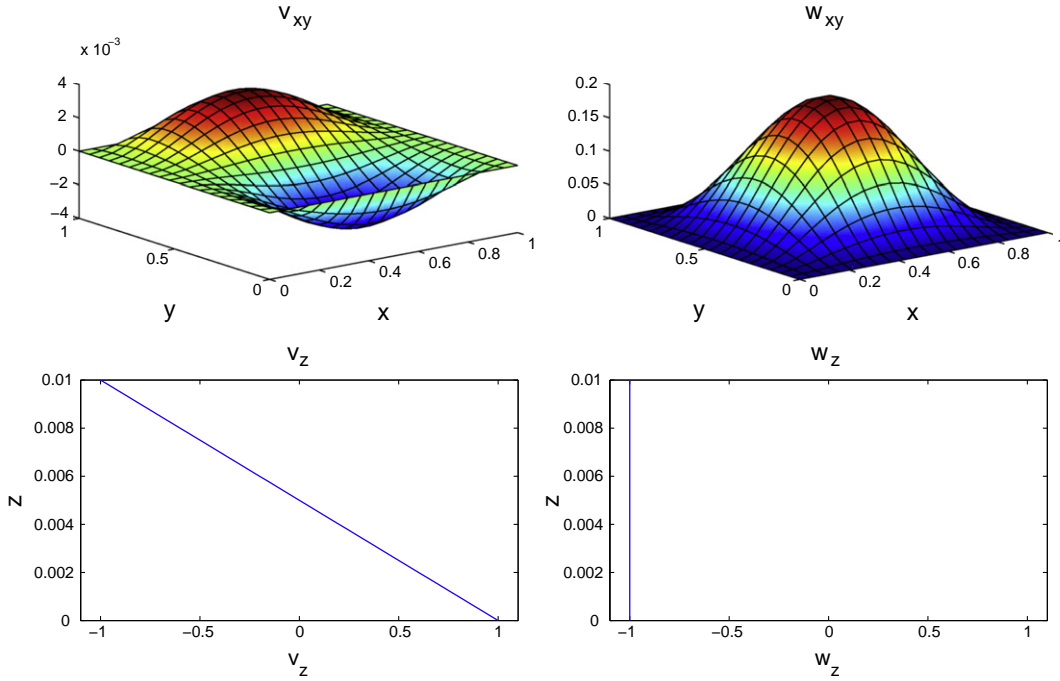


Fig. 5. First mode of the PGD solution. Left: v_{xy}^1 - top; v_z^1 - down. Right: w_{xy}^1 - top; w_z^1 - down. By symmetry, we have $u_{xy}^1 = v_{xy}^1$ and $u_z^1 = v_z^1$.

plate remain plane after deformation, and as the plate thickness decreases, they become more and more perpendicular to the plate middle surface, as expected from the Kirchoff and Reissner-Mindlin theories.

The second mode of the PGD solution shows a more complex z -dependence but it should be noticed that in the xy -plane it essentially contributes to the solution in the plate edges neighborhood where one expects to observe a truly 3D displacement field. The subsequent modes of the PGD solution gradually improve the solution quality close to the plate edges and corners.

4.2. Analysis of a laminate composite plate

In the previous example, we have implicitly assumed that the generalized Hooke tensor \mathbf{K}_{ij} was constant. By making \mathbf{K}_{ij} z -depen-

dent as proposed earlier in Eq. (5), we can represent a composite plate where each ply would have different principal directions.

We consider a square plate with a hole in its center subjected to a uniform displacement on the right plate face. As sketched in Fig. 7 one of the plate edges is kept fixed while we impose a uniform displacement \mathbf{u}_d on the opposite edge.

Along the z -dimension, the plate is made of a stacking of 16 layers of an unidirectional reinforcement, the orientation in the different layers being: $[0, 45^\circ, 90^\circ, -45^\circ]$ repeated four times. Young's modulus along (E_1) and perpendicular (E_2) to the principal direction are reported in Fig. 7.

The xy -mesh is composed of 1824 bi-linear quadrangular elements involving 5700 in plane degrees of freedom. In the z -directions, we have used 10 linear elements per ply which gives a total of 160 nodes and 483 degrees of freedom. If we refer to

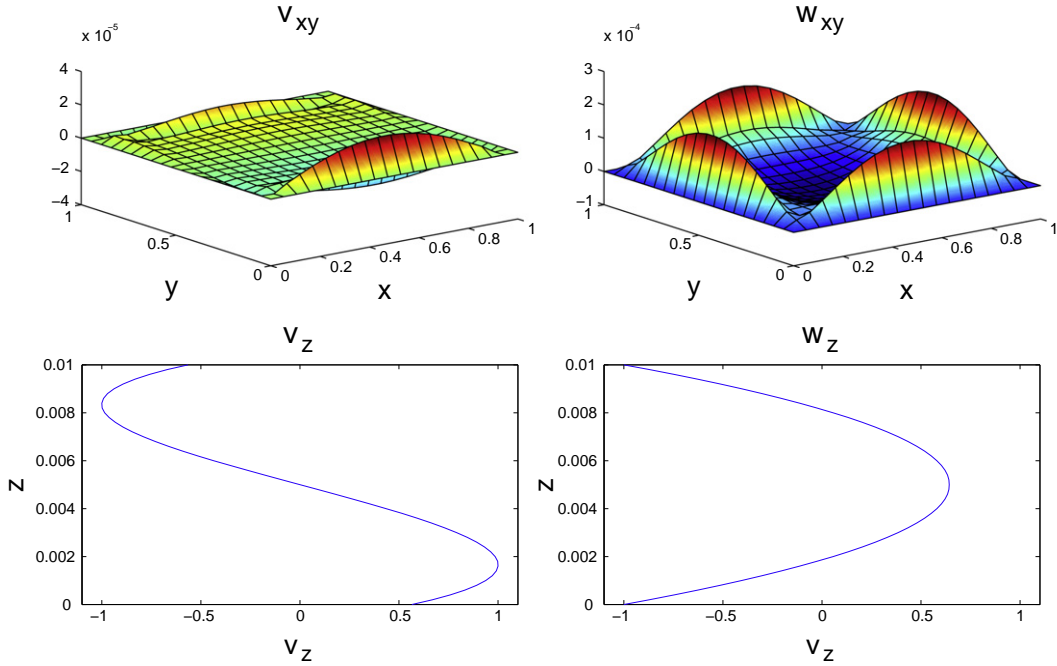


Fig. 6. Second mode of the PGD solution. Left: v_{xy}^2 – top; v_z^2 – down. Right: w_{xy}^2 – top; w_z^2 – down. By symmetry, we have $u_{xy}^2 = v_{xy}^2$ and $u_z^2 = v_z^2$.

Fig. 4 we notice that the PGD solution is expected to introduce computing time savings of many orders of magnitude.

In this particular example the separated representation only involve three modes to approximate the 3D problem solution. For computing these three modes 45 fixed point iterations were needed. The total CPU time using Matlab did not exceed 30 s on a laptop, while solving the full 3D problem would have involved almost three millions degrees of freedom and for sure a much greater computational and memory cost.

In the right part of Fig. 7, we see that the PGD method is able to fully capture the stress concentration on the rim of the hole. In Fig. 8, we show the first mode of the PGD solution.

4.3. Modeling honeycomb composites

In the previous sections we have implicitly assumed that the generalized Hooke tensor \mathbf{K} does not depend on the in-plane coordinates but this fact is not a restriction for applying the PGD based solver. An efficient implementation of the PGD only requires a separated representation of the elasticity tensor

$$K_{ij}(x, y, z) = \sum_I K_{ij}^{I,xy}(x, y) \cdot K_{ij}^{I,z}(z). \quad (33)$$

Remark 1. The PGD can be also applied even when the material behavior does not accept a separated representation. In that case it suffices to perform the integrals in the whole 3D domain. Even if the computing time is higher, the PGD solver runs.

For example by making K_{ij} z -dependent, only one mode is necessary to represent the material parameters of a laminate preform composite plate where each ply could have different principal directions of anisotropy.

In what follows we consider a quite complex structure consisting of two plates with a honeycomb core in between, as depicted in Fig. 9. It is possible to define a function $\chi(x, y, z)$, taking a unit value inside the honeycomb cells and vanishing in the cells walls as well as in the upper and lower plates, having the following separated representation:

$\chi(x, y, z) = \chi^{xy}(x, y) \cdot \chi^z(z)$. This separated representation is directly and explicitly computed from the geometry of the problem.

Assuming that the honeycomb cells are filled with a material described by the generalized Hooke tensor \mathbf{K}^1 while the cell walls and the upper and lower plates can be described with a tensor \mathbf{K}^2 , we define the generalized Hooke tensor for the whole structure as:

$$\mathbf{K}(x, y, z) = \mathbf{K}^1 \cdot \chi(x, y, z) + \mathbf{K}^2 \cdot (1 - \chi(x, y, z)). \quad (34)$$

Remark 2. If the honeycomb cells are empty it suffices to consider a null elasticity tensor.

We can therefore notice that only two modes are needed to perform a separated representation of the elasticity tensor related to such a complex geometry.

Fig. 10 depicts the solution obtained with the PGD method for a honeycomb-core plate subject to a uniform pressure on its upper face. The honeycomb structure is modeled as an isotropic material while the upper and lower plates of the structure are laminated composites with 16 plies each. Each ply is modeled as a transversely isotropic material. In this example we have used about 30,000 linear triangular elements for defining the in-plane mesh (implying 45,000 degrees of freedom) and 520 linear elements for discretizing the thickness-dependent functions. The equivalent three-dimensional finite element model implies 16 millions degrees of freedom. The solution of such a complex problem involves 225 modes to represent the solution to the desired degree of accuracy. In Fig. 10, we see that although the plate deflection is smooth, large stress concentrations in the honeycomb walls are captured by the separated representation solution. Although 225 modes might appear as too many modes, one should recall that:

- The problem is very complex and the solution quite rich. An equivalent fully 3D solution using the finite elements method would have implied 16 millions degrees of freedom.
- All those modes have been computed sequentially.
- The computed modes are the ones naturally produced by the algorithm described earlier. Neither orthogonalization nor *a posteriori* optimization has been carried out.

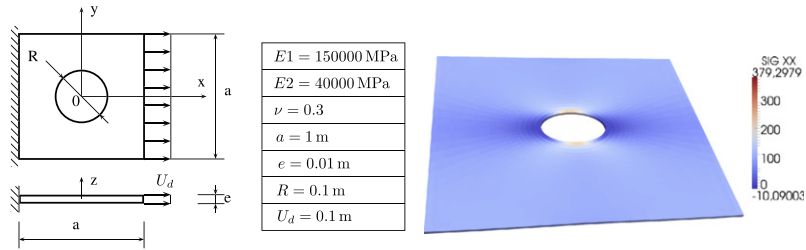


Fig. 7. Details of the problem and material parameters of the composite plate (left). Stress σ_{xx} evaluated from the PGD solution at the plate surface (right).

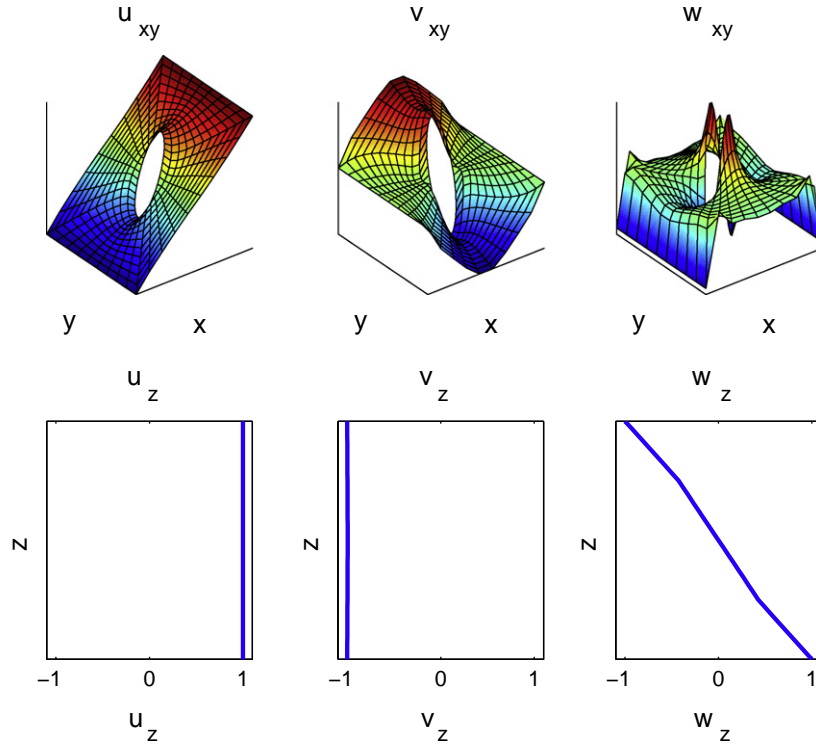


Fig. 8. Details of the first mode of the PGD solution.

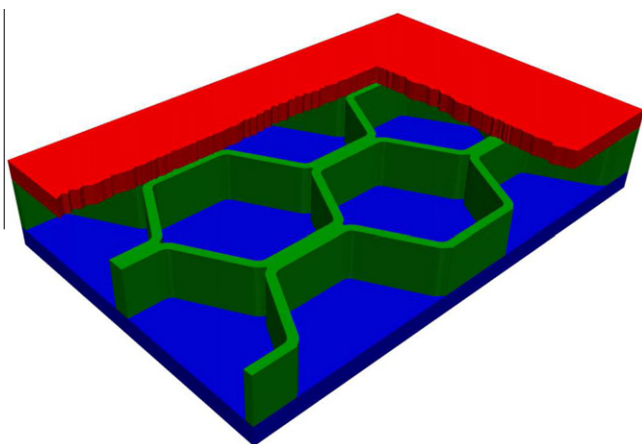


Fig. 9. Honeycomb structure between two thin plates.

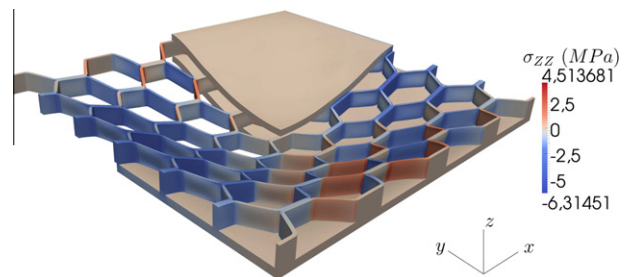


Fig. 10. Prediction of the deformed shape and σ_{zz} stress component in a complex honeycomb structure with the PGD method.

5. Parametric modeling

When using the PGD one could consider many extra-coordinates because the increase of the dimensionality of the resulting

model does not affect the possibility of computing the solution. Indeed the method was first designed as a way to circumvent the so-called “curse of dimensionality”. Thus, one could for example consider the orientation of the different plies of the composite laminate as extra-coordinates allowing through the solution of a single problem the prediction of the displacement field for any orientation of the plies. Thus, a single off-line solution could be used for the on-line particularization of the solution related to different

laminate configurations. This particularization is a simple postprocessing of the parametric solution and is computationally so inexpensive that it can be carried out on mobile computing platforms, as for example smartphones. A detailed example of the PGD algorithm with extra-coordinates can be found in the appendix.

For illustrating the capabilities of such approach, we are considering a laminate composed of four plies of a unidirectional reinforcement. For the sake of clarity we are assuming the orientations of the first and fourth plies θ_1 and θ_4 taking values in the interval $\mathcal{I} = [-20^\circ, +20^\circ]$, i.e. $\theta_1 \in \mathcal{I}$ and $\theta_4 \in \mathcal{I}$, whereas the orientation of both central plies are fixed to $\theta_2 = \theta_3 = 90^\circ$.

Now, the parametric solution is defined in a space of dimension 5, involving the three physical coordinates (x, y, z) and the two configuration coordinates θ_1 and θ_4 :

$$\mathbf{u}(x, y, z, \theta_1, \theta_4) = \begin{pmatrix} u(x, y, z, \theta_1, \theta_4) \\ v(x, y, z, \theta_1, \theta_4) \\ w(x, y, z, \theta_1, \theta_4) \end{pmatrix} \approx \sum_{i=1}^N \begin{pmatrix} u_{xy}^i(x, y) \cdot u_z^i(z) \cdot \theta_1^i(\theta_1) \cdot \theta_4^i(\theta_4) \\ v_{xy}^i(x, y) \cdot v_z^i(z) \cdot \theta_1^i(\theta_1) \cdot \theta_4^i(\theta_4) \\ w_{xy}^i(x, y) \cdot w_z^i(z) \cdot \theta_1^i(\theta_1) \cdot \theta_4^i(\theta_4) \end{pmatrix}. \quad (35)$$

The different functions involved in the separated representation of the displacement field are calculated by applying the strategy previously described for treating parametric models (Section 2).

As soon as the parametric solution (35) is available the displacement field can be obtained for any configuration $\theta_1 \in \mathcal{I}$ and $\theta_4 \in \mathcal{I}$ by simple postprocessing. Thus, we can compute the solutions envelope related to the different possible configurations.

In the case of considering a thermoelastic behavior instead of the elastic one previously considered, the heating of a non-equilibrated laminate will induce a plate deformation whose shape will depend on the values of θ_1 and θ_4 as depicted in Fig. 11. The solution envelope for all possible values of θ_1 and θ_4 is depicted in Fig. 12.

Because the on-line postprocessing involves a few amount of calculations, one could imagine to perform the solution postprocessing by using a mobile computing platform, as light as a smartphone, in order to select the orientation of both plies and then display the particularized solution.

To illustrate this capability, we consider that the parametric solution has already been computed and that it is available in a separated form. Now this solution can be introduced in a smartphone that will carry out all the on-line calculations, i.e. compute display the displacement field for the considered orientations. In our applications we considered a Nokia, with 256 MB of RAM, 16 GB of internal memory, a 680 MHz ARM 11 CPU and with Symbian 3 as operative system. The multidimensional separated solution is represented with 25 modes and requires only 460 kB of memory, while the equivalent five-dimensional grid would contain more than eight millions degrees of freedom.

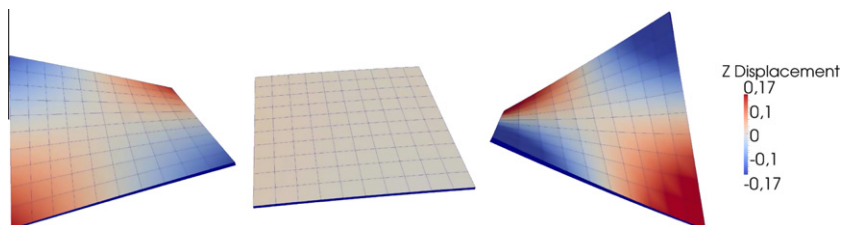


Fig. 11. Thermoelastic laminate deformed shape for different configurations $[\theta_1, \theta_2, \theta_3, \theta_4]$: (left) $[0^\circ, 90^\circ, 90^\circ, 20^\circ]$, (center) equilibrated laminate $[0^\circ, 90^\circ, 90^\circ, 0^\circ]$ and (right) $[10^\circ, 90^\circ, 90^\circ, -20^\circ]$.

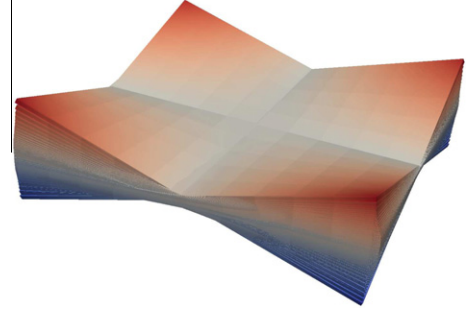


Fig. 12. Solution envelope for the full range of parameters variation.

Fig. 13 illustrates the application environment in which we can observe that the orientations of the first and fourth plies, θ_1 and θ_2 , can be selected from the two lateral sliders. Fig. 14 depicts two non-equilibrated scenarios among the numerous that the parametric solution contains implicitly.

6. Discussion

We compare here the complexity of PGD-based solvers with respect to the standard finite element method. For the sake of simplicity we will consider a hexahedral domain discretized using a regular structured grid with respectively N_x , N_y and N_z nodes in the x , y and z directions respectively. Even if the domain thickness is much lower than the other characteristic in-plane dimensions, the physics in the thickness direction could be quite rich, mainly when we consider composites plates composed of hundreds of plies in which the complex physics involved requires fully 3D descriptions. In that case thousands of nodes in the thickness direction could be required to represent accurately the solution behavior in that direction. In usual mesh-based discretization strategies this fact induces a challenging issue because the number of nodes involved in the model scales with $N_x \times N_y \times N_z$, however, if one applies a PGD based discretization, and the separated representation of the solution involves N modes, one should solve N 2D problems related to the functions involving the in-plane coordinates and N 1D problems related to the functions involving the thickness coordinate. The cost related to the solution of the one-dimensional problems can be neglected with respect to the one required for solving the two-dimensional ones. Thus, in terms of degrees of freedom, the PGD complexity scales as $N \times N_x \times N_y$. Similarly, one can estimate the computation time reduction if using e.g. a direct band solver for solving either the full 3D problem (bandwidth $\sim N_x \times N_z$) or the many 2D problems (bandwidth $\sim N_x$) arising from the PGD formulation:

$$\text{FEM computation time} \sim N_x^3 \times N_y \times N_z^3,$$

$$\text{PGD computation time} \sim N_x^3 \times N_y \times N.$$

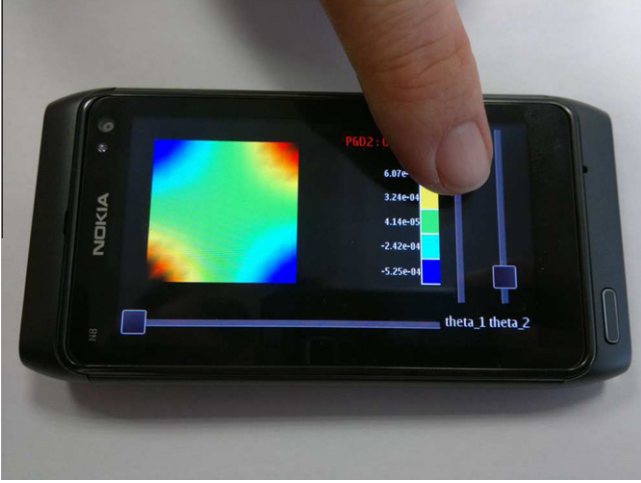


Fig. 13. Application environment for the selection of the plies orientation.

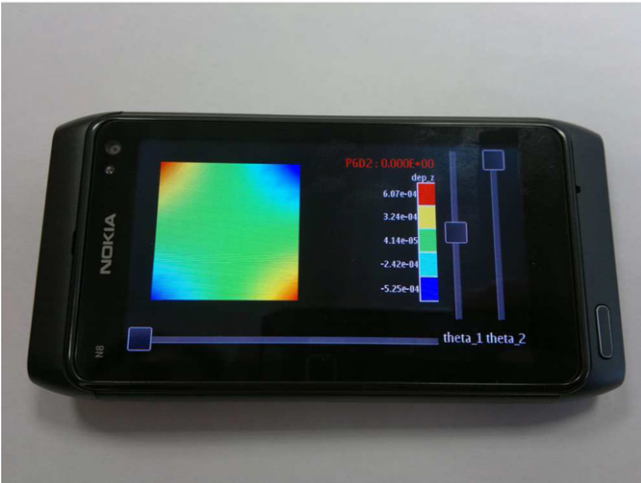
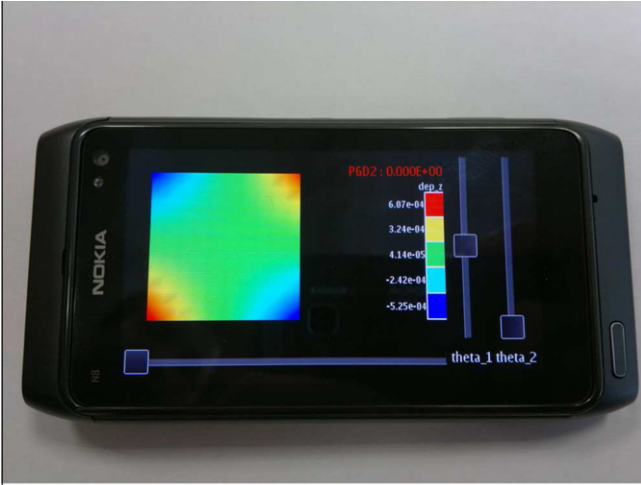


Fig. 14. Thermoelastic laminate vertical displacement for different non-equilibrated scenarios.

By comparing both complexities we can notice that as soon as $N_z \gg N$ the use of PGD-based discretization leads to impressive computing time savings, making possible even the solution of models never until now solved, even using low performance computing platforms.

7. Conclusions

In this paper we proved that a in-plane–out-of-plane separated representation can be an appealing alternative to the fully 3D solution of mechanical models defined in plate geometries. When the physics involved in the models is rich the dimensionality reduction by introducing kinematic or mechanical hypotheses seems quite delicate, and full 3D solutions seems compulsory. However, when the number of nodes in the thickness direction increases usual mesh based discretization techniques fails because of the excessive number of degrees of freedom involved. The use of a separated representation allows circumventing such drawback and solving efficiently 3D models while keeping a 2D computational complexity.

Appendix A. Revisiting the separated representation of parametric partial differential equations

In this appendix, we illustrate the PGD of parametric models on the following parametric heat conduction equation:

$$\frac{\partial u}{\partial t} - k\Delta u - f = 0. \quad (\text{A.1})$$

Here $(\mathbf{x}, t, k) \in \Omega \times I \times \mathfrak{I}$, and the source term f is assumed constant. For the sake of simplicity we assume homogeneous Dirichlet boundary conditions. The enforcement of more complex boundary conditions has been addressed previously [8]. Similar to the space and time coordinates, the conductivity k is viewed as a new coordinate defined in the interval \mathfrak{I} . Thus, instead of solving the thermal model for different discrete values of the conductivity parameter, we wish to solve at once a more general problem, the price to pay being an increase of the problem's dimensionality. As the complexity of the PGD scales only linearly with the problem dimensionality, introducing the conductivity as a new coordinate still allows the computation of an accurate and computationally inexpensive solution.

The weak form related to Eq. (A.1) reads:

$$\int_{\Omega \times I \times \mathfrak{I}} u^* \cdot \left(\frac{\partial u}{\partial t} - k\Delta u - f \right) d\mathbf{x} dt dk = 0, \quad (\text{A.2})$$

for all test functions u^* selected in an appropriate functional space.

The PGD solution is sought in the form:

$$u(\mathbf{x}, t, k) \approx \sum_{i=1}^N X_i(\mathbf{x}) \cdot T_i(t) \cdot K_i(k). \quad (\text{A.3})$$

At enrichment step n of the PGD algorithm, the following approximation is already known:

$$u^n(\mathbf{x}, t, k) = \sum_{i=1}^n X_i(\mathbf{x}) \cdot T_i(t) \cdot K_i(k). \quad (\text{A.4})$$

We wish to compute the next functional product $X_{n+1}(\mathbf{x}) \cdot T_{n+1}(t) \cdot K_{n+1}(k)$, which we write as $R(\mathbf{x}) \cdot S(t) \cdot W(k)$ for notational simplicity.

Thus, the solution at enrichment step $n + 1$ reads:

$$u^{n+1} = u^n + R(\mathbf{x}) \cdot S(t) \cdot W(k). \quad (\text{A.5})$$

Inspired from the calculus of variations, we propose the simplest choice for the test functions u^* used in Eq. (A.2):

$$u^* = R^*(\mathbf{x}) \cdot S(t) \cdot W(k) + R(\mathbf{x}) \cdot S^*(t) \cdot W(k) + R(\mathbf{x}) \cdot S(t) \cdot W^*(k). \quad (\text{A.6})$$

With the trial and test functions given by Eqs. (A.5) and (A.6) respectively, Eq. (A.2) is a non-linear problem that must be solved by means of a suitable iterative scheme. In our earlier papers [2,3], we used Newton's method. Simpler linearization strategies can also be applied, however. The simplest one is an alternating direction, fixed-point algorithm, which was found remarkably ro-

bust in the present context. Each iteration consists of three steps that are repeated until convergence, that is, until reaching the fixed point. The first step assumes $S(t)$ and $W(k)$ known from the previous iteration and computes an update for $R(\mathbf{x})$ (in this case the test function reduces to $R^*(\mathbf{x}) \cdot S(t) \cdot W(k)$). From the just-updated $R(\mathbf{x})$ and the previously-used $W(k)$, we can update $S(t)$ (with $u^* = R(\mathbf{x}) \cdot S^*(t) \cdot W(k)$). Finally, from the just-computed $R(\mathbf{x})$ and $S(t)$, we update $W(k)$ (with $u^* = R(\mathbf{x}) \cdot S(t) \cdot W^*(k)$). The converged functions $R(\mathbf{x})$, $S(t)$ and $W(k)$ yield the new functional product of the current enrichment step: $X_{n+1}(\mathbf{x}) = R(\mathbf{x})$, $T_{n+1}(t) = S(t)$ and $K_{n+1}(k) = W(k)$. The explicit form of these operations is described below.

Computing $R(\mathbf{x})$ from $S(t)$ and $W(k)$:

We consider the weak form of Eq. (A.1):

$$\int_{\Omega \times I \times \mathfrak{I}} u^* \left(\frac{\partial u}{\partial t} - k \Delta u - f \right) d\mathbf{x} dt dk = 0. \quad (\text{A.7})$$

Here, the trial function is given by:

$$u(\mathbf{x}, t, k) = \sum_{i=1}^n X_i(\mathbf{x}) \cdot T_i(t) \cdot K_i(k) + R(\mathbf{x}) \cdot S(t) \cdot W(k). \quad (\text{A.8})$$

Since S and W are known from the previous iteration, the test function reads:

$$u^*(\mathbf{x}, t, k) = R^*(\mathbf{x}) \cdot S(t) \cdot W(k). \quad (\text{A.9})$$

Introducing (A.8) and (A.9) into (A.7) yields:

$$\begin{aligned} & \int_{\Omega \times I \times \mathfrak{I}} R^* \cdot S \cdot W \cdot \left(R \cdot \frac{\partial S}{\partial t} \cdot W - k \cdot \Delta R \cdot S \cdot W \right) d\mathbf{x} dt dk \\ &= - \int_{\Omega \times I \times \mathfrak{I}} R^* \cdot S \cdot W \cdot \mathcal{R}^n d\mathbf{x} dt dk, \end{aligned} \quad (\text{A.10})$$

where \mathcal{R}^n is the residual at enrichment step n :

$$\mathcal{R}^n = \sum_{i=1}^n X_i \cdot \frac{\partial T_i}{\partial t} \cdot K_i - \sum_{i=1}^n k \cdot \Delta X_i \cdot T_i \cdot K_i - f. \quad (\text{A.11})$$

Since all functions involving time and conductivity are known, we can integrate Eq. (A.10) over $I \times \mathfrak{I}$. With the following notations,

$$\left[\begin{array}{lll} w_1 = \int_{\mathfrak{I}} W^2 dk & s_1 = \int_I S^2 dt & r_1 = \int_{\Omega} R^2 d\mathbf{x} \\ w_2 = \int_{\mathfrak{I}} k W^2 dk & s_2 = \int_I S \cdot \frac{dS}{dt} dt & r_2 = \int_{\Omega} R \cdot \Delta R d\mathbf{x} \\ w_3 = \int_{\mathfrak{I}} W dk & s_3 = \int_I S dt & r_3 = \int_{\Omega} R d\mathbf{x} \\ w_4^i = \int_{\mathfrak{I}} W \cdot K_i dk & s_4^i = \int_I S \cdot \frac{dT_i}{dt} dt & r_4^i = \int_{\Omega} R \cdot \Delta X_i d\mathbf{x} \\ w_5^i = \int_{\mathfrak{I}} kW \cdot K_i dk & s_5^i = \int_I S \cdot T_i dt & r_5^i = \int_{\Omega} R \cdot X_i d\mathbf{x} \end{array} \right], \quad (\text{A.12})$$

Eq. (A.10) reduces to:

$$\begin{aligned} & \int_{\Omega} R^* \cdot (w_1 \cdot s_2 \cdot R - w_2 \cdot s_1 \cdot \Delta R) d\mathbf{x} = \\ & - \int_{\Omega} R^* \cdot \left(\sum_{i=1}^n w_4^i \cdot s_4^i \cdot X_i - \sum_{i=1}^n w_5^i \cdot s_5^i \cdot \Delta X_i - w_3 \cdot s_3 \cdot f \right) d\mathbf{x}. \end{aligned} \quad (\text{A.13})$$

Eq. (A.13) defines in weak form an elliptic steady-state boundary value problem for the unknown function R that can be solved using any suitable discretization technique, e.g. finite elements after the integration by parts of Eq. (A.13). Another possibility consists in coming back to the strong form of Eq. (A.13):

$$\begin{aligned} & w_1 \cdot s_2 \cdot R - w_2 \cdot s_1 \cdot \Delta R = \\ & - \sum_{i=1}^n w_4^i \cdot s_4^i \cdot X_i - \sum_{i=1}^n w_5^i \cdot s_5^i \cdot \Delta X_i - w_3 \cdot s_3 \cdot f, \end{aligned} \quad (\text{A.14})$$

that can be solved using any classical collocation technique (finite differences, SPH, etc.).

Computing $S(t)$ from $R(\mathbf{x})$ and $W(k)$:

In the present step, the test function writes:

$$u^*(\mathbf{x}, t, k) = S^*(t) \cdot R(\mathbf{x}) \cdot W(k), \quad (\text{A.15})$$

and the weak form becomes:

$$\begin{aligned} & \int_{\Omega \times I \times \mathfrak{I}} S^* \cdot R \cdot W \cdot \left(R \cdot \frac{\partial S}{\partial t} \cdot W - k \cdot \Delta R \cdot S \cdot W \right) d\mathbf{x} dt dk \\ &= - \int_{\Omega \times I \times \mathfrak{I}} S^* \cdot R \cdot W \cdot \mathcal{R}^n d\mathbf{x} dt dk. \end{aligned} \quad (\text{A.16})$$

Integrating over $\Omega \times \mathfrak{I}$, one obtains:

$$\begin{aligned} & \int_I S^* \cdot \left(w_1 \cdot r_1 \cdot \frac{dS}{dt} - w_2 \cdot r_2 \cdot S \right) dt = \\ & - \int_I S^* \cdot \left(\sum_{i=1}^n w_4^i \cdot r_5^i \cdot \frac{dT_i}{dt} - \sum_{i=1}^n w_5^i \cdot r_4^i \cdot T_i - w_3 \cdot r_3 \cdot f \right) dt. \end{aligned} \quad (\text{A.17})$$

Eq. (A.17) represents the weak form of the scalar ODE defining the time evolution of the field S that can be solved using any stabilized discretization technique (SU, Discontinuous Galerkin, etc.). The strong form of Eq. (A.17) reads

$$w_1 \cdot r_1 \cdot \frac{dS}{dt} - w_2 \cdot r_2 \cdot S = - \left(\sum_{i=1}^n w_4^i \cdot r_5^i \cdot \frac{dT_i}{dt} - \sum_{i=1}^n w_5^i \cdot r_4^i \cdot T_i - w_3 \cdot r_3 \cdot f \right). \quad (\text{A.18})$$

Eq. (A.18) can be solved using backward finite differences, or higher order Runge–Kutta schemes, among many other possibilities.

Computing $W(k)$ from $R(\mathbf{x})$ and $S(t)$:

The test function is now given by:

$$u^*(\mathbf{x}, t, k) = W^*(k) \cdot R(\mathbf{x}) \cdot S(t), \quad (\text{A.19})$$

and the weak form becomes:

$$\begin{aligned} & \int_{\Omega \times I \times \mathfrak{I}} W^* \cdot R \cdot S \cdot \left(R \cdot \frac{\partial S}{\partial t} \cdot W - k \cdot \Delta R \cdot S \cdot W \right) d\mathbf{x} dt dk \\ &= - \int_{\Omega \times I \times \mathfrak{I}} W^* \cdot R \cdot S \cdot \mathcal{R}^n d\mathbf{x} dt dk. \end{aligned} \quad (\text{A.20})$$

Integration over $\Omega \times I$ yields:

$$\begin{aligned} & \int_{\mathfrak{I}} W^* \cdot (r_1 \cdot s_2 \cdot W - r_2 \cdot s_1 \cdot k \cdot W) dk = \\ & - \int_{\mathfrak{I}} W^* \cdot \left(\sum_{i=1}^n r_5^i \cdot s_4^i \cdot K_i - \sum_{i=1}^n r_4^i \cdot s_5^i \cdot k \cdot K_i - r_3 \cdot s_3 \cdot f \right) dk. \end{aligned} \quad (\text{A.21})$$

Eq. (A.21) does not involve any differential operator. The corresponding strong form reads:

$$(r_1 \cdot s_2 - r_2 \cdot s_1 \cdot k) \cdot W = - \sum_{i=1}^n (r_5^i \cdot s_4^i - r_4^i \cdot s_5^i \cdot k) \cdot K_i - r_3 \cdot s_3 \cdot f. \quad (\text{A.22})$$

Eq. (A.22) defines an algebraic problem, which is hardly a surprise since the original equation (A.1) does not contain derivatives with respect to the parameter k . The introduction of the parameter k as additional model coordinate does not increase the cost of a particular enrichment step. It does however necessitate more enrichment steps, i.e. more terms (higher N) in the decomposition (A.3).

At each enrichment step the construction of the new functional product in Eq. (A.3) requires iterations. If m_i denotes the number of iterations needed at enrichment step i , the total number of iterations involved in the construction of the PGD approximation is $m = \sum_{i=1}^N m_i$. In the above example, the entire procedure thus involves the solution of m three-dimensional problems for the functions $X_i(\mathbf{x})$, m one-dimensional problems for the functions $T_i(t)$ and

m algebraic systems for the functions $K_i(k)$. m_i rarely exceeds 10. The number N of functional products needed to approximate the solution with enough accuracy ranges between a few tens and one hundred, depending on the complexity and dimensionality of the problem. Thus, we can conclude that the complexity of the PGD procedure to compute the approximation (A.3) is of some tens of 3D steady-state problems (the cost related to the 1D and algebraic problems being negligible with respect to the 3D problems). In a classical approach, one must solve for each particular value of the parameter k a 3D problem at each time step. In usual applications, this often implies the computation of several millions of 3D solutions. Clearly, the CPU time savings by applying the PGD can be of several orders of magnitude.

References

- [1] Y. Abou Msallem, F. Jacquemin, A. Poitou, Residual stresses formation during the manufacturing process of epoxy matrix composites: resin yield stress and anisotropic chemical shrinkage, *Int. J. Mater. Forum* 3 (22) (2010) 1363–1372.
- [2] A. Ammar, B. Mokdad, F. Chinesta, R. Keunings, A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids, *J. Non-Newtonian Fluid Mech.* 139 (2006) 153–176.
- [3] A. Ammar, B. Mokdad, F. Chinesta, R. Keunings, 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, *J. Non-Newtonian Fluid Mech.* 144 (2007) 98–121.
- [4] F. Chinesta, A. Ammar, A. Falco, M. Laso, On the reduction of stochastic kinetic theory models of complex fluids, *Model. Simul. Mater. Sci. Engrg.* 15 (2007) 639–652.
- [5] F. Chinesta, A. Ammar, P. Joyot, The nanometric and micrometric scales of the structure and mechanics of materials revisited: an introduction to the challenges of fully deterministic numerical descriptions, *Int. J. Multiscale Comput. Engrg.* 6 (3) (2008) 191–213.
- [6] F. Chinesta, A. Ammar, E. Cueto, On the use of proper generalized decompositions for solving the multidimensional chemical master equation, *Eur. J. Comput. Mech.* 19 (2010) 53–64.
- [7] F. Chinesta, A. Ammar, E. Cueto, Recent advances in the use of the Proper Generalized Decomposition for solving multidimensional models, *Arch. Comput. Methods Engrg.* 17/4 (2010) 327–350.
- [8] D. Gonzalez, A. Ammar, F. Chinesta, E. Cueto, Recent advances in the use of separated representations, *Int. J. Numer. Methods Engrg.* 81/5 (2010) 637–659.
- [9] Ch. Hochard, P. Ladeveze, L. Proslie, A simplified analysis of elastic structures, *Eur. J. Mech. A/Solids* 12 (4) (1993) 509–535.
- [10] P. Ladeveze, *Nonlinear Computational Structural Mechanics*, Springer, NY, 1999.
- [11] P. Ladeveze, L. Arnaud, P. Rouch, C. Blanze, The variational theory of complex rays for the calculation of medium-frequency vibrations, *Engrg. Comput.* 18 (1–2) (2001) 193–214.
- [12] P. Ladeveze, A. Nouy, On a multiscale computational strategy with time and space homogenization for structural mechanics, *Comput. Methods Appl. Mech. Engrg.* 192 (2003) 3061–3087.
- [13] P. Ladeveze, J.-C. Passieux, D. Neron, The LATIN multiscale computational method and the proper generalized decomposition, *Comput. Methods Appl. Mech. Engrg.* 199 (21–22) (2010) 1287–1296.
- [14] B. Mokdad, E. Pruliere, A. Ammar, F. Chinesta, On the simulation of kinetic theory models of complex fluids using the Fokker–Planck approach, *Appl. Rheol.* 17/2 (26494) (2007) 1–14.
- [15] B. Mokdad, A. Ammar, M. Normandin, F. Chinesta, J.R. Clermont, A fully deterministic micro–macro simulation of complex flows involving reversible network fluid models, *Math. Comput. Simul.* 80 (2010) 1936–1961.
- [16] A. Nouy, A generalized spectral decomposition technique to solve a class of linear stochastic partial differential equations, *Comput. Methods Appl. Mech. Engrg.* 196 (2007) 4521–4537.
- [17] A. Nouy, Generalized spectral decomposition method for solving stochastic finite element equations: invariant subspace problem and dedicated algorithms, *Comput. Methods Appl. Mech. Engrg.* 197 (2008) 4718–4736.
- [18] E. Pruliere, A. Ammar, N. El Kissi, F. Chinesta, Recirculating flows involving short fiber suspensions: numerical difficulties and efficient advanced micro–macro solvers, *Arch. Comput. Methods Engrg., State Art Rev.* 16 (2009) 1–30.
- [19] S. Timoshenko, S. Woinowsky-Krieger, *Theory of Plates and Shells*, second ed., McGraw-Hill, 1959.