RESEARCH

Effect of the separated approximation of input data in the accuracy of the resulting PGD solution

Sergio Zlotnik[1*], Pedro Díez[1], Elías Cueto[2], David Gonzalez[2] and Antonio Huerta[1]

[*]Correspondence:
sergio.zlotnik@upc.edu
[1]LaCàN, UPC, Jordi Girona 1,
08034 Barcelona, Spain
Full list of author information is
available at the end of the article

Abstract

The Proper Generalized Decomposition (PGD) requires separability of the input data (e.g. physical properties, source term, boundary conditions, initial state). In many cases the input data is not expressed in a separated form and it has to be replaced by some separable approximation. These approximations constitute a new error source that, in some cases, may dominate the standard ones (discretization, truncation...) and control the final accuracy of the PGD solution.

In this work the relation between errors in the separated input data and the errors induced in the PGD solution is discussed. Error estimators proposed for homogenized problems and oscillation terms are adapted to asses the behaviour of the PGD errors resulting from approximated input data. The PGD is stable with respect to error in the separated data, with no critical amplification of the perturbations.

Interestingly, we identified a high sensitiveness of the resulting accuracy on the selection of the sampling grid used to compute the separated data. The separation has to be performed on the basis of values sampled at integration points: sampling at the nodes defining the functional interpolation results in an important loss of accuracy.

For the case of a Poisson problem separated in the spacial coordinates (a complex diffusivity function requires a separable approximation), the final PGD error is linear with the truncation error of the separated data. This relation is used to estimate the number of terms required in the separated data, that has to be in good agreement with the truncation error accepted in the PGD truncation (tolerance for the stoping criteria in the enrichment procedure). A sensible choice for the prescribed accuracy of the PGD solution has to be kept within the limits set by the errors in the separated input data.

Keywords: Proper Generalized Decomposition; Error assessment; Separable functions

1 Background

The Proper Generalized Decomposition (PGD) [1, 2] is an *a priori* reduced basis technique designed to deal efficiently with highly-dimensional Boundary Value Problems (BVP). Differently from other discretisation techniques such as Finite Elements of Finite Differences, PGD avoids the exponential growth of the number of degrees of freedom with the number of dimensions. This is achieved by using of a separated representation of the solution. A *separable function* f with rank q,

separated on n dimensions has the form,

$$f(x_1, x_2, \ldots, x_n) = \sum_{m=1}^{q} F_{x_1}^m(x_1) F_{x_2}^m(x_2) \ldots F_{x_n}^m(x_n) = \sum_{m=1}^{q} \prod_{p=1}^{n} F_{x_p}^m(x_p). \quad (1)$$

In addition to the separability of the solution, PGD requires the separability of input data that depends on several separated dimensions, for example physical parameters varying on space or time. Theoretically, separability of input data is not necessary; although, in practice it is required to avoid an exponential growth of the number of operations in obtaining the PGD solution.

Input data is in general not separable and therefore it is in practice replaced by separable approximations of it. For example, in following sections a Poisson problem is solved using PGD. The solution u is separated into two spatial dimensions as $u(x, y) = \sum_{m=1}^{q} F_x^m(x) F_y^m(y)$. A diffusivity function $k(x, y)$ varying in space is introduced:

$$k(x, y) = \sin\left(0.5(x + y)^2\right) + 1.5 \quad (2)$$

This function does not admit an exact separated representation and therefore it is replaced by an approximation

$$k(x, y) \approx k^{\text{sep}}(x, y) = \sum_{l=1}^{n_k} G_x^l(x) G_y^l(y), \quad (3)$$

having n_k terms.

The PGD solver, therefore, introduces new errors into the solution that are not present in a standard Finite Element (FE) solution: first, a truncation error due to the finite number of terms (n_k) used to describe k^{sep} and, second, a discretization error in the spatial representation of the functions $G_x^l(x)$ and $G_y^l(y)$. The goal of this work is to understand how these errors affect the accuracy of the PGD solution. The relation between the errors in k^{sep} and the error of the PGD solution can be used in practice to determine the n_k required to achieve a certain accuracy by PGD. Moreover, this relation can also be used as stopping criteria for the PGD enrichment process, as the PGD solution will not be able to achieve an accuracy under that imposed by k^{sep}.

1.1 Motivation examples

The errors introduced when imposing separability to input data ultimately control the maximum accuracy that a PGD solution may achieve. The evolution of the error with the number of terms of the solution (e.g. Figure 5) show clearly a flattening of the curves and a limit in the accuracy the PGD could obtain for a certain accuracy of the separated data. This behaviour was observed in some practical examples briefly presented next.

A first example is the solution of a BVP whose domain depends on a set of parameters, $\boldsymbol{\mu}$. These control the geometry of the domain or the location of internal interfaces. An example is shown in Figure 5 where two parameters control the shape

of an airfoil and a flow problem is solved far any geometry within the parameter range. The methodology to solve the geometrically parameterized problems is proposed in [3] and later extended in [4]. It is based on expressing the Jacobians of the elements, $\mathbf{J}_e(\boldsymbol{\mu})$, as a function of the parameters. For example, the weak form of the bilinear operator arising from a Poisson problem reads,

$$a(u,v) = \int_{\Omega^e(\boldsymbol{\mu})} \nabla u \cdot (k \nabla v)\, d\Omega = \int_{\mathcal{T}} \nabla_{\hat{\boldsymbol{x}}} u_e \cdot \underbrace{(k\, |\mathbf{J}_e(\boldsymbol{\mu})| \mathbf{J}_e(\boldsymbol{\mu})^{-\mathsf{T}} \mathbf{J}_e(\boldsymbol{\mu})^{-1}}_{\mathbf{D}_e(\boldsymbol{\mu})} \nabla_{\hat{\boldsymbol{x}}} v_e)\, d\hat{\boldsymbol{x}}$$

where $\hat{\boldsymbol{x}}$ are some reference coordinates and the matrix $\mathbf{D}_e(\boldsymbol{\mu})$ accounts for the geometrical parameterization. The analytical expression of the coefficients of $\mathbf{D}_e(\boldsymbol{\mu})$ is known but, in a general case, it is not separable. Therefore, a separable approximation of its coefficients is utilized and, consequently, a truncation error is introduced.

A second example presented in [5] is a real-time integration scheme for the equations of solid dynamics. It is based on a combination of POD–PGD approaches. The method is based upon the formulation of solid dynamics equations as a parametric problem depending on their initial conditions (that need to be separated). This parametric problem, depending on the initial conditions in the interval $(t, t+\Delta t)$, is solved off–line, for any value of the parameters within a prescribed interval. The separation of the initial conditions is done as a pre–process introducing again truncated input data and therefore generating an error on the solution.

1.2 A priori estimates for FE

Different sources of errors are present in the solution provided by PGD (see for example [6, 7, 8]. If u is the analytical solution of the BVP and $u_{H,M}$ is the solution of PGD characterized by a mesh size H and a number of terms M, the PGD error is then defined by $e := u - u_{H,M}$. This error can be divided into several sources: first, an interpolation error, $e_{FE} = u - u_H$, related with the space discretization, where u_H is the standard FE solution of the problem. Second, a truncation error $e_M := u_H - u_{H,M}$ that comes from the finite number of terms computed by PGD. The PGD error e, therefore, can be written as

$$e = u - u_{H,M} = \underbrace{u - u_H}_{e_{FE}} + \underbrace{u_H - u_{H,M}}_{e_M} \tag{4}$$

where the contribution of each type of error becomes explicit. Figure 5 shows schematically the relation between these errors. When the input data separation is required and functions are replaced by separable approximations, another source of errors gets into the equation. The replacement of function k by k^{sep} is assumed to affect similarly to the FE solution and the PGD solution (i.e. the truncation error is assumed to be independent of the error introduced by FE). If the error affects the source term, error estimators proposed for data oscillation could be used, for example [9].

If the diffusivity function k is the one separated, the ideas of homogenization theory (e.g. [10, 11]) can be recalled: k can be understood as $k = k^{\text{sep}} + \varepsilon$, being ε a highly oscillatory function with small amplitude compared to k (as shown in Figure

5). Note that ε can be reduced by increasing the number of terms n_k in k^{sep}. The problem, although, is inverse to the standard homogenization problem: the exact solution here is smooth and the high frequency terms are the errors introduced by the separation.

The standard error estimates for FE read

$$e_H = ||u - u_H|| \leq CH^\alpha,$$

for some value of alpha depending on the norm chosen. When oscillation terms are included an extra term appears:

$$e_H^{\text{sep}} = ||u - u_H^{\text{sep}}|| \leq CH^\alpha + \text{Osc},$$

being $\text{Osc} = ||k - k^{\text{sep}}||$. The truncation error, e_M, introduced by PGD is a function decreasing with the number of terms M, so its norm is bounded by $||e_M|| \leq \tilde{C}F(M)$. Note that, as mentioned above, for error affecting the source term s the standard estimates for oscillation terms provide a similar expression for Osc [9].

The final error of the PGD solution, therefore can be stated as

$$||u - u_{H,M}^{\text{sep}}|| \leq CH^\alpha + \tilde{C}F(M) + \text{Osc}. \tag{5}$$

This bound expression shows that if Osc dominates over the truncation error, the error of the PGD solution cannot be reduced. On the other hand, if an estimation for Osc and for e_H^{sep} at enrichment step i are available, (5) can be used as stopping criteria of the enrichment process.

2 Problem statement and PGD solution for separated space dimensions

In order to study the propagation of the errors within the PGD scheme a boundary value problem governed by a Poisson equation is considered. Its solution u, taking values in Ω, satisfies,

$$-\nabla \cdot (k\nabla u) = s \qquad \text{in } \Omega \tag{6a}$$

$$(k\nabla u) \cdot \boldsymbol{n} = g_N \qquad \text{on } \Gamma_N \tag{6b}$$

$$u = u_D \qquad \text{on } \Gamma_D \tag{6c}$$

where the source term s, the prescribed values on the Dirichlet boundary u_D, the prescribed flux on the Neumann value g_N and the diffusivity k are the data set. The usual variational form for this problem reads: find $u \in V$ such that

$$a\,(u, v) = \ell(v), \quad \text{for all } v \in V_0, \tag{7}$$

where $V := \{u \in \mathcal{H}^1(\Omega) : u = u_D \text{ in } \Gamma_D\}$ and its corresponding test functions space is $V_0 := \{u \in \mathcal{H}^1(\Omega) : u = 0 \text{ on } \Gamma_D\}$. The bilinear and linear forms $a(\cdot, \cdot)$ and $\ell(\cdot)$ are given by

$$a\,(u, v) := \int_\Omega \nabla u \cdot (k\nabla v)\, d\Omega \quad \text{and} \quad \ell(v) := \int_\Omega sv\, d\Omega + \int_{\Gamma_N} g_N v\, ds. \tag{8}$$

2.1 Space–separated PGD algorithm

The space–separated PGD algorithm for problem (7) is based on a separated solution $u^{\text{sep}}(x,y)$ with the form

$$u \approx u^{\text{sep}}(x,y) = \sum_{m=1}^{n_u} F_x^m(x) F_y^m(y). \tag{9}$$

As usual in FE, u^{sep} is inserted in the weak form (7). In this case, the diffusivity function $k(x,y)$ is also replaced by its separable approximation k^{sep}, and therefore the operator $a(\cdot,\cdot)$ is somehow redefined. The problem then reads: find u^{sep} such that

$$a^{\text{sep}}\left(u^{\text{sep}}, v\right) = \ell(v), \text{ for all } v, \tag{10}$$

where

$$a^{\text{sep}}\left(u, v\right) := \int_\Omega \nabla u \cdot \left(k^{\text{sep}} \nabla v\right) d\Omega \tag{11}$$

and the definition of $\ell(\cdot)$ remains unchanged.

Assuming that the initial n_u terms of u^{sep} are known (that is all functions F_x^m and F_y^m, for $m = 1 \dots n_u$ are known), the addition of a new term $F_x F_y$ to u^{sep} is done by solving the following problem:

$$a^{\text{sep}}\left(F_x F_y, v\right) = \ell(v) - \sum_{m=1}^{n_u} a^{\text{sep}}\left(F_x^m F_y^m, v\right). \tag{12}$$

To simplify notation, the dependence of each function F_*^m is kept implicit in the subindex, for example F_x^i stands for $F_x^i(x)$.

The PGD solution is constructed one term at a time using the incremental procedure suggested in (12). The addition of a new term involves solving problem (12) with all the previously computed terms in their right hand side. Note that this problem is non linear because the unknown F_x multiplyes F_y. This non linearity is usually handled by an alternate–directions algorithm consisting in first solving for F_x, assuming F_y is known, and then solving for F_y, assuming F_x is known. These two (linear) subproblems are iterated until convergence.

The test functions v belong to V_0 and they are written as $v = \delta F_x F_y + F_x \delta F_y$. When solving the first problem, F_y is assumed to be fix and therefore δF_y vanishes. The test function v, then, is simplified to $v = \delta F_x F_y$ and the first subproblem is stated as,

$$a^{\text{sep}}\left(F_x F_y, \delta F_x F_y\right) = \ell(\delta F_x F_y) - \sum_{m=1}^{n_u} a^{\text{sep}}\left(F_x^m F_y^m, \delta F_x F_y\right). \tag{13}$$

The second problem is exactly symmetric inverting the dimensions x and y.

3 Separation of the input data

Several procedures can be applied to obtain separable approximations of known functions. The Proper Orthogonal Decomposition (POD) and the Singular Value Decomposition (SVD) are the most common techniques when the separation is done in two dimensions. Many techniques have been proposed to extend SVD to higher number of dimensions. These techniques are usually called *higher–order*, as they were originally proposed to decompose higher–order tensors. An overview can be found, for example, at [12]. Some examples are the *Higher–Order SVD* (HOSVD) [13], the CANDECOMP/PARAFAC (CP) [14, 15] and the Tucker decomposition [16].

When the number of separated dimensions is two, the POD and the SVD are equivalent and they provide a optimal decomposition in the sense that they provide the minimum number of required to obtain an given accuracy. Unfortunately, for $n > 2$ this property is lost and usually there is no guarantee of the optimality of the separated tensor.

Recently Modesto et al. [17] proposed a method based on PGD to perform separation of functions. Their approach has the advantages of being equivalent to SVD when the separation is done in two-dimensions and it is trivial to extend it to higher dimensions. Their technique produced decompositions having lower rank than HOSVD for all tested cases and it does not require to specify the order of the separated function before starting the process (as CP does).

The application of SVD to obtain a two–dimensional separable approximation is explained next: consider a discrete approximation of a function $f(x, y)$ supported on a Finite Element (FE) mesh, that is, f is determined by a set of nodal values f_i for $i = 1$ to n_t. In this case the dimensions in which the function will be separated are the cartesian axis x and y. The form of the approximation is,

$$f(x, y) \approx \sum_{m=1}^{q} \alpha^m F^m(x) G^m(y),$$

where the set of function $F^m(x)$ and $G^m(y)$ are to be determined. A scalar α^m holding the amplitude of each term is added in order to normalize F^m and G^m. These functions are also supported in a FE mesh with the corresponding dimensionality; in this example both are 1D meshes.

Let $\mathbf{M}$ be a $m \times n$ matrix with rank r and coefficients $f(x_i, y_i)$, where x_i and y_i are the nodal locations. The SVD provides a factorisation $\mathbf{M}$ in the form

$$\mathbf{M} = \mathbf{U} \cdot \mathbf{S} \cdot \mathbf{V}^T \tag{14}$$

where the columns of $\mathbf{U} \in \mathcal{R}^{m \times m}$ are called the left-singular-vectors and denoted here as $\mathbf{U}_i$. The columns of $\mathbf{V} \in \mathcal{R}^{n \times n}$ are called the right-singular-vectors and denoted $\mathbf{V}_j$. The matrix $\mathbf{S} \in \mathcal{R}^{m \times n}$ is rectangular and diagonal and holds the singular values of $\mathbf{M}$ sorted from larger ($\mathbf{S}_{11}$) to smaller. Matrices $\mathbf{U}$ and $\mathbf{V}$ are both unitary, in the sense that their transpose its equal to their right inverse.

The factorisation provided be SVD allows to construct a separated representation of the matrix $\mathbf{M}$ as,

$$\mathbf{M} = \sum_{i=1}^{r} \mathbf{S}_{ii} \cdot \mathbf{U}_i \cdot \mathbf{V}_i^T. \tag{15}$$

In practice, the rank of the separated tensor is kept as low as possible as the computational effort is usually proportional to the number of terms in it. Therefore, it is usual to truncate the sum and discard all terms with amplitud smaller than a given threshold. That is, the terms corresponding to the largest eigenvalues are kept and terms with smaller eigenvalues are discarded.

As an example the function $k(x,y) = \sin\left(\frac{1}{2}(x+y)^2\right) + 2$ introduced in (2) is separated using SVD to obtain $k^{\text{sep}}(x,y)$ as defined in (3). This function is chosen because it does not admits an exact separated representation. Figure 5 shows the function $k(x,y)$ (top right), the amplitude of the initial terms in the separated version of k, that is, the diagonal coefficients of the matrix $\mathbf{S}$ (top left), and the functions F^m and G^m for the four initial terms of $k^{\text{sep}}(x,y)$. Note that with the initial 25 terms the function k is approximated to machine precision. The meshes corresponding to F and G, both have 402 nodes.

3.1 Influence of the sampling points

The first idea is sampling the input date (material parameters, source terms...) on the nodes of the grid used for the space and parametric discretization. As it is shown in the next section, this choice is not particularly sensible because the values of these functions are required at the integration points of the FE mesh used to solve the weak form of the equation. This extends not only to the spacial coordinates but also to the parametric coordinates because the parametric modes are approximated in a least squares sense (Galerkin L_2 projection). Thus, separation has to be performed on the basis of values sampled at integration points: sampling at the nodes defining the functional interpolation results in an important loss of accuracy.

4 Results

The behaviour of the PGD scheme with respect to errors in the input data is studied next via a series of numerical experiments. The problem (6) is solved using PGD as described above in a unit square domain.It is closed with Dirichlet boundary conditions on the top and bottom sides with values one and zero respectively and homogeneous Neumann in the lateral sides. The separated diffusivity function 3 is used. The mesh is structured and regular and has 100 elements in each dimension.

The relative errors shown in convergence curves is computed as the H1 norm of the relative difference between the PGD solution and a reference Finite Element solution computed over the same mesh. Note that the FE solution is computed using the exact analytic expression for the diffusivity k.

4.1 Input data sampling

The diffusivity function (2) is separated using the SVD approach described in Section 3. To do that, the spatial grid to sample the function $k(x,y)$ needs to be

selected. The first choice taken here is to evaluate k in the same mesh that will be later used in the discretization of u. In this case, it is a regular grid with 101×101 nodes. This is an overkill mesh to represent the function k (see first panel of Figure 5). The k^{sep} separated function described with 26 terms has an maximum nodal relative error of the order of machine tolerance (10^{-14}).

When k^{sep} is introduced into the weak form and the problem is solved via PGD, the solution obtained is rather inaccurate having relative errors of order 10^{-2} (Figure 5). This poor behaviour comes from the fact that the diffusivity function was sampled at the nodes but it is required by PGD (and by FE) at the integration points. The values of k^{sep} used in the integrals are interpolated spatially and therefore an "H-like" error is introduced. This error is not related with the truncation on the number of terms used in k^{sep}, but is only dependent on the grid chosen to sample k.

In the example above, despite the nodal values of k^{sep} have errors that could be negligible, the interpolated values at the mid points of the elements have relative error of order 10^{-2}, coinciding with the maximum accuracy that PGD could provide.

To overcome this limit the grid used to sample k^{sep} is modified so that the grid nodes coincide with the integration points that will be used later by the integrals of PGD. Figure 5 shows an example of such a mesh for a quadrature of 4 points per element in each direction x and y. Same as in the previous grid, the nodal values of k^{sep} have errors comparable of machine tolerance but, in this case, the spatial interpolation is completely avoided. When this new k^{sep} is used, the limit imposed by the interpolation disappears and PGD recovers it normal convergence.

4.2 Accuracy of k^{sep}

A second set of tests is done to evaluate the relation between the accuracy of PGD and the truncation error of k^{sep}. To do that, the problem is solved several times using different truncated versions of k^{sep} for $n_k = 5, 6, 7, \ldots, 11, 12, 14$. Note that all n_k are smaller than 26 (26 terms were required to get machine tolerance at the nodes) and therefore we do not expect the errors to vanish at the nodes. The grid for k^{sep} is taken coinciding with the integration points. Figure 5 shows the different convergence curves of the error on the PGD solution as a function of the number of terms. Recall that the errors are computed against the FE solution having the exact k function. All curves present a final flattening and a convergence to an error that is imposed by truncation error of k^{sep}. On other words, at some point, the error Osc (that does not depends on the number of terms) dominates in (5) and therefore the PGD error cannot decrease. The better the description of k^{sep} (that is, the larger n_k and the smaller the Osc term), the smaller the final error achieved by PGD. For this example and when Osc dominates, the relation between the PGD error and k^{sep} truncation error is linear with slope equal one (as shown in Figure 5).

5 Conclusions

The stability of PGD with respect to errors in the input data was studied by means of numerical experiments. These errors are in practice present due to the need of approximate input data by truncated separable expressions. Moreover, separation requires discretization introducing into the input data a "spatial?? h–like error.

Results show that PGD is stable (it does not amplify errors). In the tested case of a BVP governed by the Poisson equation, the errors introduced on the diffusivity function are linear with the final error that PGD commits. The grid in which the separated data is represented is crucial to the accuracy of PGD; to minimize interpolation error, the mesh for the input data should coincide with the integration points used for the solution of u.

The relation between the errors in the input data and the final error of PGD can be used to decide the accuracy required in the input data to get a certain accuracy on the PGD solution. Another use of this relation is as stoping criteria for the enrichment of the PGD solution, as the error of PGD cannot overcome the limit imposed by the errors in the separated input data.

Competing interests
The authors declare that they have no competing interests.

Author's contributions
All authors discussed the content of the article based on their previous experiences using PGD. SD and PD prepare and run the numerical examples based on the Poisson problem. EC and DG prepare and run the numerical examples based on the XXX problem. All authors were involved in writing of the article.

Acknowledgements
This work has been partially supported by the Spanish Ministry of Science and Competitiveness, through grant number CICYT-DPI2011-27778-C02-01/02 and by the Generalitat de Catalunya, grant number 2014-SGR-1471.

Author details
[1]LaCàN, UPC, Jordi Girona 1, 08034 Barcelona, Spain. [2]Universidad de Zaragoza, Spain.

References

1. 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 modelling of complex fluids. J. Non-Newtonian Fluid Mech. **139**, 153–176 (2006)
2. Ammar, A., Mokdad, B., Chinesta, F., Keunings, R.: A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids. Part II: transient simulation using space-time separated representations. J. Non-Newtonian Fluid Mech. **144**, 98–121 (2007)
3. Ammar, A., Huerta, A., Chinesta, F., Cueto, E.: Parametric solutions involving geometry: a step towards efficient shape optimization. Comput. Methods Appl. Mech. Eng. **268**, 178–193 (2014)
4. Sergio Zlotnik, D.M. Pedro Díez, Huerta, A.: Proper Generalized Decomposition of a geometrically parametrized heat problem with geophysical applications. Int. J. Numer. Meth. Engng. (2015). accepted for publication, doi: 10.1002/nme.4909
5. González, D., Cueto, E., Chinesta, F.: Real-time direct integration of reduced solid dynamics equations. International Journal for Numerical Methods in Engineering **99**(9), 633–653 (2014)
6. Ammar, A., Chinesta, F., Díez, P., Huerta, A.: An error estimator for separated representations of highly multidimensional models. Comput. Methods Appl. Mech. Eng. **199**, 1872–1880 (2010)
7. Bouclier, R., Louf, F.c.c.o., Chamoin, L.: Real-time validation of mechanical models coupling PGD and constitutive relation error. Computational Mechanics **52**(4), 861–883 (2013)
8. Nadal, E., Leygue, A., Chinesta, F., Beringhier, M., Ródenas, J.J., Fuenmayor, F.J.: A separated representation of an error indicator for the mesh refinement process under the proper generalized decomposition framework. Computational Mechanics **55**(2), 251–266 (2015)
9. Morin, P., Nochetto, R.H., Siebert, K.G.: Data oscillation and convergence of adaptive FEM. SIAM Journal on Numerical Analysis **38**(2), 466–488 (2000)
10. Abdulle, A.: On a priori error analysis of fully discrete heterogeneous multiscale FEM. Multiscale Modeling and Simulation. A SIAM Interdisciplinary Journal **4**(2), 447–459 (2005)
11. E, W., Ming, P., Zhang, P.: Analysis of the Heterogeneous Multiscale Method for Elliptic Homogenization Problems. Journal of the American Mathematical Society **18**(1), 121–156 (2005)
12. Kolda, T.G., Bader, B.W.: Tensor decompositions and applications. SIAM Rev. **51**(3), 455–500 (2009)
13. De Lathauwer, L., De Moor, B., Vandewalle, J.: A multilinear singular value decomposition. SIAM J. Matrix Anal. Appl. **21**(4), 1253–1278 (2000)
14. Carroll, J.D., Chang, J.-J.: Analysis of individual differences in multidimensional scaling via an n-way generalization of "Eckart-Young" decomposition. Psychometrika **35**(3), 283–319 (1970)
15. Harshman, R.A.: Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multimodal factor analysis. UCLA working papers in phonetics **16**, 1–84 (1970)
16. Tucker, L.R.: Some mathematical notes on three-mode factor analysis. Psychometrika **31**, 279–311 (1966)
17. Modesto, D., Zlotnik, S., Huerta, A.: Proper Generalized Decomposition for parameterized helmholtz problems in heterogeneous and unbounded domains: application to harbor agitation. Comput. Methods Appl. Mech. Eng. (2015). accepted for publication

Figure 1 Motivation example: flow around a geometrically parameterized airfoil. The solution of parameterized geometries involves the separation of the Jacobians and, therefore, a truncation error in introduced in the operators.

Figure 2 Motivation example: real-time integration of solid dynamics. The initial conditions of the problem are parameterized. The number of terms in used in the parametric initial conditions determines its accuracy.

Figure 3 Error sources in the PGD solution The approximation introduced by data separation affects the PGD and the FE solutions.

Figure 4 Nodal errors for the separated diffusivity function.

Figure 5 Separation of the diffusivity function using SVD. Top left panel shows the analytic diffusivity function k. Top right panel show the relative weight of the initial 30 terms of the separated functions; the relative weight is computed as the sum of the amplitude of all previous terms, divided by largest amplitude. Lower panels show the functions F and G corresponding to the initial four terms

Figure 6 Evolution of the error of the PGD solution with the number of terms for a k^{sep} function with discretized on the nodes. The separated diffusivity function is discretized on a grid that coincides with the nodal points used to describe the solution u^{sep}.

Figure 7 Grid for the discretization of k^{sep} coinciding with the location of the integration points. Left: mesh for u^{sep} in gray lines and location of integration points. Right: mesh for u^{sep} in thick gray lines and mesh for k^{sep} in thin red lines.

Figure 8 Evolution of the error of the PGD solution with the number of terms. Errors are relative and computed against a FE solution. Each curve corresponds with a PGD solution including a different accuracy of k^{sep}.

Figure 9 Dependence of the final PGD error as a function of the errors introduced by k^{sep}. In the range where the separation error is dominant a linear dependence (having slope equal to one) is obtained.