



HAL
open science

Non-Conventional Numerical Strategies in the Advanced Simulation of Materials and Processes

Hajer Lamari, Francisco Chinesta, Amine Ammar, Elías Cueto

► **To cite this version:**

Hajer Lamari, Francisco Chinesta, Amine Ammar, Elías Cueto. Non-Conventional Numerical Strategies in the Advanced Simulation of Materials and Processes. *International Journal of Modern Manufacturing Technologies*, 2009, 1, pp.49-56. hal-01007107

HAL Id: hal-01007107

<https://hal.science/hal-01007107>

Submitted on 24 Apr 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.

Public Domain

NON-CONVENTIONAL NUMERICAL STRATEGIES IN THE ADVANCED SIMULATION OF MATERIALS AND PROCESSES

Hajer Lamari ¹, Francisco Chinesta ², Amine Ammar ³ & Elias Cueto ⁴

¹ GEM CNRS – Centrale Nantes, 1 rue de la Noe, BP 92101, F-44321 Nantes cedex 3, France

² EADS Foundation International Chair, Centrale Nantes, 1 rue de la Noe, BP 92101, F-44321 Nantes cedex 3, France

³ Laboratoire de Rhéologie, 1301 rue de la piscine, BP 53, Domaine universitaire, F-38041 Grenoble cedex 9, France

⁴ I3A, Universidad de Zaragoza, María de Luna, 7, E-50018 Zaragoza, Spain

Corresponding author: Francisco Chinesta, Francisco.Chinesta@ec-nantes.fr

Abstract: In this work we analyze the possibilities of applying model reduction in the advanced simulation of materials and processes. The use of such strategies allows impressive computing time savings in the numerical simulations of complex models without degrading the solution accuracy. For this purpose we apply proper generalized decompositions of multidimensional models that can be associated to usual models in computational mechanics.

Key words: Proper Generalized Decomposition, Model reduction, Multidimensional models, Curse of dimensionality, Shape optimization, Multi-scale modelling

1. INTRODUCTION

The fine description of the mechanics and structure of materials at the micro, nano and sub-nanometric scales introduces some specific challenges related to the impressive number of degrees of freedom required or to the highly dimensional spaces in which those models are defined. Moreover, usual models encountered in computational physics and mechanics can be transformed in multi-dimensional models allowing for very general solutions as we describe later. Despite the fact that spectacular progresses have been accomplished in the context of computational mechanics in the last decade, the efficient treatment of those multi-dimensional models needs further developments.

The brute force approach cannot be considered as a possibility for treating this kind of models. We can understand the catastrophe of dimension by assuming a model defined in a hyper-cube Ω of dimension D :

$\Omega =]-L, L[^D$. Now, if we define a grid to discretize the model, as it is usually performed in the vast majority of numerical methods (finite differences, finite elements, finite volumes, spectral methods etc.), consisting of M nodes on each direction, the total number of nodes will be M^D . If we assume that for example $M = 10$ (an extremely coarse description) and $D = 80$ (much lower than the usual dimensions required in quantum or statistical mechanics) the number of nodes in Ω reaches the

astronomical value of 10^{80} that represents the presumed number of elementary particles in the universe!

We come back to the practical interest of multi-dimensional models later. In what follows in the present section we are revisiting a technique able to circumvent the curse of dimensionality issue.

1.1 Multidimensional solvers based on the Proper Generalized Decomposition – PGD -

We start writing the polynomial approximation of a generic multi-dimensional function $u(x_1, x_2, \dots, x_D)$ in the whole domain as:

$$u(\mathbf{x}) \approx \sum_{i=1}^{i=N} X_1^i(x_1) \cdots X_D^i(x_D) = \sum_{i=1}^{i=N} \prod_{k=1}^{k=D} X_k^i(x_k) \quad (1)$$

The coordinate x_i is not necessarily one-dimensional, but in any case it is defined in a space of moderate dimensions (1D, 2D or 3D), i.e. $x_i \in \Omega_i$, $\Omega_i \subset \mathcal{R}^{d_i}$, $d_i \leq 3$. The model results then defined in the whole domain $\Omega = \Omega_1 \times \cdots \times \Omega_D \subset \mathcal{R}^{d_1 + \cdots + d_D}$. One of this coordinates could be the time involved in transient models.

It is also well known that several model solutions can be approximated by a finite, and sometimes quite reduced number of functional products. Expression (1) involves $N \times M \times D$ degrees of freedom instead of the M^D required in mesh-based discretization techniques.

In what follows we are describing a new advanced technique that combines a separated representation and an adaptation procedure able to build up gradually each product of functions involved in (1) until reaching the convergence. It has some resemblances with the functional approximation used within the LATIN framework, the radial approximation making use of a space-time separated representation (see [6] and the references therein) as well as with the ones employed in the post-Hartree-

Fock methods [4]. This technique has been successfully applied in a variety of linear, non linear, stationary and transient problems [1] [2] [5] [7]. In what follows we are revisiting the main ideas of such decomposition technique. For the sake of simplicity we are considering a simple multi-dimensional diffusion problem in D dimensions:

$$\begin{cases} \nabla^2 u = f(\mathbf{x}), \mathbf{x}^T = (x_1, \dots, x_D) \in \Omega =]0, L[^D \\ u(\mathbf{x} \in \partial\Omega) = 0 \end{cases} \quad (2)$$

where the general form of the right term is given by

$$f(\mathbf{x}) \approx \sum_{i=1}^{i=m} F_1^i(x_1) \cdots F_D^i(x_D) = \sum_{i=1}^{i=m} \prod_{k=1}^{k=D} F_k^i(x_k) \quad (3)$$

Such a decomposition can be performed by using singular value decomposition.

The iteration scheme used to build up the solution (1) proceeds performing an enrichment of the approximation basis at each iteration. Thus, knowing the approximation at iteration the n :

$$u^n(\mathbf{x}) = \sum_{i=1}^{i=n} X_1^i(x_1) \cdots X_D^i(x_D) = \sum_{i=1}^{i=n} \prod_{k=1}^{k=D} X_k^i(x_k) \quad (4)$$

the approximation basis could be enriched by adding

a new product of functions $\prod_{k=1}^{k=D} X_k^{n+1}(x_k)$

$$u^{n+1}(\mathbf{x}) = u^n(\mathbf{x}) + \prod_{k=1}^{k=D} X_k^{n+1}(x_k) \quad (5)$$

that needs for the determination of the D involved functions $X_k^{n+1}(x_k)$. For this purpose the trial function:

$$u(\mathbf{x}) = \sum_{i=1}^{i=n} \prod_{k=1}^{k=D} X_k^i(x_k) + \prod_{k=1}^{k=D} R_k(x_k) \quad (6)$$

is injected in the weak formulation, where $R_k(x_k) \equiv X_k^{n+1}(x_k)$ are the unknown fields of the non linear system obtained, whose size is $D \times M$. The associated test functions are taken, again in the Galerkin's framework, as:

$$u^*(\mathbf{x}) = \sum_{j=1}^{j=D} \left(R_j^*(x_j) \prod_{\substack{k=1 \\ k \neq j}}^{k=D} R_k(x_k) \right) \quad (7)$$

Now, as soon as the functions $R_k(x_k)$ have been determined, the searched functions $X_k^{n+1}(x_k)$ are obtained by identifying these functions with the converged $R_k(x_k)$ functions.

This algorithm has been successfully used to solve models involving one hundred dimensions needing about $\sim 10^{300}$ degrees of freedom if one proceeds in the finite element framework. The construction of the separated solution only needed of around 20 minutes using Matlab on a standard personal computer!. The separate representation considered in (1) only needs approximations defined in spaces of moderate

dimensions d_i and then integrations in such moderate dimensional spaces because the integral of a product of functions in a hyper-domain can be written as the product of the integrals defined in the domains Ω_i .

1.2 Illustrating the Proper Generalized Decomposition construction

In what follows we are illustrating the construction of the Proper Generalized Decomposition by considering a quite simple problem, the parametric heat transfer equation:

$$\frac{\partial u}{\partial t} - k \Delta u - f = 0 \quad (8)$$

where $(\mathbf{x}, t, k) \in \Omega \times I \times \mathfrak{S}$ and for the sake of simplicity the source term is assumed constant, i.e. $f = cte$. Because the conductivity is considered unknown, it is assumed as a new coordinate defined in the interval \mathfrak{S} . Thus, instead of solving the thermal model for different values of the conductivity parameter we prefer introducing it as a new coordinate. The price to be paid is the increase of the model dimensionality; however, as the complexity of the PGD scales linearly with the space dimension the consideration of the conductivity as a new coordinate allows for faster and cheaper solutions.

The solution of Eq. (8) is searched under the form:

$$u(\mathbf{x}, t, k) \approx \sum_{i=1}^{i=N} X_i(\mathbf{x}) \cdot T_i(t) \cdot K_i(k) \quad (9)$$

In what follows we are assuming that the approximation at iteration n is already done:

$$u^n(\mathbf{x}, t, k) = \sum_{i=1}^{i=n} X_i(\mathbf{x}) \cdot T_i(t) \cdot K_i(k) \quad (10)$$

and at present iteration we look for the next functional product $X_{n+1}(\mathbf{x}) \cdot T_{n+1}(t) \cdot K_{n+1}(k)$ that for alleviating the notation will be denoted by $R(\mathbf{x}) \cdot S(t) \cdot W(k)$. Prior to solve the resulting non linear model related to the calculation of these three functions a model linearization is compulsory. The simplest choice consists in using an alternating directions fixed point algorithm. It proceeds by assuming $S(t)$ and $W(k)$ given at the previous iteration of the non-linear solver and then computing $R(\mathbf{x})$. From the just updated $R(\mathbf{x})$ and $W(k)$ we can update $S(t)$, and finally from the just computed $R(\mathbf{x})$ and $S(t)$ we compute $W(k)$. The procedure continues until reaching convergence. The converged functions $R(\mathbf{x})$, $S(t)$ and $W(k)$ allow defining the searched functions: $X_{n+1}(\mathbf{x}) = R(\mathbf{x})$, $T_{n+1}(t) = S(t)$ and $K_{n+1}(k) = W(k)$.

We are illustrating each one of the just referred steps:

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

We consider the global weak form of Eq. (8):

$$\int_{\Omega \times I \times \mathfrak{S}} u^* \left(\frac{\partial u}{\partial t} - k \Delta u - f \right) d\mathbf{x} dt dk = 0 \quad (11)$$

where the trial and test functions write respectively:

$$u(\mathbf{x}, t, k) = \sum_{i=1}^{i=n} X_i(\mathbf{x}) \cdot T_i(t) \cdot K_i(k) + R(\mathbf{x}) \cdot S(t) \cdot W(k) \quad (12)$$

and

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

Introducing (12) and (13) into (11) it results

$$\begin{aligned} \int_{\Omega \times I \times \mathfrak{S}} R^* \cdot S \cdot W \cdot \left(R \cdot \frac{\partial S}{\partial t} \cdot W - k \cdot \Delta R \cdot S \cdot W \right) d\mathbf{x} dt dk = \\ = - \int_{\Omega \times I \times \mathfrak{S}} R^* \cdot S \cdot W \cdot \mathfrak{R}^{(n)} d\mathbf{x} dt dk \end{aligned} \quad (14)$$

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

Now, being known all the functions involving the time and the parametric coordinate, we can integrate Eq. (14) in their respective domains $I \times \mathfrak{S}$. Integrating in $I \times \mathfrak{S}$ and taking into account the notation

$$\left[\begin{array}{lll} w_1 = \int_{\mathfrak{S}} W^2 dk & s_1 = \int_I S^2 dt & r_1 = \int_{\Omega} R^2 d\mathbf{x} \\ w_2 = \int_{\mathfrak{S}} kW^2 dk & s_2 = \int_I S \cdot \frac{dS}{dt} dt & r_2 = \int_{\Omega} R \cdot \Delta R d\mathbf{x} \\ w_3 = \int_{\mathfrak{S}} W dk & s_3 = \int_I S dt & r_3 = \int_{\Omega} R d\mathbf{x} \\ w_4^i = \int_{\mathfrak{S}} W \cdot K_i dk & s_4^i = \int_I S \cdot \frac{dT_i}{dt} dt & r_4^i = \int_{\Omega} R \cdot \Delta X_i d\mathbf{x} \\ w_5^i = \int_{\mathfrak{S}} kW \cdot K_i dk & s_5^i = \int_I S \cdot T_i dt & r_5^i = \int_{\Omega} R \cdot X_i d\mathbf{x} \end{array} \right] \quad (15)$$

Eq. (14) reduces to:

$$\begin{aligned} \int_{\Omega} R^* \cdot (w_1 \cdot s_2 \cdot R - w_2 \cdot s_1 \cdot \Delta R) d\mathbf{x} = \\ = - \int_{\Omega} R^* \cdot \left(\sum_{i=1}^{i=n} w_4^i \cdot s_4^i \cdot X_i - \sum_{i=1}^{i=n} w_5^i \cdot s_5^i \cdot \Delta X_i - w_3 \cdot s_3 \cdot f \right) d\mathbf{x} \end{aligned} \quad (16)$$

Eq. (16) defines an elliptic steady state boundary value problem that can be solved by using any discretization technique operating on the model weak form (finite elements, finite volumes ...). Another possibility consists in coming back to the strong form of Eq. (16):

$$\begin{aligned} w_1 \cdot s_2 \cdot R - w_2 \cdot s_1 \cdot \Delta R = \\ = - \left(\sum_{i=1}^{i=n} w_4^i \cdot s_4^i \cdot X_i - \sum_{i=1}^{i=n} w_5^i \cdot s_5^i \cdot \Delta X_i - w_3 \cdot s_3 \cdot f \right) \end{aligned} \quad (17)$$

that could be solved by using any collocation technique (finite differences, SPH ...).

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

In the present case the test function writes:

$$u^*(\mathbf{x}, t, k) = S^*(t) \cdot R(\mathbf{x}) \cdot W(k) \quad (18)$$

Now, the weak form reads

$$\begin{aligned} \int_{\Omega \times I \times \mathfrak{S}} S^* \cdot R \cdot W \cdot \left(R \cdot \frac{\partial S}{\partial t} \cdot W - k \cdot \Delta R \cdot S \cdot W \right) d\mathbf{x} dt dk = \\ = - \int_{\Omega \times I \times \mathfrak{S}} S^* \cdot R \cdot W \cdot \mathfrak{R}^{(n)} d\mathbf{x} dt dk \end{aligned} \quad (19)$$

that integrated in the domain $\Omega \times \mathfrak{S}$ and taking into account the notation (15) results:

$$\begin{aligned} \int_I S^* \cdot \left(w_1 \cdot r_1 \cdot \frac{dS}{dt} - w_2 \cdot r_2 \cdot S \right) dt = \\ = - \int_I S^* \cdot \left(\sum_{i=1}^{i=n} w_4^i \cdot r_5^i \cdot \frac{dT_i}{dt} - \sum_{i=1}^{i=n} w_5^i \cdot r_4^i \cdot T_i - w_3 \cdot r_3 \cdot f \right) dt \end{aligned} \quad (20)$$

Eq. (20) represents the weak form of the ODE defining the time evolution of the field S that can be solved by using any stabilized discretization technique (SU, Discontinuous Galerkin, ...). The strong form of Eq. (20) reads:

$$\begin{aligned} w_1 \cdot r_1 \cdot \frac{dS}{dt} - w_2 \cdot r_2 \cdot S = \\ = - \left(\sum_{i=1}^{i=n} w_4^i \cdot r_5^i \cdot \frac{dT_i}{dt} - \sum_{i=1}^{i=n} w_5^i \cdot r_4^i \cdot T_i - w_3 \cdot r_3 \cdot f \right) \end{aligned} \quad (21)$$

than can be solved by using backward finite differences, or higher order Runge-Kutta schemes, among many other possibilities.

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

In the present case the test function writes:

$$u^*(\mathbf{x}, t, k) = W^*(t) \cdot R(\mathbf{x}) \cdot S(k) \quad (22)$$

Now, the weak form reads

$$\begin{aligned} \int_{\Omega \times I \times \mathfrak{S}} W^* \cdot R \cdot S \cdot \left(R \cdot \frac{\partial S}{\partial t} \cdot W - k \cdot \Delta R \cdot S \cdot W \right) d\mathbf{x} dt dk = \\ = - \int_{\Omega \times I \times \mathfrak{S}} W^* \cdot R \cdot S \cdot \mathfrak{R}^{(n)} d\mathbf{x} dt dk \end{aligned} \quad (23)$$

that integrated in $\Omega \times I$ and taking into account the notation (15) results:

$$\begin{aligned} \int_{\mathfrak{S}} W^* \cdot (r_1 \cdot s_2 \cdot W - r_2 \cdot s_1 \cdot W) dk = \\ = - \int_{\mathfrak{S}} W^* \cdot \left(\sum_{i=1}^{i=n} r_5^i \cdot s_4^i \cdot K_i - \sum_{i=1}^{i=n} r_4^i \cdot s_5^i \cdot K_i - r_3 \cdot s_3 \cdot f \right) dk \end{aligned} \quad (24)$$

Eq. (24) does not involve any differential operator. The strong form of Eq. (24) reads:

$$\begin{aligned} & (r_1 \cdot s_2 - r_2 \cdot s_1) \cdot W = \\ & = - \left(\sum_{i=1}^{i=n} (r_5^i \cdot s_4^i - r_4^i \cdot s_5^i) \cdot K_i - r_3 \cdot s_3 \cdot f \right) \end{aligned} \quad (25)$$

that represents an algebraic equation. Thus, the introduction of parameters as additional model coordinates has not a noticeable effect in the computational cost.

There are other minimization strategies more robust and exhibiting faster convergence for building-up the PGD (see [10]).

3. APPLICATIONS

In this section we are analyzing different applications of the proper generalized decomposition in computational mechanics, and in particular in the advanced simulation of material and processes, through some academic examples.

3.1 Parametric models

We consider the 1D heat equation defined by:

$$\frac{\partial u}{\partial t} - \frac{\partial}{\partial x} \left(k \frac{\partial u}{\partial x} \right) = f(t, x); \quad t \in \Omega_t, \quad x \in \Omega_x \quad (26)$$

If k is constant, this equation can be solved for every value of k using the separated representation previously illustrated without further difficulties. Now, we are focusing on a more complicated and realistic problem. In general, for homogenous materials the thermal conductivity depends on temperature, following a linear dependence:

$$k = au + b \quad (27)$$

If we introduce this expression into Eq. (26) the resulting heat equation writes:

$$\frac{\partial u}{\partial t} - b \frac{\partial^2 u}{\partial x^2} - au \frac{\partial^2 u}{\partial x^2} - a \left(\frac{\partial u}{\partial x} \right)^2 = f \quad (28)$$

We want to solve this equation for any value of a and b . We can easily understand that the computed solution allows efficient optimization and inverse identification strategies. For that purpose the following separated representation approximation is considered:

$$u(t, x, a, b) \approx \sum_{i=1}^N T_i(t) \cdot X_i(x) \cdot A_i(a) \cdot B_i(b) \quad (29)$$

The numerical solution was carried out for $t \in [0, 1]$, $x \in [-1, 1]$, $a \in [-1, 1]$, $b \in [1, 5]$ being the initial condition $u(t=0, x) = 1 - x^{10}$, the temperature vanishing at both boundaries. In this example, 500 nodes were employed for discretizing each of the 4 coordinates. A mesh would have involved 500^4 degrees of freedom, but using the PGD this impressive number reduces to the order of 2000. The

solution was performed in few minutes using Matlab on a personal laptop. Fig. 1 depicts the solution at point $x=0$ at the final time $t=1$ as a function of both parameters a and b .

The non linearity was accounted by assuming the conductivity given at previous iteration, i.e.

$k = au^n + b$, where

$$u^n(t, x, a, b) = \sum_{i=1}^n T_i(t) \cdot X_i(x) \cdot A_i(a) \cdot B_i(b) \quad (30)$$

Other linearizations were analyzed and discussed in [3] and [8].

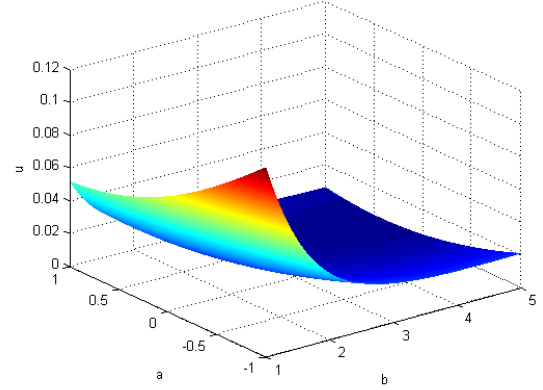


Fig. 1. Temperature at point $x=0$ at the final time $t=1$ as a function of both parameters a and b .

Now, we come back to the non-linear thermal model (28), with $a=1$ and $b=1$, but we are focusing on its steady solution for a source term given by $f = -\beta(1-x^2)$, $\beta \in [0, 1]$. Now, the solution is searched from the finite sum decomposition:

$$u(x, \beta) \approx \sum_{i=1}^N X_i(x) \cdot B_i(\beta) \quad (31)$$

The computed solution is depicted in figure 2 where we can notice, as expected, that the solution vanishes everywhere when $\beta=0$.

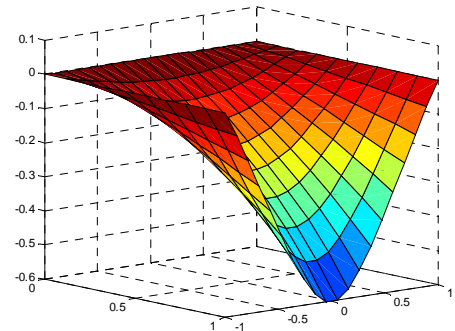


Fig. 2. Steady state temperature field $u(x, \beta)$.

Finally, we are considering the same problem, the steady state solution of a thermal model (non-linear) in which the heat source is punctual and can be applied everywhere in the domain. The thermal model is then defined as follows:

$$\frac{\partial}{\partial x} \left(k \frac{\partial u}{\partial x} \right) = f(x) \quad (32)$$

where the source term writes in the present case:

$$f(x) = \beta \cdot \delta(x - x') \quad (33)$$

being $\delta(\cdot)$ the Dirac's mass and x' the point in which the thermal load of intensity β applies. As we are interested in solving the model (32) for any position of the thermal load x' and any value of the intensity β , we introduce the load location and the load intensity as new coordinates, searching the solution in the form:

$$u(x, x', \beta) \approx \sum_{i=1}^N X_i(x) \cdot S_i(x') \cdot B_i(\beta) \quad (34)$$

By applying the procedure described in the previous section one could determine easily the functions $X_i(x)$ by solving a second order elliptic problem, then function $S_i(x')$ by solving an algebraic equation (there are not derivatives in that coordinate in the thermal model model) and finally function $B_i(\beta)$ solving another algebraic equation.

The strategy here described allows an off-line pre-calculus and very fast post-processing required in many branches of computational sciences with real-time simulation purposes.

3.2 Models defined in evolving domains

The main issue in treating models defined in evolving domains lies in the fact that \mathbf{x} belongs to a domain that is evolving in time, i.e. $\mathbf{x} \in \Omega(t)$. Obviously, in that case we cannot apply the procedure described previously, but the deepest difficulty lies in the fact that the space coordinate is not independent of the time coordinate.

The simplest alternative for circumventing this difficulty consists in defining the mapping between the initial space coordinates $\mathbf{X} \in \Omega(t=0)$ and the present ones $\mathbf{x} \in \Omega(t)$. This mapping writes:

$\mathbf{x} = \mathbf{x}(\mathbf{X}, t)$. Now, the model given by $\mathbb{L}_{x,t}(u(\mathbf{x}, t)) = \mathbb{F}$ (where $\mathbb{L}_{x,t}(\cdot)$ denotes a differential operator involving the present coordinates \mathbf{x} and t) is redefined by considering (\mathbf{X}, t) as new coordinates (now both being independent): $\mathbb{L}_{\mathbf{X},t}^*(u(\mathbf{X}, t)) = \mathbb{F}^*$ and solved by using the natural decomposition:

$$u(\mathbf{X}, t) \approx \sum_{i=1}^N X_i(\mathbf{X}) \cdot T_i(t) \quad (35)$$

We are at present analyzing this approach that suggests fully Lagrangian formulation in thermo-mechanical problems. As the procedure described in

section 1.2 does not enforce the balance equations at any particular time, fully Lagrangian formulations could run as soon as the mapping $\mathbf{x} = \mathbf{x}(\mathbf{X}, t)$ is well defined even in very large geometrical transformations.

In what follows we are illustrating this procedure. For this purpose we consider the filling process of a 1D porous domain by an incompressible fluid. The flow in porous media is usually modelled by the Darcy's law that establishes the proportionality between the flow velocity and the gradient of pressures. In the 1D case that equation writes:

$$v = -\frac{k}{\eta} \frac{dp}{dx} \quad (36)$$

where v is the flow velocity, k is the porous medium permeability, η is the fluid viscosity (assumed constant) and p represents the pressure field. In what follows we assume without loss of generality $k/\eta = 1$. Introducing Eq. (36) into the mass balance equation of an incompressible fluid that in the 1D case writes

$$\frac{dv}{dx} = 0 \quad (37)$$

it results:

$$\frac{d^2 p}{dx^2} = 0 \text{ in } \Omega_f(t) \quad (38)$$

where $\Omega_f(t)$ represents the domain occupied by the fluid at time t , $\Omega_f(t) =]0, L_t[$. We assume that $\Omega_f(t=0) =]0, L_0[$. If the injection process is performed at constant flow rate (a), then the boundary conditions write in the case here considered:

$$-\frac{k}{\eta} \frac{dp}{dx} \Big|_{x=0} = a \Rightarrow \frac{dp}{dx} \Big|_{x=0} = -a \quad (39)$$

and

$$p(x = L_t) = 0 \quad (40)$$

where $L_t = L_0 + a \cdot t$.

The coordinate's transformation writes:

$$x = X \cdot \left(1 + \frac{a \cdot t}{L_0} \right) \quad (41)$$

where $X \in [0, L_0]$ and $x \in [0, L_t]$. Now, instead of solving the problem (38) in $\Omega_f(t)$, we are solving this problem in the reference domain $\Omega_f(t=0) =]0, L_0[$. Applying the coordinates transformation (41), equation (38) results:

$$\frac{d^2 p}{dX^2} \left(\frac{dX}{dx} \right)^2 = \frac{d^2 p}{dX^2} \left(\frac{L_0}{L_0 + a \cdot t} \right)^2 = 0 \quad (42)$$

which is defined in $\Omega_f(t=0)$.

The boundary conditions become:

$$\frac{dp}{dX} \Big|_{X=0} \left(\frac{L_0}{L_0 + a \cdot t} \right) = -a \quad (43)$$

and

$$p(X = L_0) = 0 \quad (44)$$

One could try to solve Eq. (43) for each time, by considering the time t as a model parameter. Obviously as in the present case the resulting problem is two-dimensional it could be solved using any standard discretization technique (finite elements, ...), however, in more complex scenarios the physical space will contain more coordinates (2D or 3D) and the coordinates transformation will imply many parameters. Thus, the resulting general parametric problem will be multi-dimensional and the use of the separated representation described in the previous section should be mandatory.

Taking into account Eqs. (43) and (44), the solution of Eq. (42) results:

$$p(X, t) = a \cdot (L_0 + a \cdot t) - \frac{a}{L_0} \cdot (L_0 + a \cdot t) \cdot X \quad (46)$$

in $[0, L_0] \times [0, T_{\max}]$, that represents a separated representation:

$$p(X, t) = \sum_{i=1}^2 X_i(X) \cdot T_i(t) \quad (47)$$

with

$$\begin{cases} X_1(X) = 1 \\ T_1(t) = a \cdot (L_0 + a \cdot t) \\ X_2(X) = X \\ T_2(t) = -\frac{a}{L_0} \cdot (L_0 + a \cdot t) \end{cases} \quad (48)$$

We can notice that the injection pressure P_i goes up in time in order to keep the solution slope constant (constant flow rate):

$$P_i = p(X = 0, t) = a \cdot (L_0 + a \cdot t) \quad (49)$$

3.3 Multi-scale in time

Consider the simple model

$$\frac{du}{dt} = 2t \cos^2(\omega t) - 2t^2 \omega \cos(\omega t) \sin(\omega t) \quad (50)$$

with $t \in I =]0, 10]$ and zero initial condition. The exact solution writes $u^{ex} = t^2 \cos^2(\omega t)$. When the frequency increases, the time step must be reduced in order to capture the load evolution. Fig. 3 compares for a frequency $\omega = 10$ the computed solution by using a time step of $\Delta t = 0.001$ (high resolution curve) and $\Delta t = 0.5$ (low resolution curve).

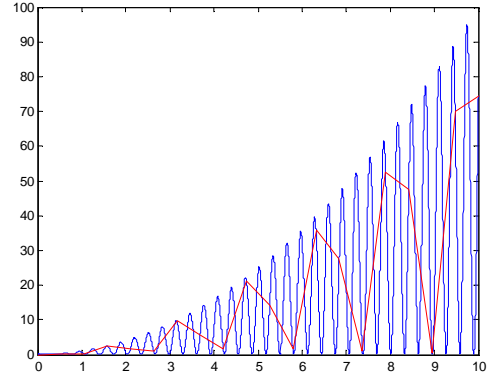


Fig. 3. Solution of model (50) with two different sampling times.

For accounting with the two time scales involved in the model (that we consider separated) we introduce two different times, both assumed independent: $T = t \in I$ and $\tau = \omega t \in [0, 2\pi]$. Thus, we postulate

$u(T, \tau)$ and Eq. (50) becomes:

$$\begin{aligned} \frac{du(T, \tau)}{dt} &= \frac{\partial u}{\partial T} \cdot \frac{\partial T}{\partial t} + \frac{\partial u}{\partial \tau} \cdot \frac{\partial \tau}{\partial t} = \frac{\partial u}{\partial T} + \omega \frac{\partial u}{\partial \tau} = \\ &= 2T \cos^2(\tau) - 2T^2 \omega \cos(\tau) \sin(\tau) \end{aligned} \quad (51)$$

whose solution is searched by assuming the decomposition:

$$u(T, \tau) \approx \sum_{i=1}^N F_i(T) \cdot G_i(\tau) \quad (52)$$

Obviously, in more complex multi-scale models involving different scales in space ($\mathbf{x}_1, \mathbf{x}_2, \dots$) and time (t_1, t_2, \dots) we should write:

$$u(t_1, t_2, \dots, \mathbf{x}_1, \mathbf{x}_2, \dots) \quad (53)$$

and in that case a separated representation seems compulsory.

In the present case the decomposition involves a single product of functions:

$$u(T, \tau) = F(T) \cdot G(\tau) = T^2 \cdot \cos^2(\tau) \quad (54)$$

that correspond with the exact solution, both depicted in Fig. 4. In that figure we depict also the reconstructed solution $u(T, \tau)$.

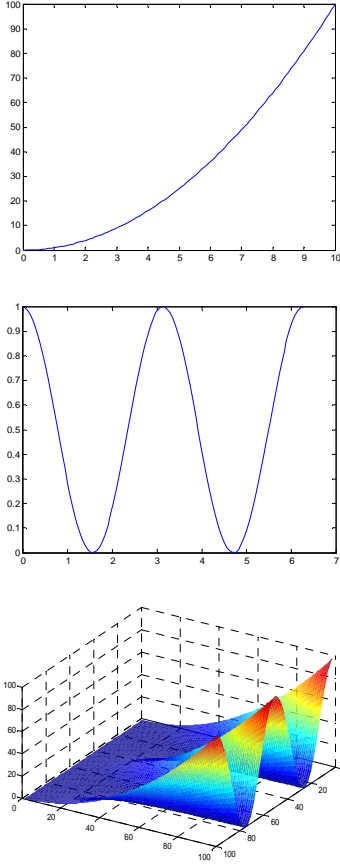


Fig. 4. Functions $F(T)$ (top) and $G(\tau)$ (middle) associated with the decomposition of $u(T, \tau)$ in (51) and well as the reconstructed solution $u(T, \tau)$ (bottom).

3.4 Heterogeneous materials

In this section we are focusing in the thermal models defined in heterogeneous materials. Imagine a composite material involving a matrix and a fibrous reinforcement. A typical representative volume showing the microstructure heterogeneity is depicted in figure 5.

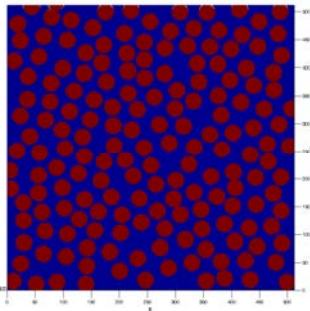


Fig. 5. Microstructure of a composite material

Now, in order to apply the PGD one should perform a separated representation of the thermal conductivity, allowing for an efficient thermal simulation, i.e.

$$k(\mathbf{x}) \approx \sum_{i=1}^P K_x^i(x) \cdot K_y^i(y) \quad (55)$$

This separated representation can be performed by applying the SVD (singular value decomposition) to the matrix containing as entries the image pixels. However, because the irregular distribution of the inclusions the number of sums in (55) becomes very high. Different possibilities exist to alleviate this representation some of them are being analyzed at present. A first possibility lies in considering only a reduced number of the modes of the SVD. The computational cost is significantly reduced and sometimes the precision is not too much degraded. Figure 6 depicts the image reconstruction for different number of sums in (55).

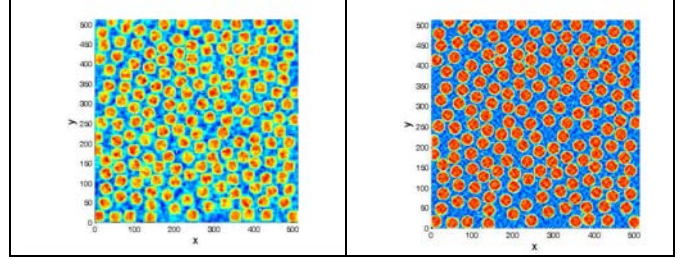


Fig. 6. Reconstructed microstructure for $P = 23$ (left) and $P = 46$ in (55).

Another possibility lies in the substitution of each circle by a square parallel to the coordinate axes, whose area equals the one of the associated circle. Eq. (56) represents a generic rectangle:

$$\mathbf{x} \in [x_a, x_b] \times [y_a, y_b] \quad (56)$$

Now, we define the characteristic function:

$$\chi_{a,b}(s) = \begin{cases} 0 & \text{if } s < a \\ 1 & \text{if } a \leq s \leq b \\ 0 & \text{if } s > b \end{cases} \quad (57)$$

Using this notation, the square defined in (56) can be represented in a separated form by:

$$\chi_{x_a, x_b}(x) \times \chi_{y_a, y_b}(y).$$

Obviously, if the representative volume contains P inclusions, each one represented by a square, the separated representation (55) will contain P sums.

Obviously, all these approaches imply the solution of a thermal model for each representative volume (as soon as the microstructure evolve, the thermal conductivity is modified and then a new solution of the thermal model is required). If we consider a stochastic nature of the microstructure many realizations of the microstructure must be solved. One possibility for alleviating this task consists of considering the representative volume composed of a certain number of cells related to a grid of the representative volume as depicted in figure 7. Now, rather than solving a thermal model for each possible microstructure (represented by the different values of the thermal conductivity in each cell), we are introducing the thermal conductivity of each cell as an extra coordinate. In the example depicted in Fig.

7 the model will be defined in a space of dimension 27 (the x and y space coordinates and the 5×5 cells thermal conductivities).

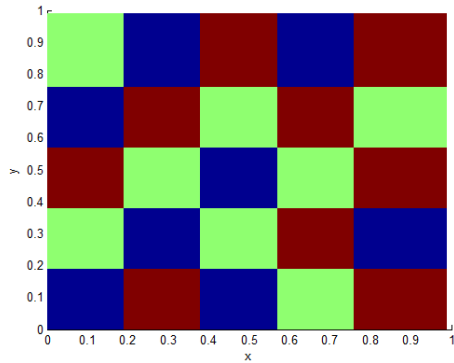


Fig. 7. Cellular microstructure of the representative volume.

Thus, the solution is searched under the form:

$$u(x, y, k_{1,1}, \dots, k_{5,5}) \approx \sum_{i=1}^N X_i(x) \cdot Y_i(y) \cdot K_{1,1}^i(k_{1,1}) \dots K_{5,5}^i(k_{5,5}) \quad (58)$$

Thus, the thermal field for any possible microstructure only needs the solution of only one multidimensional problem, solution that can be performed efficiently by applying the PGD. As soon as the solution (58) is computed, the thermal field for any microstructure realization is obtained by substituting the known conductivities of each cell in Eq. (58). Thus, the thermal field related to the microstructure shown in Fig. 7 is depicted in Fig. 8 (a simple boundary condition $u(\mathbf{x} \in \partial\Omega, k_{1,1}, \dots, k_{5,5}) = x$ was enforced)

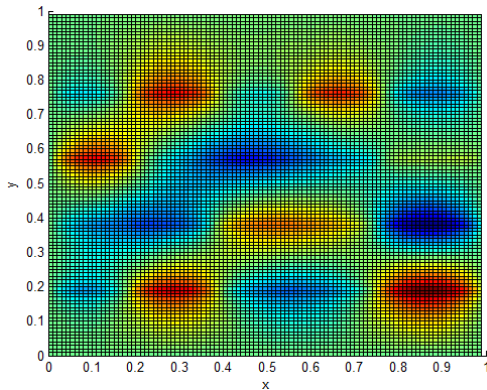


Fig. 8. Thermal field associated with the microstructure depicted in Fig. 7.

4. CONCLUSIONS

This paper explores some possibilities related to the use of Proper Generalized Decomposition in the advanced simulation of materials and processes. This novel discretization technique allows the efficient solution of highly multidimensional models by circumventing the redoubtable curse of dimensionality that mesh based discretization techniques suffer.

As soon as an efficient solver for multidimensional models is available, many models of computational

mechanics can be rewritten in higher dimensional spaces. Thus one could for example compute the thermal field for any value of the thermal conductivity (as illustrated in this paper for homogeneous and heterogenous materials) simplifying inverse identification of optimization; compute the solution for any geometry for addressing evolving domains or shape optimization; or addressing multi-scale models in space or time encountered in many manufacturing processes, as for example the ones involving ultrasons, microwaves or localization.

5. REFERENCES

1. Ammar, A., Mokdad, B., Chinesta, F., Keunings, R. (2006). 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.
2. Ammar, A., Mokdad, B., Chinesta, F., Keunings, R. (2007). A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modelling of complex fluids. Part II: Transient simulation using space-time separated representations. *J. Non-Newtonian Fluid Mech.*, 144, 98-121.
3. Ammar, A., Normandin, M., Daim, F., Gonzalez, D., Cueto, E., Chinesta, F. Non-incremental strategies based on separated representations: Applications in computational rheology. *Communications in Mathematical Sciences*, In press.
4. Cancès, E., Defranceschi, M., Kutzelnigg, W., Le Bris, C., Maday, Y. (2003). Computational quantum chemistry: A primer. *Handbook of Numerical Analysis*, Elsevier, Vol. X, 3-270.
5. Chinesta, F., Ammar, A., Lemarchand, F., Beauchene, P., Boust, F. (2008). Alleviating mesh constraints: Model reduction, parallel time integration and high resolution homogenization. *Comput. Methods Appl. Mech. Engrg.*, 197, 400-413.
6. Ladeveze, P. (1999). *Nonlinear computational structural mechanics*. Springer, NY.
7. Mokdad, B., Prulière, E., Ammar, A., Chinesta, F. (2007). On the simulation of kinetic theory models of complex fluids using the Fokker-Planck approach. *Applied Rheology*, 17/2, 26494, 1-14.
8. Pruliere, E., Chinesta, F., Ammar, A. On the deterministic solution of parametric models by using the proper generalized decomposition. *Mathematics and Computer Simulation*. Submitted.