CSMA 2017
13ème Colloque National en Calcul des Structures
15-19 Mai 2017, Presqu'île de Giens (Var)

A scalable implementation of a volume coupling scheme for the modeling of polycrystalline materials

T. Milanetto Schlittler[1], R. Cottereau[2]

[1] *MSSMat , CentraleSupélec, thiago.milanetto-schlittler@centralesupelec.fr*
[2] *MSSMat , CentraleSupélec, regis.cottereau@centralesupelec.fr*

Résumé — We will show in this presentation a new implementation of a multi-scale, multi-model coupling algorithm, based on the Arlequin framework. We propose a parallelization scheme for the construction of the coupling terms between the models, resulting into a scalable algorithm on large computer clusters. As an application example, we will consider a system composed by an homogeneous, macroscopic elasto-plastic model and an anisotropic polycrystalline material model.
Mots clés — Model coupling, multi-scale models, Arlequin method, FETI, polycrystalline material model, homogenization

1 Motivation

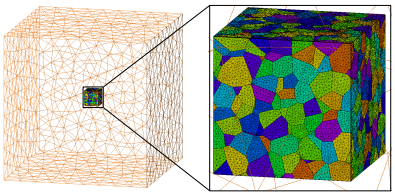


FIGURE 1 – Example of a micro scale heterogeneous polycrystalline domain coupled to a homogeneous macro scale model.

We will focus here on the study of polycrystalline materials through the usage of numeric multi-model coupling methods. This class of materials include most metals and many types of ceramics, which are composed by several grains with different crystalline orientations. They are of high interest to many fields of engineering and industry, due to their usage in systems that sustain extreme conditions. Due to their complexity, many different scales and models are associated to them. Here, we interested in two of them (Figure 1) : a *micro scale*, where a heterogeneous elasto-plastic model is considered, and a *macro scale*, with a homogeneous model. Simulations can focus on a single scale, but this entails a trade-off between the numerical cost of the former, limiting it to small applications, and the lack of detail of the latter when analyzing phenomena such as stress localization, fracture or fatigue.

The motivation of this work is to identify the properties of the macro scale model knowing those of the heterogeneous and micro scale model. From the numerical point of view, one can perform a homogenization [11][7] on the multi-scale system. Such methods are very sensitive to the boundary conditions and to the ratio between the characteristic scales of the models, though, and can introduce a bias to the resulting homogenized macro scale tensor if this ratio is too high, due to the effects of the macro scale boundary conditions on the micro scale model. In the cases where a stochastic model is used to represent the micro scale, this bias is still present even after using an infinite number of Monte-Carlo realizations [1]. It is possible to use periodic boundary conditions to reduce it, but this method does not work optimally for micro structures that are not themselves periodic. In [1], an algorithm based on the Arlequin method [1][3] is proposed to reduce this bias, in the context of a micro scale stochastic model.

This work benefited from French state funding managed by the National Research Agency under project number ANR-14-CE07-0007 CouESt.

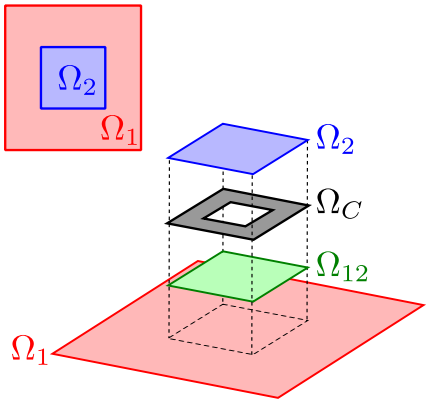


FIGURE 2 – Domains used in the Arlequin framework. The model domains are Ω_1 and Ω_2, the overlapping domain is Ω_{12}, and the gluing zone (marked in gray) is Ω_C.

2 Outline

We will show in this presentation a scalable implementation of the Arlequin algorithm, developed to be, ultimately, used in the context of the numerical homogenization proposed in [1]. It parallelizes not only the resolution of both macro and micro systems, but also the coupling step - which is not usually parallelized, breaking the scalability as a whole. More precisely, we will present a new, parallel intersection search algorithm (needed by the coupling construction step) and a scalable implementation of a solver for the Arlequin method, based on the FETI method [6, 2]. Before we start, we should note that, while we focus here in the Arlequin framework, the parallelization scheme presented here is applicable to other couping methods following similar formulations, such as the non-overlapping domain decomposition methods [4]. As an application example, we will study a macroscopic system with an homogeneous elasticity model coupled to a microscopic system with an heterogeneous, anisotropic elasticity model.

3 Arlequin formulation

Consider two models, each associated to bounded regular domains Ω_1 and Ω_2, and which overlap over a region Ω_{12}. The latter is decomposed into two non-overlapping regions, $\Omega_{12} = \Omega_C \cup \Omega_F$, with Ω_C defining the gluing zone between the two models, and effectively serving as the mediator between them. Figure 2 shows the model domains. Let us note as $\mathbf{W}_1$ and $\mathbf{W}_2$ the functional spaces associated to these models, and as $\mathbf{M}$ a functional space associated to the mediator domain. The Arlequin formulation of the coupled problem can be written then as follows : find $(\underline{u}_1, \underline{u}_2, \underline{\Phi}) \in \mathbf{W}_1 \times \mathbf{W}_2 \times \mathbf{M}$ such that

$$a_1(\underline{u}_1, \underline{v}_1) + c(\underline{\Phi}, \underline{v}_1) = \ell_1(\underline{v}_1), \qquad \forall \underline{v}_1 \in \mathbf{W}_1; \tag{1a}$$

$$a_2(\underline{u}_2, \underline{v}_2) - c(\underline{\Phi}, \underline{v}_2) = \ell_2(\underline{v}_2), \qquad \forall \underline{v}_2 \in \mathbf{W}_2; \tag{1b}$$

$$c\left(\underline{\Psi}, \underline{u}_1 - \underline{u}_2\right) = 0, \qquad \forall \underline{\Psi} \in \mathbf{M}, \tag{1c}$$

In Equation (1), a_l and ℓ_l are the internal and external works associated to the model l, weighted in such a way to guarantee the energy partitioning between the models over the coupling region (see [3] for details). The weak formulations of the models are also modified by adding a coupling operator $c(\cdot,\cdot)$ and a Lagrange multiplier $\underline{\Phi}$, defined on the mediator space. The third equation guarantees that the solutions $\underline{u}_1$ and $\underline{u}_2$ are continuous under the coupling operation. $c(\cdot,\cdot)$ is defined as

$$c(\underline{\Psi}, \underline{\Phi}) = \int_{\Omega_C} \kappa \left(\underline{\underline{\varepsilon}}(\underline{\Psi}) : \underline{\underline{\varepsilon}}(\underline{\Phi}) + \frac{1}{e^2} \underline{\Psi} \cdot \underline{\Phi} \right) d\Omega. \tag{2}$$

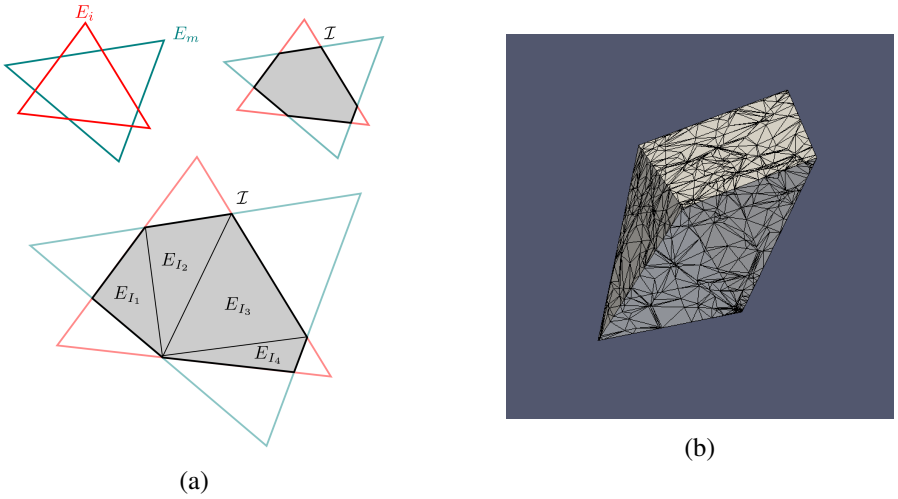
The domains Ω_1 and Ω_2 can be discretized by associating to them, respectively, the meshes $\mathcal{T}_1$ and $\mathcal{T}_2$. Similarly, the domain Ω_C can be discretized by a mesh $\mathcal{T}_C$, but in most cases it is more convenient to represent $\mathbf{M}$ using a mediator mesh $\mathcal{M}$ different from $\mathcal{T}_C$. This mediator mesh can be, for example,

formed by the ensemble of elements of one of systems' meshes that intersect the domain Ω_C. Using this discretization, the formulation (1) can be rewritten as

$$\begin{bmatrix} \underline{\underline{K}}_1 & 0 & \underline{\underline{C}}_1^T \\ 0 & \underline{\underline{K}}_2 & -\underline{\underline{C}}_2^T \\ \underline{\underline{C}}_1 & -\underline{\underline{C}}_2 & 0 \end{bmatrix} \begin{bmatrix} \underline{u}_1 \\ \underline{u}_2 \\ \underline{\Phi} \end{bmatrix} = \begin{bmatrix} \underline{F}_1 \\ \underline{F}_2 \\ 0 \end{bmatrix}. \tag{3}$$

Following standard notation, $\underline{\underline{K}}_l$ and $\underline{F}_l$ are the tensors and vectors representing the internal and external virtual works of the model l.

4 Coupling construction and intersection search



(a) (b)

FIGURE 3 – (a) Intersection between two elements $E_m \in \mathcal{M}$ (blue) and $E_l \in \mathcal{T}_l$ (red), corresponding to a non-zero coupling term $\underline{\underline{C}}^l_{i,j}$. The intersection I (gray) does not follow the geometry of either of these two meshes, and thus it must be triangulated to allow the calculation of the coupling term ; (b) example of an intersection mesh "chunk" generated by our algorithm.

The coupling tensors $\underline{\underline{C}}_l$ in Equation (3) are rectangular matrices, with each element (i, j) given by applying (2) to the corresponding form functions :

$$\underline{\underline{C}}^l_{i,j} = \int_{\Omega^c_{12}} \kappa \left(\underline{\underline{\varepsilon}}(\nu^m_i) : \underline{\underline{\varepsilon}}\left(\nu^l_j\right) + \frac{1}{e^2} \nu^m_i \cdot \nu^l_j \right) d\Omega, \tag{4}$$

ν^m_i and ν^l_j are form functions associated respectively to the coupling region and the model l. These form functions are defined on different spaces, and are associated to incompatible meshes. A common mesh, upon which the form functions ν^m_i and ν^l_j can be projected, must be defined to evaluate the coupling matrices using Equation (4). An example of such a mesh is the mesh I_l, defined by triangulating the intersections between the meshes $\mathcal{T}_l$ and $\mathcal{M}$ (Figure 3).

Several serial search algorithms for two meshes exist in the literature, organizing the bounding boxes of the elements into hierarchical data structures [9], or using the neighbor information and advancing front methods to reduce the number of operations [5]. In our case, we also want to restrict the intersections to the mediator domain Ω_C, which effectively translates into finding the intersections between three meshes : $\mathcal{T}_l$, $\mathcal{M}$ and $\mathcal{T}_C$.

We can take advantage of this difference to parallelize the intersection search. Essentially, we can use the mesh $\mathcal{T}_C$ to partition the intersection search into smaller problems. For each element $E_C \in \mathcal{T}_C$, two patches formed by the elements of $\mathcal{T}_l$ and $\mathcal{M}$ overlapping E_C can be constructed (Figure 4). The intersection search problem is then reduced into $|\mathcal{T}_C|$ smaller intersection problems, which can be solved using serial intersection search methods. Furthermore, since the mesh $\mathcal{T}_C$ has no physical system associated to it, we can choose its element size and partitioning in such a way to better distribute these smaller searches over the processors (see Figure 5 for the strong scaling of the intersection search and construction using this algorithm).

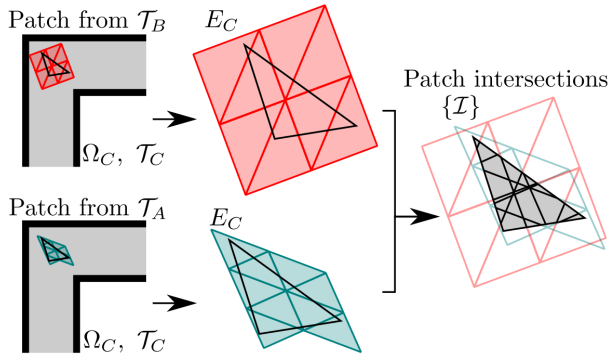
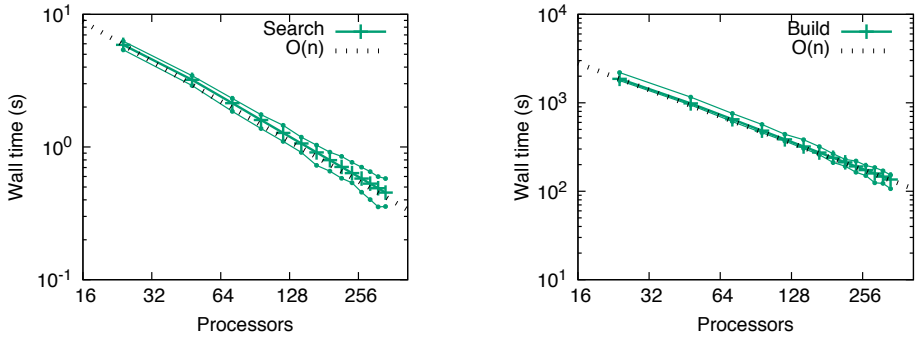


FIGURE 4 – Graphical representation of our intersection search algorithm. For each element E_C from the coupling mesh $\mathcal{T}_C$, two mesh "patches" are built, and then the intersections between these meshes are constructed.



(a) (b)

FIGURE 5 – (a) Intersection search and (b) intersection construction strong scaling, wall time vs. number of processors.

5 Arlequin / FETI solver

The system (3) can be solved as a monolithic problem, but this approach has the limitation of not allowing the usage of the proper solvers of the super-imposed models. One can think, for example, of the coupling between models such as linear / nonlinear, deterministic / stochastic, continuum / atomistic ... . Some works have been proposed to solve this problem, adapting domain decomposition methods such as the FETI [6, 2] or the LATIN [8] methods. In the context of the Arlequin problem, each model corresponds to a different domain of the decomposition, and the coupling to the interface terms.

We focus here on a Arlequin solver based on the FETI method, presented in ref. [2]. In this context, the FETI method essentially consists on solving each model inside its domain, without the coupling effects, and then calculating the coupling correction. The final solution is

$$\underline{u}_1 = \underline{u}_1^0 - \underline{\underline{K}}_1^{-1} \underline{\underline{C}}_1^T \underline{\Phi}, \tag{5}$$

$$\underline{u}_2 = \underline{u}_2^0 + \underline{\underline{K}}_2^{-1} \underline{\underline{C}}_2^T \underline{\Phi}, \tag{6}$$

where $\underline{u}_1^0$ and $\underline{u}_2^0$ are the solutions of the decoupled models, $\underline{\underline{K}}_l \underline{u}_l^0 = \underline{F}_l$, $l = 1, 2$. The Lagrange multiplier $\underline{\Phi}$ is obtained from the system

$$\left(\underline{\underline{C}}_1 \underline{\underline{K}}_1^{-1} \underline{\underline{C}}_1^T + \underline{\underline{C}}_2 \underline{\underline{K}}_2^{-1} \underline{\underline{C}}_2^T \right) \cdot \underline{\Phi} = \left(\underline{\underline{C}}_1 \underline{u}_1^0 - \underline{\underline{C}}_2 \underline{u}_2^0 \right). \tag{7}$$

The system (7) can be solved iteratively using an iterative algorithm, such as the Conjugate Gradient (CG) method. Explicitly assembling the system matrix, though, poses a serious scalability problem to this algorithm, due to the presence of the inverse matrices $\underline{\underline{K}}_l^{-1}$. These matrices are dense, and hence their explicit construction in a parallel algorithm causes a performance bottleneck due to the communications between the processors.

In our implementation of the Arlequin / FETI solver, we avoid this problem by exchanging each application of a term $\underline{\underline{C}}_l \underline{\underline{K}}_l^{-1} \underline{\underline{C}}_l^T$ by a multiplication by $\underline{\underline{C}}_l^T$, a call to the model l's solver, and a multiplication by $\underline{\underline{C}}_l$. This has also the advantage of allowing us to naturally use external solvers without being intrusive.

More details on the resolution of Equation (7) - mainly concerning the case where one of the models has a singular matrix - can be found in [6]. A full description of our implementation of the Arlequin / FETI solver, with a detailed analysis of the numerical effects on the CG algorithm and the treatment of the singular matrix cases, is presented in [10], which is under preparation. The resulting algorithm, together with other numerical optimizations, has a good weak scaling, as we can see from black curve in Figure 6, which presents the number of iterations of the Arlequin / FETI solver for a traction test, as a function of the number of elements of the mesh $\mathcal{T}_2$ (the other curves represent less optimized algorithms).

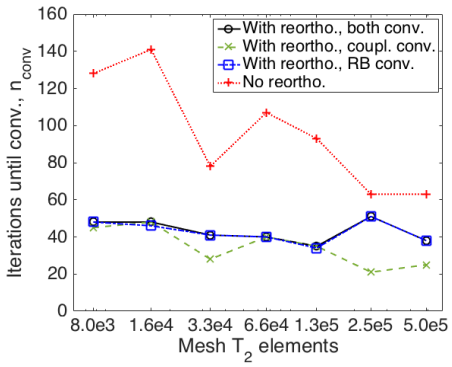


FIGURE 6 – Weak scaling for different implementations of the projected CG algorithm : number of iterations until convergence, for different numbers of elements for the mesh $\mathcal{T}_2$.

6 Application example

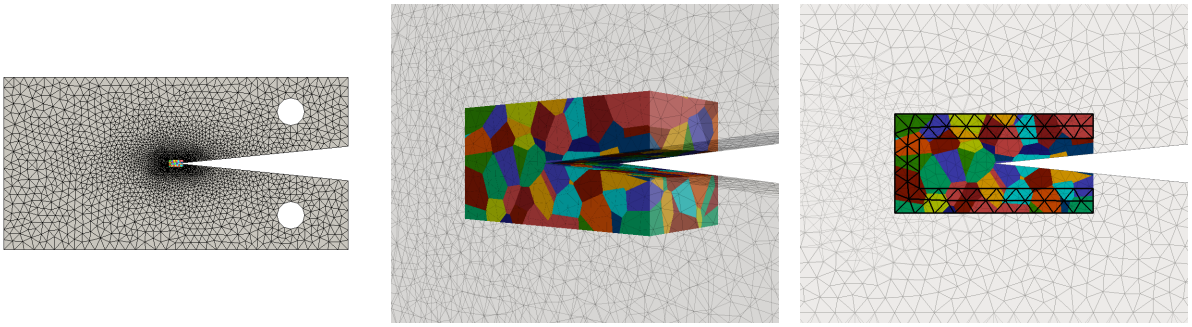
For an application test, we considered the coupled fissure test, with the meshes shown in Figure 7. The macroscopic system is described by an homogeneous 3D linear elasticity model, with a Young's modulus $E = 200$ GPa and a shear modulus $\mu = 80$ GPa. The corresponding domain Ω_1 is represented by the mesh $\mathcal{T}_1$ (Figure 7a), with $|\mathcal{T}_1| \sim 3.4 \cdot 10^4$ elements. Its leftmost side is clamped, while displacements on right side holes, on the $\vec{z}$ and $-\vec{z}$ directions. The microscopic system is described by an heterogeneous 3D anisotropic linear elasticity model, with physical parameters $c_{11} = 198$ GPa, $c_{12} = 125$ GPa, $c_{44} = 122$ GPa. Its domain Ω_2 (Figure 7b) is divided into 250 crystals, each with its own random anisotropy direction following an uniform direction distribution. The corresponding mesh has $|\mathcal{T}_2| \sim 7.2 \cdot 10^6$ elements. This model is located at the fissure junction of the macroscopic mesh $\mathcal{T}_1$, and both meshes are coupled by the region marked in Figure 7c, with a coupling mesh with $|\mathcal{T}_C| \sim 1.1 \cdot 10^3$ elements. Overall, the domains were meshed in such a way that the mesh $\mathcal{T}_1$'s element length is $\sim 10\times$ smaller near the fissure junction than on the outer region, while the mesh $\mathcal{T}_2$'s element length is $\sim 10\times$ smaller than the smallest element length of $\mathcal{T}_1$.

The results are presented in Figure 8, where we see that the displacements conditions on the macroscopic model result on a coherent deformation and von Mises stress in the microscopic model. The simulation was run on 96 processors, and converged after 41 iterations, with a wall time of 1073.90s .Most of this time (984.17s, or 91.6% of the time) was spent by the (several) calls to the solvers of the two different models, indicating that the other operations associated to the coupled solver are well optimized.

Références

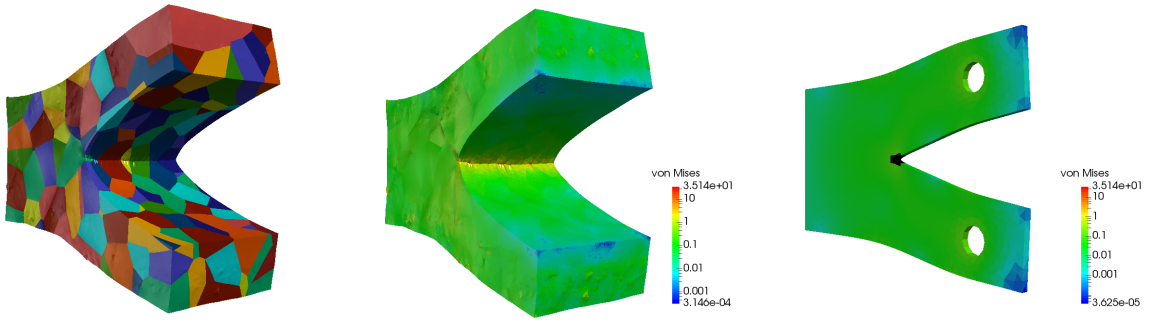
[1] R. Cottereau, *Numerical strategy for unbiased homogenization of random materials*, International Journal for Numerical Methods in Engineering, vol 95, issue 1, pp. 71–90, 6 July 2013.

[2] H.B. Dhia, N. Elkhodja, F.X. Roux, *Multimodeling of multi-alterated structures in the Arlequin framework*, European Journal of Computational Mechanics, vol. 27 number 5-7, pp. 969-980 , 2008.



(a) (b) (c)

FIGURE 7 – Meshes used for the fissure test : (a) macroscopic homogeneous model mesh, $\mathcal{T}_1$, with the microscopic, polycrystalline and anisotropic model mesh inset, $\mathcal{T}_2$. (b) zoom on the microscopic mesh. (c) coupling region mesh, $\mathcal{T}_C$ (black, thick lines). Number of elements in each mesh : $|\mathcal{T}_1| \sim 3.4 \cdot 10^4$ elements, $|\mathcal{T}_2| \sim 1.1 \cdot 10^6$ elements.



(a) (b) (c)

FIGURE 8 – Results of the fissure test : (a) deformed microscopic model mesh, $\mathcal{T}_2$, (b) von Mises stress (log scale) of the microscopic model, (c) deformed macroscopic model mesh, $\mathcal{T}_1$, and its von Mises stress (log scale).

[3] H.B. Dhia, G. Rateau, *The Arlequin method as a flexible engineering design tool*, International Journal for Numerical Methods in Engineering, vol 62, issue 11, pp. 1442–1462, 21 March 2005.

[4] V. Dolean, P. Jolivet, F. Nataf, *An Introduction to Domain Decomposition Methods*, Society for Industrial and Applied Mathematics, 2015.

[5] M.J. Gander, C. Japhet, *An Algorithm for Non-Matching Grid Projections with Linear Complexity*, Domain Decomposition Methods in Science and Engineering XVIII, no. 70, Lecture Notes in Computational Science and Engineering, Springer, pp. 185-192, 2009.

[6] C. Farhat, F.X. Roux, *A method of finite element tearing and interconnecting and its parallel solution algorithm*, International Journal for Numerical Methods in Engineering, vol. 32 number 6, Springer Netherlands, pp. 1205-1227 , 1991.

[7] M.G.D. Geers, V.G. Kouznetsova, W.A.M. Brekelmans, *Multi-scale computational homogenization : Trends and challenges*, Fourth International Conference on Advanced COmputational Methods in ENgineering (ACOMEN 2008), vol. 234, issue 7, pp. 2175–2182, 1 August 2010.

[8] D. Néron, H.B. Dhia, R. Cottereau, *A decoupled strategy to solve reduced-order multimodel problems in the PGD and Arlequin frameworks*, Computational Mechanics pp. 1-13, 2016.

[9] H. Samet, *Hierarchical spatial data structures*, Design and Implementation of Large Spatial Databases, no. 409, Lecture Notes in Computer Science, Springer, pp. 191-212, 2005.

[10] T.M. Schlittler, R. Cottereau, *Fully scalable implementation of a volume coupling scheme for the modeling of polycrystalline materials*, in preparation.

[11] T.I. Zohdi, P. Wriggers, *Computational micro-macro material testing*, Archives of Computational Methods in Engineering, vol. 8 number 2, Springer Netherlands, pp. 131-228 , 2001.