



HAL
open science

Une Toolbox Abaqus pour le calcul de propriétés effectives de milieux hétérogènes

Stéphane Lejeunes, Stéphane Bourgeois

► **To cite this version:**

Stéphane Lejeunes, Stéphane Bourgeois. Une Toolbox Abaqus pour le calcul de propriétés effectives de milieux hétérogènes. 10e colloque national en calcul des structures, May 2011, Giens, France. pp.Clé USB. hal-00592866

HAL Id: hal-00592866

<https://hal.science/hal-00592866>

Submitted on 3 May 2011

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.

Une Toolbox Abaqus pour le calcul de propriétés effectives de milieux hétérogènes

S. Lejeunes¹, S. Bourgeois²

¹ LMA, CNRS UPR7051, France, lejeunes@lma.cnrs-mrs.fr

² ECM, LMA CNRS UPR7051, France, stephane.bourgeois@centrale-marseille.fr

Résumé — Nous proposons une interface graphique, open-source, complètement intégrée à Abaqus qui permet de mettre en œuvre rapidement et facilement les problèmes de localisation associés aux méthodes classiques d'homogénéisation. Partant d'un maillage de la microstructure, l'ajout de degrés de liberté supplémentaires permet de piloter simplement le calcul à déformation ou contrainte moyenne imposée et d'obtenir sans post-traitement supplémentaire la réponse effective du VER. L'approche permet d'utiliser l'ensemble des lois disponibles dans Abaqus pour le comportement des constituants.

Mots clés — Homogénéisation, propriétés effectives, éléments finis.

1 Introduction

La mise en œuvre de problèmes d'homogénéisation numériques dans un code de calcul par éléments finis requiert l'écriture de conditions aux limites spécifiques sur le contour du volume élémentaire représentatif (VER) et l'imposition d'un chargement en moyenne. La mise en œuvre de ces problèmes peut s'avérer particulièrement lourde. La motivation principale de l'interface que nous proposons est d'automatiser et de simplifier au maximum cette mise en données dans un logiciel commercial assez largement utilisé (Abaqus). Dans sa version courante, l'interface développée offre la possibilité de mettre en œuvre les théories d'homogénéisation classiquement utilisées : déformation homogène au contour (KUBC¹), contrainte homogène au contour (SUBC²) et homogénéisation périodique (PBC³), en petites perturbations et grandes déformations (2D ou 3D).

La stratégie retenue et implémentée consiste à formuler les problèmes de localisation associés aux différentes méthodes en introduisant des ddls supplémentaires supportant les composantes des déformations macroscopiques (cf. [1]). Les conditions aux limites spécifiques aux différents problèmes (KUBC, SUBC ou PBC) sont alors imposées via des relations linéaires entre les degrés de liberté des nœuds du contour et ces ddls supplémentaires. On peut alors montrer que les forces nodales associées aux ddls supportant les composantes des déformations moyennes sont égales aux composantes de la contrainte moyenne (au volume du VER près). Le pilotage à déformation ou contrainte moyenne imposée revient alors à imposer la valeur des ddls supplémentaires ou les forces nodales associées. L'obtention de la réponse effective du VER ne nécessite donc aucun post-traitement particulier. Dans Abaqus, les relations linéaires entre ddls sont prises en compte par une méthode d'élimination. Dans la littérature, on peut trouver d'autres méthodes. Par exemple, on peut introduire les conditions aux limites par pénalisation [4, 6, 3]. Pour les problèmes périodiques, certains auteurs introduisent une discrétisation des champs fluctuants plutôt que des champs totaux [2, 5]. La méthode retenue ici a le mérite de pouvoir être implémentée simplement dans le logiciel Abaqus et de pouvoir profiter ainsi de l'ensemble des lois de comportement disponibles pour les constituants du VER.

Ce papier est organisé de la manière suivante : dans une première partie, nous rappelons brièvement les problèmes de localisation associés aux différentes méthodes d'homogénéisations qui sont gérés par l'interface, en petites déformations et en grandes transformations. Les conditions aux limites et les for-

-
1. Kinematically Uniform Boundary Conditions
 2. Statically Uniform Boundary Conditions
 3. Periodic Boundary Conditions

mulations faibles associées à ces problèmes sont détaillées pour justifier leur implémentation. Dans une seconde partie nous détaillons l'implémentation de l'interface et son utilisation. Enfin, dans la dernière partie, nous montrons quelques exemples permettant d'illustrer les capacités de cette "boîte à outils".

2 Problèmes de localisations sur le Volume Élémentaire représentatif

2.1 Petites déformations

On considère un Volume Élémentaire Représentatif (VER), assimilable à une microstructure hétérogène occupant un domaine V de frontière ∂V . La loi de comportement caractérisant le matériau en un point \mathbf{x} du volume, reliant le tenseur des contraintes $\boldsymbol{\sigma}(\mathbf{x})$ et le tenseur des déformations $\boldsymbol{\varepsilon}(\mathbf{x})$, n'est pas précisée. On écarte cependant la présence de pores et de constituants rigides, même si ces cas peuvent être traités sous certaines conditions par l'interface. Le comportement effectif est caractérisé classiquement par la relation entre les déformations et les contraintes moyennes définies par :

$$\bar{\boldsymbol{\varepsilon}} = \langle \boldsymbol{\varepsilon} \rangle = \frac{1}{|V|} \int_V \boldsymbol{\varepsilon} \, dV, \quad \bar{\boldsymbol{\sigma}} = \langle \boldsymbol{\sigma} \rangle = \frac{1}{|V|} \int_V \boldsymbol{\sigma} \, dV \quad (1)$$

où $|V|$ désigne le volume du VER. Supposant que le champ de déformation est compatible avec un champ de déplacement $\mathbf{u}(\mathbf{x})$ continu, on peut utiliser la définition équivalente suivante pour la moyenne des déformations

$$\bar{\boldsymbol{\varepsilon}} = \frac{1}{|V|} \int_{\partial V} \mathbf{u} \otimes_S \mathbf{n} \, dV \quad (2)$$

où \otimes_S désigne le produit tensoriel symétrisé et $\mathbf{n}(\mathbf{x})$ le vecteur normal sortant. Les problèmes de localisation peuvent être menés à contrainte moyenne imposée $\bar{\boldsymbol{\sigma}}^d$ ou déformation moyenne imposée $\bar{\boldsymbol{\varepsilon}}^d$. Le choix de la méthode de localisation revient ensuite à faire un choix de conditions limites. On envisage dans la suite trois méthodes : déformation homogène au contour (KUBC), contrainte homogène au contour (SUBC) et conditions périodiques. Omettant la loi de comportement, on peut résumer le problème (KUBC) de la façon suivante :

$$P_{\text{KUBC}} : \operatorname{div} \boldsymbol{\sigma}(\mathbf{x}) = 0 \text{ dans } V, \quad \mathbf{u}(\mathbf{x}) - \bar{\boldsymbol{\varepsilon}} \cdot \mathbf{x} = 0 \text{ sur } \partial V, \quad \bar{\boldsymbol{\varepsilon}} = \bar{\boldsymbol{\varepsilon}}^d \text{ ou } \bar{\boldsymbol{\sigma}} = \bar{\boldsymbol{\sigma}}^d. \quad (3)$$

La résolution du problème P_{KUBC} à déformation imposée dans un code de calcul par éléments finis classique peut généralement être mise en œuvre assez facilement. Il nécessite simplement l'introduction de conditions limites dépendant des coordonnées des noeuds du contour. Il est cependant souvent nécessaire de développer un post-traitement spécifique pour calculer la moyenne des contraintes. On remarquera aussi que le pilotage à contrainte moyenne imposée est plus délicat. Un moyen de contourner ces difficultés techniques liées au chargement en moyenne est d'introduire les composantes de la déformation moyenne comme des degrés de libertés supplémentaires du problème. La condition limite $\mathbf{u}(\mathbf{x}) - \bar{\boldsymbol{\varepsilon}} \cdot \mathbf{x} = 0$ sur ∂V se traduit alors par des relations linéaires entre ddls. La formulation variationnelle associée au problème P_{KUBC} devient alors, en tenant compte des relations linéaires introduites :

$$P_{\text{KUBC}} : \text{pour tout } \mathbf{u}^*(\mathbf{x}) \text{ et } \bar{\boldsymbol{\varepsilon}}^* \text{ vérifiant } \mathbf{u}^* - \bar{\boldsymbol{\varepsilon}}^* \cdot \mathbf{x} = 0 \text{ sur } \partial V, \quad \int_V \boldsymbol{\sigma} : \boldsymbol{\varepsilon}(\mathbf{u}^*) \, dV = |V| \bar{\boldsymbol{\sigma}} : \bar{\boldsymbol{\varepsilon}}^*. \quad (4)$$

La formulation (4) montre que les forces nodales associées aux nouveaux ddls sont égales aux contraintes moyennes multipliées par le volume du VER. Avec cette formulation, le pilotage à déformation moyenne (resp. à contrainte moyenne) s'effectue en imposant simplement la valeur des ddls supplémentaires (resp. la valeur des forces nodales aux ddls supplémentaires). On remarquera que la formulation (4) ne dépend pas de la loi de comportement des constituant du VER.

Considérons maintenant le problème à contrainte homogène au contour :

$$P_{\text{SUBC}} : \operatorname{div} \boldsymbol{\sigma}(\mathbf{x}) = 0 \text{ dans } V, \quad \boldsymbol{\sigma}(\mathbf{x}) \cdot \mathbf{n} - \bar{\boldsymbol{\sigma}} \cdot \mathbf{n} = 0 \text{ sur } \partial V, \quad \bar{\boldsymbol{\varepsilon}} = \bar{\boldsymbol{\varepsilon}}^d \text{ ou } \bar{\boldsymbol{\sigma}} = \bar{\boldsymbol{\sigma}}^d. \quad (5)$$

Pour le problème P_{SUBC} , il est naturel de mener les calculs à contrainte moyenne imposée et de calculer en post-traitement la déformation moyenne. Le pilotage à déformation moyenne imposée est délicat.

Une approche analogue à celle exposée pour traiter le problème P_{KUBC} peut être adoptée. Elle consiste de même à introduire des dds supplémentaires supportant les composantes de la déformation moyenne. Les relations linéaires introduites correspondent à la version discrétisée de l'équation (2), permettant de définir la déformation moyenne à partir des déplacements des noeuds du contour. La formulation variationnelle associée au problème P_{KUBC} s'écrit alors :

$$P_{SUBC} : \text{pour tout } \mathbf{u}^*(\mathbf{x}) \text{ et } \bar{\boldsymbol{\varepsilon}}^* \text{ vérifiant } \frac{1}{|V|} \int_{\partial V} \mathbf{u}^* \otimes_S \mathbf{n} dS - \bar{\boldsymbol{\varepsilon}}^* = 0, \int_V \boldsymbol{\sigma} : \boldsymbol{\varepsilon}(\mathbf{u}^*) dV = |V| \bar{\boldsymbol{\sigma}} : \bar{\boldsymbol{\varepsilon}}^*. \quad (6)$$

Cette formulation variationnelle est équivalente aux deux premières équations de (5). L'introduction des relations linéaires associées à l'équation (2) permet ainsi d'imposer par dualité les conditions de contrainte homogène au contour.

Pour les milieux périodiques, on choisit classiquement de mener les calculs sur une cellule de base qui génère le milieu par périodicité. Cette périodicité est définie par la donnée de vecteurs permettant de construire le milieu par translation de la cellule de base le long de ceux-ci. Le problème associé s'écrit (toujours en omettant de préciser la loi de comportement des constituants) :

$$P_{PBC} : \text{div } \boldsymbol{\sigma}(\mathbf{x}) = 0 \text{ dans } V, \boldsymbol{\sigma}(\mathbf{x}) \cdot \mathbf{n} - \# \text{ sur } \partial V, \mathbf{u}(\mathbf{x}) - \bar{\boldsymbol{\varepsilon}} \cdot \mathbf{x} \# \text{ sur } \partial V, \bar{\boldsymbol{\varepsilon}} = \bar{\boldsymbol{\varepsilon}}^d \text{ ou } \bar{\boldsymbol{\sigma}} = \bar{\boldsymbol{\sigma}}^d, \quad (7)$$

où les symboles « # » (resp. « -# ») caractérisent des champs qui prennent des valeurs égales (resp. opposées) en deux points qui se correspondent par translation suivant les vecteurs de périodicités sur des parties opposées du contour. La même démarche conduit à introduire des relations linéaires entre des dds additionnels supportant les composantes de la déformation moyenne et les déplacements des noeuds du bords, traduisant la condition $\mathbf{u}(\mathbf{x}) - \bar{\boldsymbol{\varepsilon}} \cdot \mathbf{x} \#$ sur ∂V . La formulation variationnelle associée

$$P_{PBC} : \text{pour tout } \mathbf{u}^*(\mathbf{x}) \text{ et } \bar{\boldsymbol{\varepsilon}}^* \text{ vérifiant } \mathbf{u}^* - \bar{\boldsymbol{\varepsilon}}^* \cdot \mathbf{x} \# \text{ sur } \partial V, \int_V \boldsymbol{\sigma} : \boldsymbol{\varepsilon}(\mathbf{u}^*) dV = |V| \bar{\boldsymbol{\sigma}} : \bar{\boldsymbol{\varepsilon}}^* \quad (8)$$

permet notamment d'assurer par dualité la condition $\boldsymbol{\sigma}(\mathbf{x}) \cdot \mathbf{n} - \#$ sur ∂V .

2.2 Grandes déformations

Contrairement au cas des petites perturbations, en grandes déformations, le choix de la mesure des déformations et des contraintes se pose. Dans la littérature, le comportement effectif est souvent défini à l'aide des moyennes, sur le VER, du tenseur gradient de la transformation \mathbf{f} du premier tenseur de Piola-Kirchoff $\boldsymbol{\pi}$ (cf [4, 6, 3]) :

$$\bar{\mathbf{f}} = \langle \mathbf{f} \rangle = \frac{1}{|V_0|} \int_{V_0} \mathbf{f} dV, \quad \bar{\boldsymbol{\pi}} = \langle \boldsymbol{\pi} \rangle = \frac{1}{|V_0|} \int_{V_0} \boldsymbol{\pi} dV, \quad (9)$$

où V_0 désigne le domaine occupé par le VER dans la configuration initiale de référence et $|V_0|$ son volume. On note \mathbf{x}_0 le vecteur position initiale d'un point matériel dans cette configuration. Introduisant le champ de déplacement $\mathbf{u}(\mathbf{x}_0)$ associé au tenseur gradient de la transformation, on peut utiliser la définition équivalente suivante :

$$\bar{\mathbf{f}} = \mathbf{g} + \frac{1}{|V_0|} \int_{\partial V_0} \mathbf{u} \otimes \mathbf{n}_0 dS, \quad (10)$$

avec \mathbf{g} le tenseur identité et \mathbf{n}_0 la normale extérieure au domaine occupé par le VER dans sa configuration initiale. Les trois méthodes de localisation classiquement utilisées (conditions homogènes en déformations et en contraintes, conditions périodiques) peuvent être utilisées dans ce cadre. Le tableau 1 précisent les conditions limites associées à chacune des méthodes et les relations linéaires à mettre en œuvre après introduction des dds supplémentaires supportant les composantes du gradient de transformation moyen.

2.3 Une formulation unifiée pour la mise en œuvre dans Abaqus

La méthode proposée pour la mise en œuvre des problèmes de localisations, que ce soit en petites perturbations ou en grandes déformations, et quelque soit la méthode (KUBC, SUBC, PBC), repose sur

Problèmes	Conditions limites	Relations linéaires
P_{KUBC}	$\mathbf{u}(\mathbf{x}_0) - (\bar{\mathbf{f}} - \mathbf{g}) \cdot \mathbf{x}_0 = 0$ sur ∂V_0	$\mathbf{u}(\mathbf{x}_0) - (\bar{\mathbf{f}} - \mathbf{g}) \cdot \mathbf{x}_0 = 0$ sur ∂V_0
P_{SUBC}	$\boldsymbol{\pi}(\mathbf{x}_0) \cdot \mathbf{n}_0 = \bar{\boldsymbol{\pi}} \cdot \mathbf{n}_0$ sur ∂V	$\bar{\mathbf{f}} - \mathbf{g} - \frac{1}{ V_0 } \int_{\partial V_0} \mathbf{u} \otimes \mathbf{n}_0 dS = 0$
P_{PBC}	$\boldsymbol{\pi}(\mathbf{x}_0) \cdot \mathbf{n}_0 - \#$ et $\mathbf{u}(\mathbf{x}_0) - (\bar{\mathbf{f}} - \mathbf{g}) \cdot \mathbf{x}_0 \#$ sur ∂V_0	$\mathbf{u}(\mathbf{x}_0) - (\bar{\mathbf{f}} - \mathbf{g}) \cdot \mathbf{x}_0 \#$ sur ∂V_0

TABLE 1 – Traitements des conditions limites pour les problèmes de localisation en grandes déformations

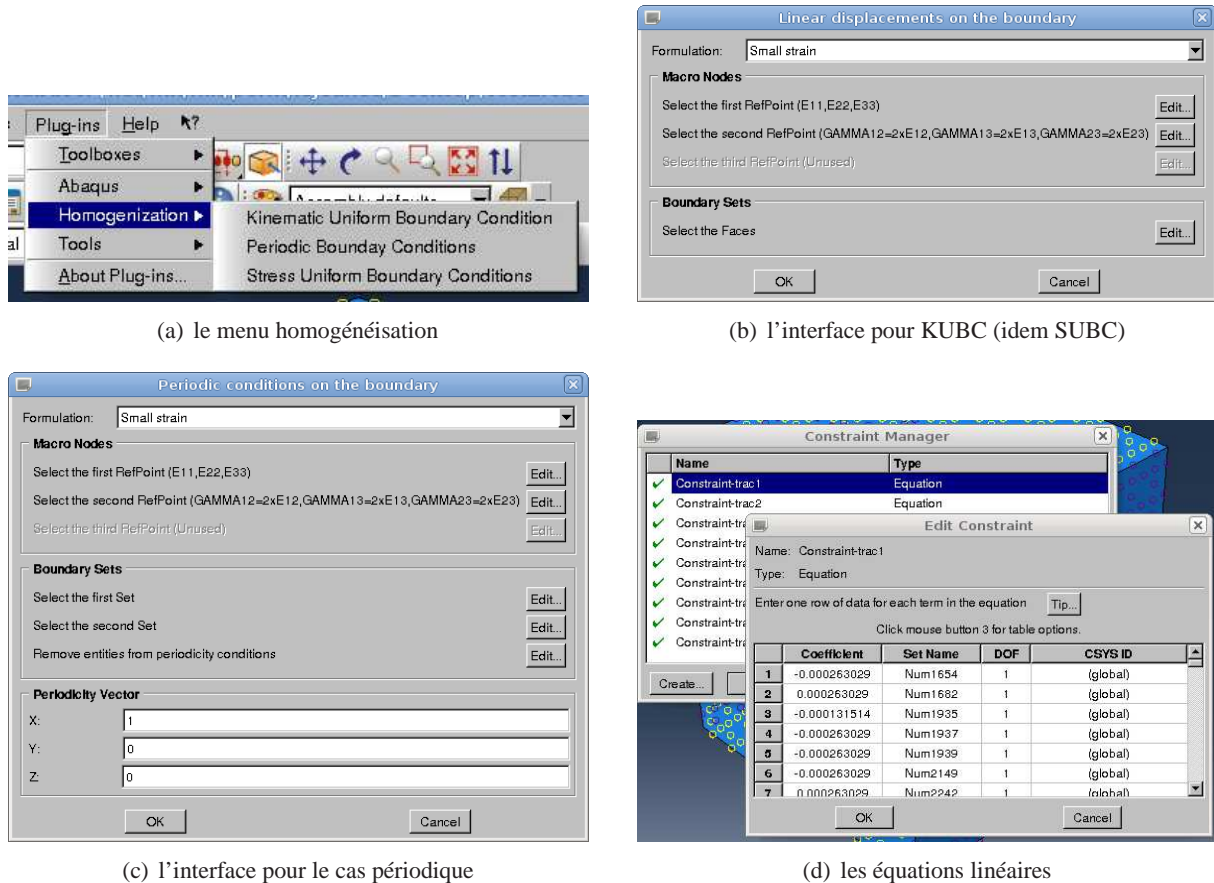


FIGURE 1 – Le module homogénéisation

l'introduction de ddls supplémentaires et de relations linéaires entre degrés de liberté. Dans le logiciel Abaqus, les ddls supplémentaires peuvent être introduits en créant dans l'interface CAE des points de référence (Reference Point RP). La création des relations linéaires entre ddls, spécifiques à chaque cas, est réalisée via une interface conviviale intégrée au menu de l'interface CAE.

3 Implémentation et mode d'emploi

La "boîte à outils" proposée est écrite en Python et est constituée de trois classes (trois interfaces) : KUBC, SUBC et PBC. Parallèlement à cette partie graphique qui est très simple, nous avons également trois moteurs (ou classes) de génération de conditions aux limites qui sont chargés de construire les relations linéaires propres à chaque méthode et qui peuvent être appelés par les différentes interfaces ou directement par des scripts Python⁴. Concrètement, pour l'utilisateur, ces interfaces sont disponibles dans le module Interaction d'Abaqus, depuis le menu déroulant Plug-ins (voir fig. 1(a)).

4. L'utilisation direct des moteurs de génération peut permettre d'automatiser une mise en données du modèle, pour faire de l'optimisation par exemple.

3.1 Utilisation

Tout d'abord, toutes les interfaces pré-supposent qu'un maillage du VER ait déjà été réalisé et que les points de références soient déjà créés. Au démarrage de chacune des interfaces, l'utilisateur doit préciser, via un menu déroulant, dans quel cadre il souhaite travailler : petites perturbations ou grandes déformations. Ceci permet d'être informé d'une part du nombre de points de références nécessaires et d'autre part de la signification de chaque ddl de ces points :

- en 2D HPP (déformations planes) : RP 1 $\{\bar{\epsilon}_{11}, \bar{\epsilon}_{22}\}$, RP 2 $\{2\bar{\epsilon}_{12}, \text{non utilisé}\}$,
- en 2D GD (déformations planes) : RP 1 $\{\bar{f}_{11} - 1, \bar{f}_{12}\}$, RP 2 $\{\bar{f}_{22} - 1, \bar{f}_{21}\}$,
- en 3D HPP : RP 1 $\{\bar{\epsilon}_{11}, \bar{\epsilon}_{22}, \bar{\epsilon}_{33}\}$, RP 2 $\{2\bar{\epsilon}_{12}, 2\bar{\epsilon}_{13}, 2\bar{\epsilon}_{23}\}$,
- en 3D GD : RP 1 $\{\bar{f}_{11} - 1, \bar{f}_{12}, \bar{f}_{13}\}$, RP 2 $\{\bar{f}_{21}, \bar{f}_{22} - 1, \bar{f}_{23}\}$, RP 3 $\{\bar{f}_{31}, \bar{f}_{32}, \bar{f}_{33} - 1\}$.

Dans le cas SUBC et KUBC (voir 1(b)), l'utilisateur doit fournir les entrées suivantes :

- les points de références (la sélection se fait à la souris),
- les faces, arrêtes ou nœuds formant le contour du VER (toujours à la souris).

Dans le cas périodique (voir 1(c)), l'interface doit être appelée pour chaque paire géométrique se correspondant par périodicité (soit au minimum 2 fois en 2D et 3 fois en 3D). On doit fournir :

- les points de références (la sélection se fait à la souris),
- les faces, arrêtes ou nœuds se correspondant par périodicité (toujours à la souris),
- le vecteur de périodicité (au clavier).

Après validation de l'interface, les relations linéaires générées se retrouvent directement dans le "Constraint Manager" d'Abaqus (voir fig. 1(d)). Le tableau 2 résume les points forts et les points faibles de l'interface :

Points faibles	Points forts
<p>"<i>maillage dépendant</i>" : à chaque modification du maillage il faut régénérer les conditions en relançant l'interface</p> <p>"<i>cas périodique</i>" : le maillage doit être périodique</p>	<p>"<i>sélection par nœud</i>" : on peut définir les conditions avec un import de modèle (depuis un .odb)</p> <p>"<i>séparation GUI/Kernel</i>" : on peut utiliser les moteurs (kernel) indépendamment de l'interface, par exemple dérivation de nouvelles classes thermoélasticité ...</p> <p>"<i>post-traitement et pilotage</i>" : les points de référence permettent à la fois de piloter en contrainte ou déformation moyenne et de déduire directement la réponse effective</p>

TABLE 2 – Points forts/point faible de l'interface

3.2 Moteurs de génération des conditions

Les trois moteurs (classes) de génération sont chargés de créer dans la base de données du modèle toutes les relations et les groupes de nœuds nécessaires. Le nombre de ligne de chaque moteur est un bon indicateur de leur complexité en terme de programmation : KUBC 132 lignes, SUBC 387 lignes, PBC 460 lignes. Le tableau 3 résume les opérations effectuées par chaque moteur.

Le typage dynamique et les conteneurs spécifiques de python permettent au final d'avoir un code succinct et lisible. Par exemple, le cœur du code de la classe `PeriodicBoundary` reste très simple puisqu'il contient les lignes suivantes :

```
# check if we have a mesh
if(len(a.allSets[nameS1].nodes)==0):
```

```

print "A mesh is needed!"
self.SuppressPickedSet(a)
return 0
# check the mesh periodicity
if(not self.VerifNodeCoord(nameS1,nameS2,a)):
    self.SuppressPickedSet(a)
    return 0
self.MakeNodeSetsandEquations(nameS1,nameS2,nameS3,a,dim)

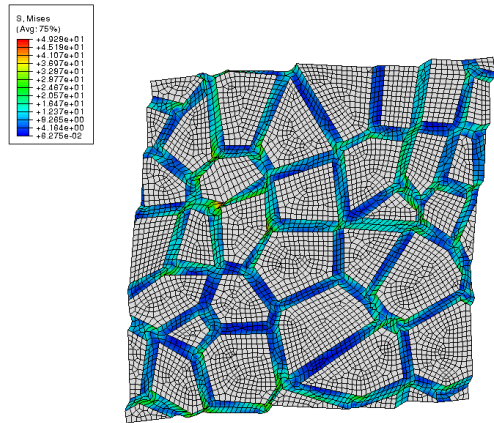
```

4 Exemples d'applications

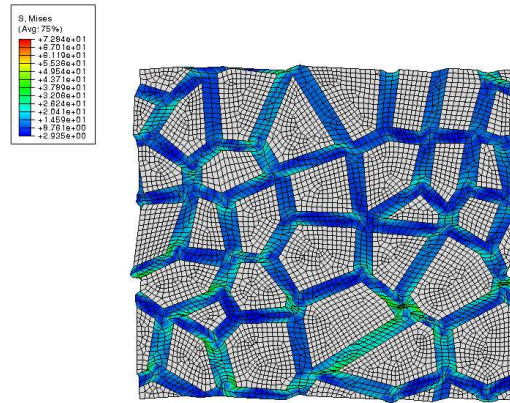
Le premier exemple concerne une cellule de base, extraite d'un milieu périodique, constituée de deux phases : des grains rigides décrits par des polygones de voronoï et une matrice souple, élastique, ayant les caractéristiques suivantes : $E = 4\text{MPa}$, $\nu = 0.489$. Il s'agit d'un cas 2D en déformations planes et petites perturbations. Les grains sont modélisés en tant que corps rigides : leur mouvement est entièrement décrit par des points de références (un pour chaque grain) qui portent les degrés de libertés de translation et de rotation de chaque grain. Le maillage est périodique. Le cas particulier des portions de grains se correspondant par périodicité est géré en affectant un point de référence à chaque portion. Ces points de références se correspondent par périodicité. L'interface gère ce cas particulier en imposant en plus des relations de périodicité classiques des conditions d'égalités de rotation.

KUBC	SUBC	PBC
<ol style="list-style-type: none"> 1. vérifier la présence d'un maillage , 2. calculer la distance $x_0^i - x_0^M$, où x_0^i représente les coordonnées du nœud i dans la configuration de référence et x_0^M les coordonnées du premier point de référence, pris comme origine des abscisses par convention, 3. écrire les relations linéaires. 	<ol style="list-style-type: none"> 1. vérifier la présence d'un maillage , 2. sur chaque arrête (cas 2D) ou chaque face (cas 3D), calculer la contribution de l'intégrale de frontière $\int_{\partial\Omega_e} \mathbf{u}^i \otimes \mathbf{n}_0 dS$ de l'élément Ω_e au nœud i, en tenant compte du type d'élément utilisé, 3. assembler les contributions de chaque élément aux noeuds du contour, 4. simplifier le système d'équation par une élimination de type pivot, 5. écrire les relations linéaires. 	<ol style="list-style-type: none"> 1. vérifier la présence et la périodicité du maillage, 2. trouver les nœuds se correspondant par périodicité, 3. construire un arbre de relation sans cycles (au sens de la théorie des graphes) : l'objet dictionary de python est à la base d'une classe Tree incluse dans la boîte à outil, 4. vérifier et gérer la non répétition de nœuds maîtres dans l'ensemble des équations (si nécessaire : permutations maîtres/esclaves), 5. écrire les relations linéaires.

TABLE 3 – Tâches de chaque moteur

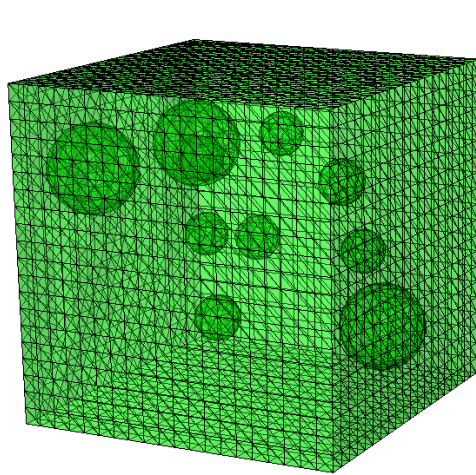


(a) Cisaillement

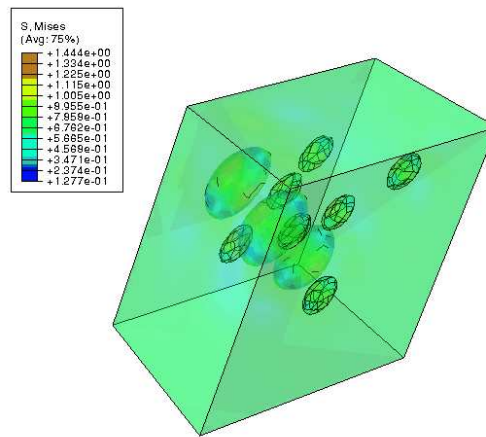


(b) Extension (suivant x)

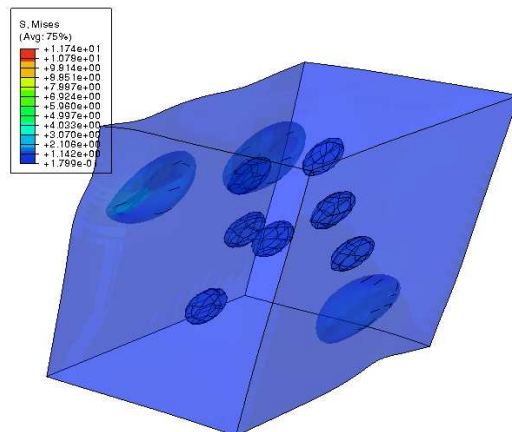
FIGURE 2 – Cellule type Voronoi en déformation plane, condition périodique



(a) Maillage



(b) Cisaillement (KUBC)



(c) Cisaillement (SUBC)

FIGURE 3 – Cube avec cavités sphériques (cas grandes def)

Les figures 2, présentent le résultat d'un test de glissement simple (seul le terme $\bar{\epsilon}_{12}$ est non nul) et d'extension (seul le terme $\bar{\epsilon}_{11}$ est non nul). On obtient le tenseur d'élasticité effectif en réalisant 3 calculs élémentaires : deux extensions pures et un glissement simple. En post-traitant directement les réactions obtenues aux points de références, on obtient (valeurs en MPa) :

$$\begin{bmatrix} \bar{\sigma}_{11} \\ \bar{\sigma}_{22} \\ \bar{\sigma}_{12} \end{bmatrix} = \begin{bmatrix} 196.45 & 165.14 & 1.15 \\ 165.14 & 197.78 & 0.02 \\ 1.15 & 0.02 & 13.28 \end{bmatrix} \begin{bmatrix} \bar{\epsilon}_{11} \\ \bar{\epsilon}_{22} \\ 2\bar{\epsilon}_{12} \end{bmatrix}$$

Le deuxième exemple consiste en un cube comportant des cavités sphériques de différents rayons et réparties de manière aléatoire. Le comportement du matériau constitutif du cube est décrit par une loi hyperélastique, faiblement compressible, de type Mooney-Rivlin :

$$\Psi(I_1, I_2) = C_1(I_1 - 3) + C_2(I_2 - 3) + \frac{k}{2}(J - 1)$$

avec $C_1 = 0.256$ MPa, $C_2 = 0.12$ MPa, $k = 2000$ MPa. Il s'agit donc d'un cas grandes déformations qui est traité par des conditions de type KUBC ou SUBC. Le maillage est constitué d'éléments tétraèdre et comprend 90330 nœuds (voir fig. 3(a)).

Les figures 3 montrent les résultats obtenus pour 50% de déformation de cisaillement. Ce cas, très simple à mettre en oeuvre, illustre néanmoins la robustesse de l'outil car l'utilisateur n'a pas à se soucier de détails techniques même en 3D et en grandes déformations.

5 Conclusions

Nous avons développé une "boite à outil", open source, intégrée à Abaqus permettant de mettre en oeuvre très simplement un problème d'homogénéisation numérique avec des conditions de bord de type KUBC, SUBC ou périodique. Les conditions aux limites sont traitées par l'ajout de ddls supplémentaires qui supportent la déformation moyenne. L'opération de localisation s'écrit alors sous la forme de relations linéaires entre ces ddls et les déplacements des nœuds du bords. Cette formulation permet de piloter le calcul à déformation ou contrainte moyenne imposée, comme on impose la valeur d'un degré de liberté ou une force en un nœud. De plus, cette stratégie est valable en petites perturbations ou en grandes déformations et est indépendante de la loi de comportement des constituants. La méthode adoptée peut être étendue à d'autres types de physique (conduction...) et à l'homogénéisation de structure (poutres et plaques périodiques), dès qu'on a des relations de moyennes et de périodicité pouvant être traduites par des relations linéaires entre ddls.

Références

- [1] O. Débordes, *Homogénéisation périodique*, UNIT, 2001
- [2] J. Fish, R. Fan, *Mathematical homogenization of nonperiodic heterogeneous media subjected to large deformation transient loading*, International Journal for Numerical Methods in Engineering, 1044-1064, 2008
- [3] J.C. Michel, H. Moulinec et P. Suquet, *Composites à microstructure périodique*, Homogénéisation en mécanique des matériaux, M. Bornert, T. Bretheau et P. Gilormini (Ed.), Hermes, Paris, 57-94, 2001.
- [4] C. Miehe, C.G. Bayreuther, *Multilevel FEM for Heterogeneous Structures : From Homogenization to Multigrid Solvers*, Lecture Notes in Applied and Computational Mechanics, 361-397, 2006
- [5] C. Miehe, J. Schröder, C.G. Bayreuther, *On the homogenization analysis of composite materials based on discretized fluctuations on the micro-structure*, Acta Mechanica, 1-16, 2002
- [6] I. Temizer, P. Wriggers, *On the computation of the macroscopic tangent for multiscale volumetric homogenization problems*, Computers Methods in Applied Mechanics and Engineering, 495-510, 2008