

Extension des solveurs FFT aux conditions aux limites non périodiques et aux méthodes multi-grilles locales

N. B. Nkoubou Kaptchouang¹, L. Gélébart¹

¹ DES/ISAS/DMN/SRMA, CEA Paris-Saclay, {noe-brice.nkouboukaptchouang,lionel.gelebart}@cea.fr

Résumé — Il est proposé dans cette communication de mettre en place un algorithme multi-échelles adapté aux solveurs mécaniques de type FFT (Fast Fourier Transform). L'approche adoptée s'appuie sur les algorithmes multi-grilles locaux notamment la méthode LDC (Local Defect Correction). Une application directe de ce type d'algorithme discutée dans l'article est l'introduction de raffinements locaux dans des calculs FFT. L'algorithme de raffinement local nécessite une modification des solveurs FFT standards pour prendre en compte des Conditions aux Limites de type Dirichlet et de type Neumann.

Mots clés — solveur FFT, multi-échelle, raffinement local, Conditions aux Limites Dirichlet et Neumann, Local Defect Correction.

1 Introduction

Très performants, les « solveurs FFT » sont de plus en plus utilisés pour simuler le comportement mécanique de matériaux hétérogènes. Ces solveurs prennent en compte un volume élémentaire parallélépipédique discrétisé de manière régulière sous la forme d'une image 3D. La résolution repose sur l'utilisation d'un opérateur de Green discret évalué dans l'espace de Fourier sous l'hypothèse de Conditions aux Limites (CL) périodiques. Hormis ces deux conditions restrictives (discrétisation régulière et conditions aux limites périodiques), la méthode est très générale et s'applique pour des comportements linéaires ou non-linéaires, sous l'hypothèse des petites perturbations ou non (i.e. grandes transformations). Cette méthode est également très adaptée pour une implémentation massivement parallèle [2]. La nécessité d'un raffinement constant imposé par l'utilisation d'un solveur FFT périodique peut conduire à des simulations très lourdes dans le cas de simulations avec des tailles d'hétérogénéités très différentes. En effet, la finesse du raffinement est alors imposée par la plus petite hétérogénéité présente dans la cellule. Afin de contourner cette contrainte, on se propose de mettre en place une méthode multi-grilles locale dédiée aux « solveurs FFT ». On considère ici une discrétisation grossière du domaine complet (problème global) et une grille fine décrivant un sous-volume contenant des hétérogénéités de petite taille (problème local). En s'appuyant sur une modification d'un solveur FFT standard [3] pour imposer des Conditions aux Limites (CL) en déplacements (CL de types Dirichlet) ou en forces (CL type Neumann), un couplage est établi entre la résolution du problème global et la résolution du problème locale : les CL en forces ou en déplacements sur la grille locale sont imposées par prolongation de la solution globale grossière, puis la solution globale grossière est améliorée par la prise en compte d'une correction, le résidu local [4], évalué à partir d'une restriction de la solution du problème local fin.

2 Solveur FFT : rappel de l'algorithme de base

On considère le problème classique de la détermination des champs locaux $\underline{\underline{\sigma}}(\underline{x})$, $\underline{\underline{\epsilon}}(\underline{x})$ dans une cellule Ω composée d'un matériau hétérogène, décrite par le tenseur d'élasticité linéaire local $\mathbb{C}(\underline{x})$, soumise à une déformation macroscopique $\underline{\underline{E}}$ et à des CL périodiques. Les équations synthétisant ce problème sont :

$$\begin{cases} \operatorname{div} \underline{\underline{\sigma}} = \underline{\underline{0}} & \text{dans } \Omega \\ \underline{\underline{\sigma}} = \mathbb{C} : \underline{\underline{\varepsilon}}(\underline{\underline{u}}) & \text{dans } \Omega \\ \underline{\underline{\varepsilon}}(\underline{\underline{u}}) = \frac{1}{2} (\nabla \underline{\underline{u}} + {}^T \nabla \underline{\underline{u}}) & \text{dans } \Omega \\ \underline{\underline{u}} = \underline{\underline{E}} \cdot \underline{\underline{x}} + \underline{\underline{\tilde{u}}} & \text{dans } \Omega \\ \underline{\underline{\tilde{u}}} \text{ périodique} & \text{et } \underline{\underline{\sigma}} \cdot \underline{\underline{n}} \text{ antipériodique sur les faces opposées de } \partial \Omega \end{cases} \quad (1)$$

La méthode proposée par Moulinec et Suquet [5, 6] pour résoudre ce problème consiste à :

1. Reformuler le problème (1) en une équation intégrale de Lippmann et Schwinger en introduisant un milieu élastique de référence de raideur $\mathbb{C}^0$:

$$(1) \iff \underline{\underline{\varepsilon}}(\underline{\underline{u}}) = -\Gamma^0 * \underline{\underline{\tau}} + \underline{\underline{E}} \quad \left(\text{avec } \underline{\underline{\tau}} = (\mathbb{C} - \mathbb{C}^0) : \underline{\underline{\varepsilon}}(\underline{\underline{u}}) \right) \quad (2)$$

où $*$ est le produit de convolution et Γ^0 est l'opérateur de Green périodique associé au tenseur d'élasticité $\mathbb{C}^0$ qui peut être calculé explicitement dans l'espace de Fourier.

2. Utiliser un algorithme de point-fixe pour résoudre l'équation (2) :

$$\begin{cases} \underline{\underline{\tau}}^j = (\mathbb{C} - \mathbb{C}^0) : \underline{\underline{\varepsilon}}^j(\underline{\underline{u}}) & \text{(évalué dans l'espace Réel)} \\ \underline{\underline{\varepsilon}}^{j+1}(\underline{\underline{u}}) = -\Gamma^0 * \underline{\underline{\tau}}^j + \underline{\underline{E}} & \text{(évalué dans l'espace de Fourier)} \end{cases} \quad (3)$$

Cette méthode s'étend également aux comportements non linéaires en petites ou grandes transformations, avec des chargements macroscopiques en contrainte imposée ou des chargements mixtes. Dans l'approche de Moulinec et Suquet [5, 6] l'expression discrète de la divergence ou encore de l'équation de compatibilité intervenant dans l'opérateur de Green (3) est faite en prenant l'équivalent discret des dérivées partielles dans l'espace de Fourier (différentiation pseudo-spectrale). D'autres approches de discrétisation des dérivées partielles basées sur les différences finis ont été proposées dans la littérature (une revue récente est proposée dans [8]).

Dans la suite, l'opérateur de Green discret retenu résulte de l'utilisation de la différence finie proposée par [10] qui est équivalente à l'utilisation d'éléments finis linéaires à intégration réduite [9]. Dans ce schéma, les déplacements sont définis aux nœuds des voxels, les déformations et les contraintes sont évaluées aux centres des voxels par différence finie.

3 Stratégie de raffinement local avec un solveur FFT : méthode LDC

Les méthodes multi-grilles locales consistent, à partir d'un problème grossier défini sur l'intégralité du domaine, à définir de manière récursive des grilles locales emboîtées avec des pas de discrétisations de plus en plus fins. Ces méthodes sont compatibles avec les solveurs FFT dans la mesure où les discrétisations sur chaque grille peuvent être régulières. Il existe plusieurs méthodes multi-grilles locales permettant de coupler les résolutions sur les différentes grilles. Dans cet article nous nous limitons à un processus à 2 grilles (une grille globale et une grille locale) et nous nous intéressons à la méthode Local Defect Correction (LDC) [4].

3.1 La méthode LDC (Local Defect Correction)

On considère une discrétisation grossière du problème (1) avec un pas de discrétisation H :

$$\begin{cases} \operatorname{div} (\mathbb{C} : \underline{\underline{\varepsilon}}(\underline{\underline{u}}_H)) = \underline{\underline{0}} & \text{dans } \Omega_H \\ \text{CL périodiques} \end{cases} \quad (4)$$

Les champs solutions $\underline{\underline{\sigma}}_H$, $\underline{\underline{\varepsilon}}_H$, $\underline{\underline{u}}_H$ peuvent être obtenus par résolution du problème (4) notamment avec un solveur FFT standard (en cas de CL périodiques). On considère un sous-domaine $\Omega^l \subset \Omega$ où une solution plus précise est recherchée. Une discrétisation Ω_h^l de Ω^l avec un pas fin h est obtenue en subdivisant le maillage grossier Ω_H^l (voir Figure 1). Le niveau de raffinement de Ω_h^l par rapport à Ω_H^l est donné par $r = H/h$. L'interface entre Ω et Ω^l est noté I . Pour formuler le problème sur la grille locale Ω_h^l on définit des CL sur I_h par prolongement des valeurs de la solution grossière à l'interface. On peut

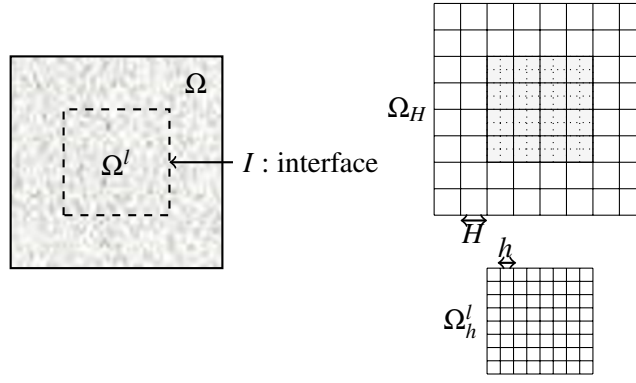


FIGURE 1 – Domaine global Ω et sous-domaine local Ω^l avec les 2 grilles correspondantes : Ω_H et Ω_h^l .

notamment envisager une prolongation du champ de déplacement (CL en déplacements imposés) ou alors du champ de contrainte (CL en forces nodales imposées). Le problème sur la grille locale s'écrit :

$$\begin{cases} \operatorname{div}(\mathbb{C} : \underline{\underline{\varepsilon}}(u_h)) = 0 & \text{dans } \Omega_h^l \\ \text{CL appropriées (Dirichlet ou Neumann)} \end{cases} \quad (5)$$

Les champs solutions $\underline{\underline{\sigma}}_h, \underline{\underline{\varepsilon}}_h, u_h$ sur la grille locale Ω_h^l , sont obtenus par résolution du problème (5).

La solution locale fine est utilisée pour améliorer la qualité de la solution grossière dans Ω_H . Cette opération consiste à :

- Effectuer une restriction dans la grille grossière des déplacements issus de la solution fine, afin d'en évaluer un résidu. En pratique le résidu de correction est calculé sur un domaine $\mathring{\Omega}_H \subset \Omega_H^l$:

$$\bar{d}_H = \operatorname{div}(\mathbb{C} : \underline{\underline{\varepsilon}}(Ru_h)) \quad \text{dans } \mathring{\Omega}_H$$

où R est un opérateur de restriction (injection, interpolation linéaire etc) de la grille fine vers la grille grossière. On notera que le résidu pourrait également être évalué à partir d'une restriction du champ de contrainte de la solution fine.

- Résoudre le problème grossier avec le résidu $\bar{d}_H$ en second membre :

$$\begin{cases} \operatorname{div}(\mathbb{C} : \underline{\underline{\varepsilon}}(u_H)) = \bar{d}_H & \text{dans } \Omega_H \\ \text{CL périodiques} \end{cases}$$

La solution grossière améliorée obtenue par résolution du problème corrigé, peut être utilisée pour définir de nouvelles CL sur la grille locale. La répétition de ce processus jusqu'à convergence constitue la méthode itérative LDC. La convergence de la méthode LDC est rapide (quelques itérations). Néanmoins, l'application de cette méthode avec des solveurs FFT (intrinsèquement associés à des CL périodiques) nécessite toutefois de préciser la stratégie de résolution du problème sur la grille locale soumise à des CL non périodiques.

3.2 Solveur FFT et Conditions aux Limites non périodiques

La résolution avec un solveur FFT périodique de problèmes faisant intervenir des CL non périodiques peut être faite en s'inspirant des méthodes d'interface immergée. Le domaine local Ω^l est plongé dans un domaine étendu Ω^e (voir Figure 2). Nous dénommerons par le terme 'buffer', la partie extérieure Ω^b entourant Ω (voir Figure 2). Les CL périodiques sont alors appliquées sur le bord du domaine étendu Ω^e et des méthodes spécifiques permettent d'imposer des CL de type Dirichlet ou Neumann sur le bord $\partial\Omega^l$ qui est vu comme une interface à l'intérieur de Ω^e . Le problème résolu sur Ω^e est alors :

$$\begin{cases} \operatorname{div} \underline{\underline{\sigma}} = 0 & \text{dans } \Omega^e \\ \underline{\underline{\sigma}} = \mathbb{C} : \underline{\underline{\varepsilon}} & \text{dans } \Omega^l \quad \text{et} \quad \underline{\underline{\sigma}} = \mathbb{C}_b : \underline{\underline{\varepsilon}} & \text{dans } \Omega^b \\ \underline{u} = \underline{E} \cdot \underline{x} + \underline{\tilde{u}} & \text{dans } \Omega^e \\ \underline{\tilde{u}} \# , \quad \underline{\underline{\sigma}} \cdot \underline{n} - \# , & \text{sur } \partial\Omega^e \\ \underline{u} = \underline{u}_d \text{ sur } \partial\Omega^l \text{ (CL de Dirichlet)} & \text{ou alors} \quad \underline{\underline{\sigma}} \cdot \underline{n} = \underline{t}, \text{ sur } \partial\Omega^l \text{ (CL de Neumann)} \end{cases}$$

Les approches pour contraindre des CL de types Dirichlet ou Neumann sur une interface immergée sont nombreuses dans la littérature (voir par exemple [7]). Ci-dessous 2 adaptations dans le cadre de solveurs FFT sont présentées succinctement.

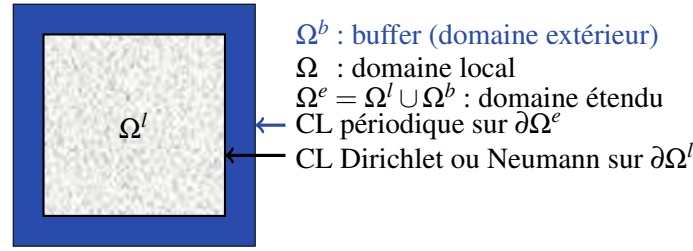


FIGURE 2 – Domaine étendue Ω^e et Conditions aux Limites (CL) sur une interface immergée.

3.2.1 Conditions aux Limites de Dirichlet

Une modification de l'algorithme de Moulinec et Suquet pour tenir compte d'une condition de type Dirichlet $\underline{u}_h = \bar{\underline{u}}_h$ a été proposée récemment dans [3]. L'idée repose sur la définition d'un champ de déformation libre $\underline{\varepsilon}(\tilde{\underline{u}}_h^*)$ dans le comportement du buffer, dérivant d'un champ de déplacement $\tilde{\underline{u}}_h^*$. Le champ $\tilde{\underline{u}}_h^*$ est défini uniquement sur les nœuds du contour I_h et est nul sur le reste du domaine. Le champ de déformation libre $\underline{\varepsilon}(\tilde{\underline{u}}_h^*)$ est donc nul partout sauf sur les voxels en contact avec les nœuds de l'interface. La démarche consiste à optimiser le champ $\tilde{\underline{u}}_h^*$ durant les itérations du point fixe du solveur FFT jusqu'à ce que la condition $\underline{u}_h = \bar{\underline{u}}_h$ soit atteinte sur les nœuds de l'interface I_h .

Cet algorithme est nommé *Dirichlet FFT* dans le reste cet article. Le lecteur est invité à se référer à l'article [3] pour plus de détails sur cet algorithme.

Il est à noter que, le champ de déplacement solution du problème grossier peut présenter des oscillations parasites (associées à l'utilisation d'éléments finis linéaires à intégration réduite comme spécifiée à la section (2)). Interpoler des CL en déplacements sur ces champs grossiers peut mener à des solutions sur Ω_h^l présentant des oscillations importantes à l'interface où ces CL sont appliquées. Il convient lors de l'opération de prolongement de filtrer ces oscillations sur les déplacements grossiers (à l'aide d'un filtre linéaire de type 'moyenne') avant interpolation des CL sur la grille fine. Le filtre 'moyenne' remplace le déplacement en chaque nœud par la moyenne pondérée sur les 27 nœuds voisins. L'opération de prolongement se décompose en 2 étapes :

- Filtrage (moyenne pondérée) des oscillations sur le champ $\underline{u}_H$.
- Interpolation des CL sur I_h :

$$\bar{\underline{u}}_h^j = P \underline{u}_H^{j-1},$$

avec P un opérateur d'interpolation linéaire.

Une fois les CL sur I_h connues, le problème sur la grille locale Ω_h^l est résolu avec l'algorithme *Dirichlet FFT* [3].

3.2.2 Conditions aux Limites de Neumann

Une approximation discrète d'une CL de type Neumann ($\underline{\sigma}(\underline{x}) \cdot \underline{n} = \underline{t}(\underline{x})$) sur $\partial\Omega^l$ peut être réalisée en ajoutant à l'équation d'équilibre un second membre $\underline{f}^*$ judicieusement choisi. L'équation d'équilibre résolue est alors :

$$\text{div } \underline{\sigma} = \underline{f}^*$$

où $\underline{f}^*$ est un champ nul partout sauf sur l'interface $\partial\Omega^l$ (plus précisément, $\underline{f}^*$ non nul sur les nœuds de l'interface dans le problème local discret). Pour la résolution du problème sur la grille locale Ω_h^l le champ de contrainte est défini nul dans le buffer (en fixant par exemple les propriétés élastiques $\mathbb{C}_b = 0$), et le champ $\underline{f}_h^*$ à appliquer à l'interface I_h se calcule simplement à partir de la solution grossière $\underline{\sigma}_H$:

- Prolongation (par interpolation linéaire) dans K^- du champ de contrainte grossier $\underline{\sigma}_H$ (voir Figure 3 pour définition de K^-) :

$$\bar{\underline{\sigma}}_h^* = P \underline{\sigma}_H \quad \text{dans } K^- \quad (6)$$

- En prolongeant $\underline{\underline{\sigma}}_h^*$ à 0 dans le buffer, on calcule à chaque nœud $\underline{k}$ de l'interface, la force nodale $\underline{f}_h^*[\underline{k}]$ à imposer :

$$\underline{f}_h^*[\underline{k}] = \text{div} \underline{\underline{\sigma}}_h^*[\underline{k}]$$

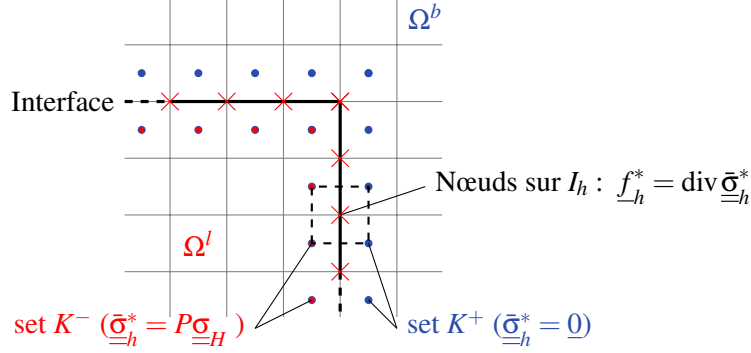


FIGURE 3 – Illustration du calcul de $\underline{f}_h^*$ à l'interface. K^- est l'ensemble des centres des voxels en vis-à-vis avec l'interface inclus dans Ω^l et K^+ centres de voxels en vis-à-vis de l'interface inclus dans Ω^b .

En pratique, une fois le champ de force $\underline{f}_h^*$ déterminé, la résolution du problème sur le domaine étendu peut être faite avec un solveur FFT standard prenant en compte la présence d'un second membre.

Le champ solution $\underline{\underline{\sigma}}_H$ dans l'équation (6) est équilibré, mais le champ prolongé peut être légèrement déséquilibré du fait de l'opération de prolongation par interpolation linéaire. La convergence en équilibre peut être légèrement améliorée en appliquant une procédure d'équilibrage global sur les forces nodales $\underline{f}_h^*$ au contour. Cette procédure va modifier légèrement les forces nodales de manière à assurer que la somme des forces $\underline{f}_h^*$ sur le contour I_h et la somme des moments des forces soient nulles.

3.3 Solveur FFT multi-grilles

A l'aide des 2 types de CL sur la grille locale Ω_h^l on peut proposer 2 versions d'un algorithme LDC avec des solveurs FFT :

Algorithme 1 : algorithme LDC 2 grilles avec déplacements imposés (**Dirichlet LDC**)

Etape 0 (initialisation). Calcul de $\underline{\underline{\sigma}}_H^0, \underline{\underline{\varepsilon}}_H^0, \underline{u}_H^0$ sur Ω_H (**algorithme standard FFT**).

Itération (j) : $\underline{\underline{\sigma}}_H^{j-1}, \underline{\underline{\varepsilon}}_H^{j-1}, \underline{u}_H^{j-1}$ connus

Etape 1. Résolution du problème local sur Ω_h^l :

- Filtre (moyenne) des oscillations parasites sur le champ de fluctuations des déplacements grossiers $\tilde{\underline{u}}_H^{j-1}$.
- *Interpolation linéaire* des CL en déplacements sur le contour de la grille locale I_h :

$$\bar{\underline{u}}_h^j = P \tilde{\underline{u}}_H^{j-1} \quad \text{sur } I_h$$

- Calcul des champs $\underline{\underline{\sigma}}_h^j, \underline{\underline{\varepsilon}}_h^j, \underline{u}_h^j$ sur Ω_h^l avec l'**algorithme FFT Dirichlet** [3].

Etape 2. Correction de la solution grossière sur Ω_H :

- *Restriction par injection* des fluctuations du déplacements $\tilde{\underline{u}}_h^j$ sur Ω_H^l

$$\bar{\underline{u}}_H^j = R \tilde{\underline{u}}_h^j \quad \text{sur } \Omega_H^l$$

- Calcul du résidu local sur $\hat{\Omega}_H$

$$\bar{\underline{d}}_H^j = \text{div} \left(\mathbb{C} : (\underline{\underline{E}} + \underline{\underline{\varepsilon}}(\bar{\underline{u}}_H^j)) \right) \quad \text{sur } \hat{\Omega}_H$$

- {facultatif} Équilibrage du résidu local $\bar{\underline{d}}_H^j$
- Calcul des champs grossiers $\underline{\underline{\sigma}}_H^j, \underline{\underline{\varepsilon}}_H^j, \underline{u}_H^j$ avec **algorithme standard FFT** avec le résidu local $\bar{\underline{d}}_H^j$ en second membre.

Etape 3. Évaluation du critère de convergence

Une version alternative de l'algorithme basée sur des CL de type Neumann (**Neumann LDC**) est obtenue en remplaçant l'étape 1 par :

Etape 1. Résolution du problème local sur Ω_h^l :

- *Interpolation linéaire* du champ de contrainte sur les centres des voxels à l'interface (ensemble K^-) :

$$\underline{\underline{\sigma}}_h^* = P \underline{\underline{\sigma}}_H \quad \text{dans } K^-$$

- Prolonger le champ de contrainte interpolé $\underline{\underline{\sigma}}_h^*$ par 0 sur K^+ , puis calculer les CL en forces nodales sur le contour de la grille locale I_h :

$$\underline{f}_h^* = -\text{div } \underline{\underline{\sigma}}_h^* \quad \text{sur } I_h$$

- {facultatif} Procédure d'équilibrage des forces nodales $\underline{f}_h^*$
- Calcul $\underline{\underline{\sigma}}_h^*, \underline{\underline{\varepsilon}}_h^j, \underline{u}_h^j$ sur Ω_h^l avec (**l'algorithme standard FFT**).

A chaque cycle du LDC, les solutions obtenues au cycle précédent sont utilisées pour initialiser les sous-algorithmes *standard FFT* ou *Dirichlet FFT*. Ainsi, au fil des itérations LDC, le nombre d'itérations à convergence pour chacun des problèmes (global et local) diminue. La convergence du LDC est évaluée en comparant les solutions obtenues entre 2 cycles successifs :

$$\frac{\|\underline{\underline{\sigma}}_h^j - \underline{\underline{\sigma}}_h^{j-1}\|}{\|\underline{\underline{\sigma}}_h^j\|} < \varepsilon \quad \text{dans } \Omega_h^l \quad \text{et} \quad \frac{\|\underline{\underline{\sigma}}_H^j - \underline{\underline{\sigma}}_H^{j-1}\|}{\|\underline{\underline{\sigma}}_H^j\|} < \varepsilon \quad \text{dans } \Omega_H \setminus \Omega_h^l$$

4 Exemple d'application : zoom local sur un polycristal 3D

4.1 Description du cas

La précision et les performances de l'algorithme LDC sont investiguées sur un exemple académique. Les calculs sont réalisés sur un solveur FFT non parallélisé sur le programme MATLAB 2018. Le processeur utilisé est l'Intel Xeon 6132 avec 2×14 cœurs, 2.6 GHZ, et 192 GO de RAM.

Lorsque l'on souhaite évaluer de manière fine les champs locaux dans un agrégat de grains, les CL périodiques au bord du domaine peuvent soulever la question de représentativité du voisinage. Une approche consiste à considérer un domaine de grandes dimensions et à définir comme domaine d'étude une zone interne (ZOI = Zone Of Interest) relativement éloignée des bords où les CL périodiques sont appliquées. Les CL aux bords de la ZOI résultent alors du comportement des grains voisins. Pour limiter la taille du problème à étudier, l'utilisation d'un raffinement local permet de définir une discrétisation fine dans la ZOI, tout en utilisant une discrétisation grossière dans les grains du voisinage.

Dans cet exemple académique, on considère une représentation (tessellation de Voronoï) d'un polycristal de taille $5d \times 5d \times 5d$ composée de $15 \times 15 \times 15$ grains. Au centre du domaine on définit une ZOI de taille $d \times d \times d$ contenant approximativement $3 \times 3 \times 3$ grains. Les grains ont un comportement élastique linéaire avec un module d'Young aléatoirement choisi entre 1 GPa et 3 GPa. Le coefficient de Poisson ν est fixé à 0.3. La ZOI considérée comme le domaine local Ω^l est discrétisée par 105^3 voxels (approximativement 35 voxels par grain dans chaque direction). En dehors de la ZOI ($\Omega \setminus \Omega^l$) la discrétisation est déraffinée. 3 niveaux de déraffinement sont investigués : $r = 3$ (≈ 11 voxels par grain), $r = 5$ (≈ 7 voxels par grain) and $r = 7$ (≈ 5 voxels par grain). La cellule est soumise à une déformation macroscopique de 1 % dans la direction x . Le nombre de voxels dans l'épaisseur du buffer pour imposer les CL en déplacements ou en forces nodales sur la grille locale est pris comme $N_b \approx 0.2 * N_0$ dans le cas des CL en forces et $N_b \approx 0.3 * N_0$ dans le cas des CL en déplacements, où $N_0 = 105$ voxels (nombre de voxels dans une direction de Ω^l).

4.2 Précision et performances du LDC

Les résultats du calcul LDC avec raffinement local sont comparés au calcul réalisé avec maillage fin sur l'intégralité du domaine. Une comparaison qualitative du champ de contrainte de von Mises dans Ω^l (Figure 4) montre un bon accord entre la solution du LDC et la solution de référence (calcul partout fin). Afin de quantifier les performances 3 quantités sont présentées dans le Tableau 1 :

— La norme de l'erreur sur le champ de contrainte évaluée par :

$$\bar{e}_{\underline{\underline{\sigma}}} = \frac{\|\underline{\underline{\sigma}}_h^s - \underline{\underline{\sigma}}_h\|_{\Omega_h^l}}{\|\underline{\underline{\sigma}}_h^s\|_{\Omega_h^l}}.$$

où $\underline{\underline{\sigma}}_h^s$ est la solution du problème raffiné partout.

- Le ratio entre la taille du problème multi-grilles (nombres de voxels) et le problème raffiné partout.
- Le temps CPU.

L'observation des données du Tableau 1 illustre la pertinence d'une approche multi-grilles locale dans ce contexte, car les champs solutions dans la ZOI du calcul LDC sont similaires à ceux du calcul raffiné partout à 1% près. Toutefois le problème multi-grilles a nécessité entre 15 et 40 fois moins de voxels en fonction du niveau de déraffinement, et des temps CPU 3 à 11 fois inférieurs au temps du calcul raffiné partout. Ces résultats sont cohérents avec ceux observés dans le cadre de l'application de la méthode LDC avec des solveurs éléments finis [1]. L'approche multigrille locale permet d'obtenir une solution relativement précise tout en réduisant de manière non négligeable le coût du calcul en terme d'espace mémoire et de temps CPU.

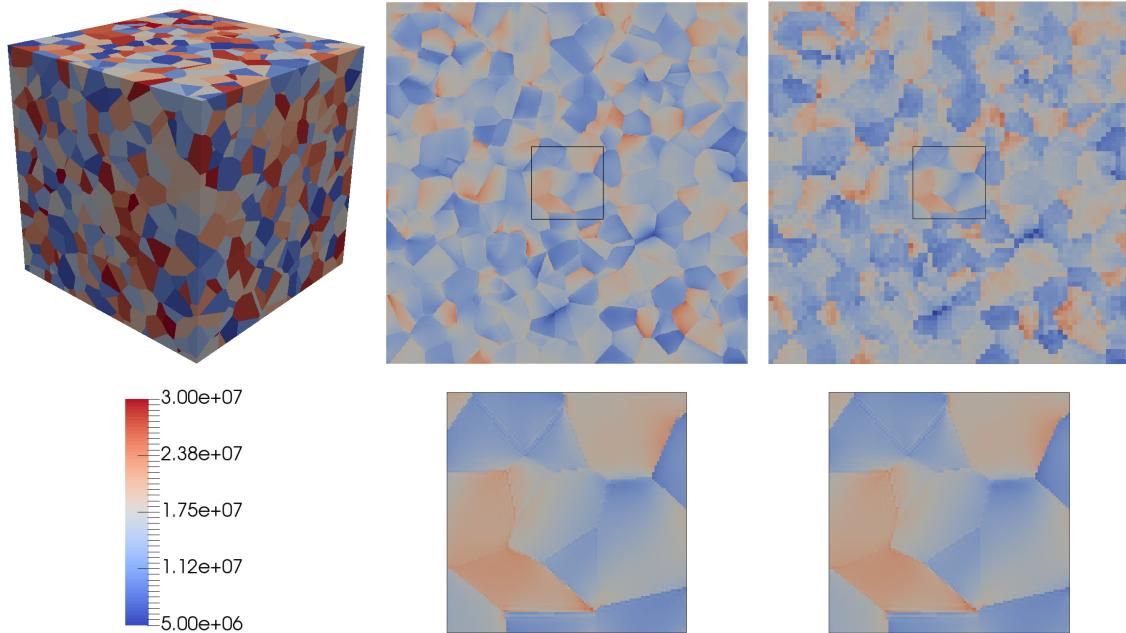


FIGURE 4 – Polyèdre de Voronoï avec 15^3 grains dans Ω et 3^3 grains dans Ω^l (ZOI). Comparaison du champ de contraintes de von Mises (Pa) dans une section au milieu du domaine. (Gauche) solution avec raffinement partout ($\approx 35^3$ voxels/grain). (Droite) solution calcul multi-grilles avec un déraffinement de $r = 7$ ($\approx 35^3$ voxels/grain dans Ω_h^l et $\approx 5^3$ voxels/grain dans $\Omega_H \setminus \Omega_H^l$) obtenue avec l'algorithme Neumann LDC. Des résultats similaires ont également été obtenus avec la version Dirichlet LDC.

r	Dirichlet LDC			Neumann LDC		
	$\bar{e}_{\underline{\underline{\sigma}}} (\%)$	taille relative (%)	temps CPU (s)	$\bar{e}_{\underline{\underline{\sigma}}} (\%)$	taille relative (%)	temps CPU (s)
$r = 3$	0.82	6.9	546	0.66	5.9	334
$r = 5$	1.33	4.0	326	1.05	3.0	185
$r = 7$	1.84	3.5	319	1.41	2.5	161

TABLE 1 – Données sur les performances et la précision de la méthode en comparaison au calcul raffiné partout. Le temps CPU pour le calcul raffiné partout est 1838 s.

5 Conclusions et perspectives

Le travail présenté ici propose une extension du cadre d'application des solveurs FFT aux simulations multi-grilles locales, dont notamment la méthode LDC (Local Defect Correction [4]). Deux implémentations de la méthode LDC ont été proposées selon que les CL du problème local sont formulées à partir d'une prolongation du champ de déplacement ou du champ de contrainte issu du problème global. Chacune de ces implémentations a nécessité d'étendre l'utilisation des solveurs FFT, intrinsèquement associés à des CL périodiques, à des CL en déplacements (Dirichlet) ou forces (Neumann) imposés au contour.

Un cas test académique constitué d'un agrégat de grains élastiques isotropes, avec un zoom sur quelques grains définissant le problème local, a permis de valider la méthode et les deux implémentations proposées. Les gains en temps de calcul et en mémoire sont importants et la méthode semble prometteuse.

Les perspectives de ce travail sont nombreuses :

- Présenté dans le cadre de l'élasticité linéaire, il devra être étendu aux comportements non-linéaires en petites perturbations puis en grandes transformations.
- L'implémentation dans un code massivement parallèle en mémoire distribuée AMITEX_FFTP[2], nécessitera de mettre en place une structure de communication adaptée pour assurer le dialogue entre les simulations locale et globale.
- La méthode multi-grilles locale proposée décrit le couplage de deux simulations FFT pour la résolution du problème global et du problème local. Une alternative pourrait être de réaliser un couplage entre un code éléments-finis pour le problème global et un solveur FFT pour le problème local.

Références

- [1] L. Barbié, I. Ramière, and F. Lebon. *Strategies involving the local defect correction multi-level refinement method for solving three-dimensional linear elastic problems*, Computers & Structures, 130 :73–90, 2014.
- [2] L. Gélébart. *Amitex-fft documentation.*, http://www.maisondelasimulation.fr/projects/amitex/user_guide/_build/html/map-site-user.html, 2021.
- [3] L. Gélébart. *A modified FFT-based solver for the mechanical simulation of heterogeneous materials with Dirichlet boundary conditions*, Comptes Rendus Mécanique, 693-704, 2020.
- [4] W. Hackbusch. *Local Defect Correction Method and Domain Decomposition Techniques*. In Klaus Böhmer and Hans J. Stetter, editors, Defect Correction Methods : Theory and Applications, Computing Supplementum, 89–113, 1984.
- [5] H. Moulinec and P. Suquet. *A FFT-Based Numerical Method for Computing the Mechanical Properties of Composites from Images of their Microstructures*. In R. Pyrz, ed, Microstructure-Property Interactions in Composite Materials, Solid Mechanics and Its Applications, 235–246, 1995.
- [6] H. Moulinec and P. Suquet. *A numerical method for computing the overall response of nonlinear composites with complex microstructure*, Computer Methods in Applied Mechanics and Engineering, 157(1) :69–94, 1998.
- [7] I. Ramière, P. Angot, and M. Belliard. *A general fictitious domain method with immersed jumps and multilevel nested structured meshes*, Journal of Computational Physics, 225(2) :1347–1387, 2007.
- [8] M. Schneider. *A review of nonlinear FFT-based computational homogenization methods*, Acta Mechanica, 2021.
- [9] M. Schneider, D. Merkert, and M. Kabel. *FFT-based homogenization for microstructures discretized by linear hexahedral elements*, International Journal for Numerical Methods in Engineering, 109(10) :1461–1489, 2017.
- [10] F. Willot. *Fourier-based schemes for computing the mechanical response of composites with accurate local fields*, Comptes Rendus Mécanique, 343(3) :232–245, 2015.