

Couplage Global-Local en asynchrone pour des problèmes linéaires

Ahmed EL KERIM^{1,3}, Pierre GOSSELET², Frédéric MAGOULES³,

¹ Université Paris-Saclay, ENS Paris-Saclay, CNRS, LMT, Gif-sur-Yvette, France, ahmed.elkerim@ens-paris-saclay.fr

² Université de Lille, CNRS, Centrale Lille / LaMcube, pierre.gosselet@univ-lille.fr

³ Université Paris-Saclay, CentraleSupélec/ MICS, Gif-sur-Yvette, France, {ahmed.el-kerim,frederic.magoules}@centralesupelec.fr

Résumé — Une version parallèle asynchrone du couplage global-local non-intrusif est mise en place. Le cas de nombreux patches, y compris couvrant l'intégralité de la structure est étudié. L'asynchronisme permet de limiter la dépendance aux communications, aux pannes et au déséquilibre de charge. Nous détaillons la méthode et illustrons ses performances sur un cas académique.

Mots clés — calcul parallèle asynchrone, couplage non-intrusif, décomposition de domaine.

1 Introduction

De nombreuses modélisations industrielles sont définies par une hiérarchie de modèles. À chaque niveau, le modèle le plus grossier permet d'estimer les grands flux d'efforts dans le domaine, il est enrichi par un ensemble de patches fins qui viennent préciser localement la géométrie, le maillage, les propriétés matériau. Pour coupler les modèles, la stratégie couramment utilisée est le zoom structural (*submodeling*) qui consiste à imposer des conditions de Dirichlet issues du calcul grossier sur les modèles fins. En négligeant les effets des patches sur le modèle global, la méthode est susceptible de conduire à de grandes erreurs sur les quantités mécaniques d'intérêt.

Le couplage global/local [1] consiste à faire remonter l'influence des patches par un effort appliqué au modèle global. Cet effort est obtenu à l'aide d'itérations qui permettent une mise en œuvre non-intrusive de la méthode autour de codes commerciaux (par exemple code_aster ou abaqus). Cette méthode peut s'interpréter de nombreuses manières, notamment comme une méthode de décomposition de domaine de Schwarz [2, 8]. Ce genre de méthode est bien adapté au calcul parallèle, et des travaux récents [3, 4, 5, 6, 7] ont montré qu'elles supportaient les itérations asynchrones qui permettent une meilleure tolérance aux latences réseaux, au déséquilibre de charge et aux architectures fortement hétérogènes.

Ce papier présente la version asynchrone du couplage global/local. Après une présentation rapide des fondamentaux de la méthode dans la section 2, l'asynchronisme est introduit dans la section 3. La section 4 présente une étude numérique de validation sur un cas-test académique d'élasticité linéaire 3D.

2 Couplage Global-Local

On considère une structure définie à plusieurs échelles : un modèle Grossier représentant la globalité du domaine est localement corrigé par N patches Fins. Les patches peuvent couvrir l'entièreté du domaine [10] ou non [9]. On note $\Omega^{(s)}$ avec $s > 0$ les zones d'intérêt dont il existe une représentation Grossière et une représentation Fine. Si elle existe, on note $\Omega^{(0),G}$ la zone Grossière non couverte par des patches, parfois appelée zone complémentaire. L'ensemble $(\Omega^{(s),G})_{s \geq 0}$ constitue le modèle Grossier, alors que $(\Omega^{(0),G}, (\Omega^{(s),F})_{s > 0})$ constitue la référence, comme illustré sur les figures 1 et 2.

L'objectif du couplage est de trouver la solution du problème de référence en alternant les calculs sur le modèle grossier et sur les modèles fins.

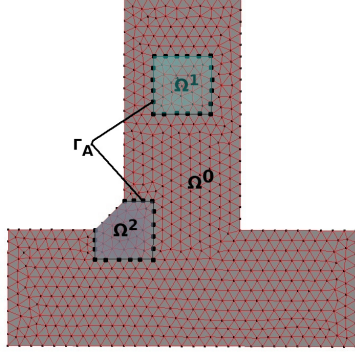


FIGURE 1 – Problème Global

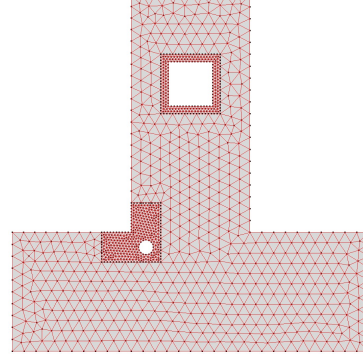


FIGURE 2 – Référence

2.1 Problème global et problème de référence

On se place dans des hypothèses de thermique ou mécanique linéaire (élasticité en petite perturbation). On utilise une discrétisation élément fini. On note \mathbf{K} les matrices de rigidité (ou leur équivalent thermique) qui sont symétriques semi-définies positives, \mathbf{f}_{ext} les vecteurs de chargements imposés, et \mathbf{u} le vecteur inconnu (température ou déplacement).

Le problème Global, indicé par G , est une simplification du problème, permettant un calcul rapide mais représentatif des flux à grande échelle. Il s'écrit :

$$-\mathbf{K}^G \mathbf{u}^G + \mathbf{f}_{ext}^G = 0. \quad (1)$$

Suite à la décomposition en sous-domaines (complémentaire et patches), on peut séparer les degrés de liberté de l'interface (indexé par b) et les degrés de liberté internes (indexé par i). On peut réécrire l'équilibre grossier sur chaque sous-domaine en introduisant les réactions nodales $\lambda^{(s),G}$ et l'opérateur de trace $\mathbf{T}^{(s)}$ tel que $\mathbf{T}^{(s)} \mathbf{u}^{(s),G} = \mathbf{u}_b^{(s),G}$:

$$-\mathbf{K}^{(s),G} \mathbf{u}^{(s),G} + \mathbf{f}_{ext}^{(s),G} + \mathbf{T}^{(s)T} \lambda^{(s),G} = 0. \quad (2)$$

Ce système s'écrit sous forme détaillée :

$$\begin{pmatrix} \mathbf{K}_{ii}^{(s),G} & \mathbf{K}_{ib}^{(s),G} \\ \mathbf{K}_{bi}^{(s),G} & \mathbf{K}_{bb}^{(s),G} \end{pmatrix} \begin{pmatrix} \mathbf{u}_i^{(s),G} \\ \mathbf{u}_b^{(s),G} \end{pmatrix} = \begin{pmatrix} \mathbf{f}_{ext,i}^{(s),G} \\ \mathbf{f}_{ext,b}^{(s),G} + \lambda^{(s),G} \end{pmatrix}. \quad (3)$$

D'après les hypothèses, nous pouvons éliminer les degrés de liberté internes, condenser le problème sur l'interface et introduire un opérateur de Dirichlet-Neumann \mathbf{S} (complément de Schur primal) :

$$\lambda^{(s),G} = \mathbf{S}^{(s),G} \mathbf{u}_b^{(s),G} - \mathbf{b}^{(s),G} \quad \text{avec} \quad \begin{cases} \mathbf{S}^{(s),G} = \mathbf{K}_{bb}^{(s),G} - \mathbf{K}_{bi}^{(s),G} (\mathbf{K}_{ii}^{(s),G})^{-1} \mathbf{K}_{ib}^{(s),G} \\ \mathbf{b}^{(s),G} = \mathbf{f}_{ext,b}^{(s),G} - \mathbf{K}_{bi}^{(s),G} (\mathbf{K}_{ii}^{(s),G})^{-1} \mathbf{f}_{ext,i}^{(s),G} \end{cases}. \quad (4)$$

Afin de connecter les sous-domaines, nous introduisons les opérateurs d'assemblage ($\mathbf{A}^{(s)}$), qui sont des matrices booléennes creuses qui projettent les bords des sous-domaines sur l'interface globale. Les conditions d'interface peuvent être écrites comme suit :

- continuité du déplacement : $\exists \mathbf{u}_A^G$ tel que $\forall s \in [0..N]$, $\mathbf{u}_b^{(s),G} = \mathbf{A}^{(s)T} \mathbf{u}_A^G$,
- équilibre des réactions nodales : $\sum_{s=0}^N \mathbf{A}^{(s)} \lambda^{(s),G} = 0$.

Avec ces notations, le problème global s'écrit :

$$\begin{aligned} &\text{Trouver le déplacement global } \mathbf{u}_A^G \text{ tel que} \\ &\underbrace{\left(\sum_{s=0}^N \mathbf{A}^{(s)} \mathbf{S}^{(s),G} \mathbf{A}^{(s)T} \right)}_{\mathbf{S}^G} \mathbf{u}_A^G = \underbrace{\sum_{s=0}^N \mathbf{A}^{(s)} \mathbf{b}^{(s),G}}_{\mathbf{b}^G} \end{aligned} \quad (5)$$

Il suffit de supposer qu'au moins un des sous-domaines possède suffisamment de conditions de Dirichlet imposées pour que ce problème soit symétrique défini positif et donc bien posé.

Le problème de référence consiste à remplacer les zones d'intérêt grossières par leur représentation fine. De même que pour le grossier, on utilise la condensation pour faire apparaître les opérateurs d'interface. Pour plus de souplesse, on autorise des maillages grossiers et fins non aux interfaces, et on introduit l'opérateur d'interpolation entre le global et le local, noté $\mathbf{J}^{(s)}$. On adopte la cinématique grossière à l'interface, de sorte que $\mathbf{u}_b^{(s),F} = \mathbf{J}^{(s)} \mathbf{u}_b^{(s),G}$. De même l'équilibre des réactions est évalué sur l'interface globale.

Ces hypothèses conduisent à formuler le problème de référence sous la forme suivante :

$$\text{Trouver le déplacement de référence } \mathbf{u}_A^R \text{ tel que}$$

$$\underbrace{\left(\mathbf{A}^{(0)} \mathbf{S}^{(0),G} \mathbf{A}^{(0)T} + \sum_{s=1}^N \mathbf{A}^{(s)} \mathbf{J}^{(s)T} \mathbf{S}^{(s),F} \mathbf{J}^{(s)} \mathbf{A}^{(s)T} \right)}_{\mathbf{S}^R} \mathbf{u}_A^R = \underbrace{\mathbf{A}^{(0)} \mathbf{b}^{(0)} + \sum_{s=1}^N \mathbf{A}^{(s)} \mathbf{J}^{(s)T} \mathbf{b}^{(s),F}}_{\mathbf{b}^R}. \quad (6)$$

2.2 Le couplage

Le couplage global-local est une technique non-intrusive qui permet de retrouver la solution du problème de référence en résolvant des problèmes grossiers modifiés par l'introduction d'un effort supplémentaire (opération supportée par tous les codes, même commerciaux) et en résolvant des problèmes fins indépendants par patches.

On peut interpréter l'algorithme comme la résolution de (6) en utilisant un préconditionnement à droite, autrement dit un changement de variable. On introduit une charge externe \mathbf{p}_A^G appliquée à l'interface Γ dans le problème global et on exprime le déplacement de référence comme la solution du problème global modifié :

$$\begin{aligned} &\text{Définir le déplacement de référence } \mathbf{u}_A^R \text{ tel que} \\ &\mathbf{S}^G \mathbf{u}_A^R = \mathbf{p}_A^G + \mathbf{b}^G \end{aligned} \quad (7)$$

On a $\mathbf{p}_A^G = \sum \mathbf{A}^{(s)} \boldsymbol{\lambda}^{(s),G}$ et donc \mathbf{p}_A^G peut être interprété comme un manque d'équilibre imposé aux réactions nodales sur l'interface dans le domaine global. Ce déséquilibre dans le modèle global doit compenser son inacuité par rapport au modèle fin :

$$\begin{aligned} &\text{Trouver l'intereffort } \mathbf{p}_A^G \text{ tel que} \\ &\mathbf{S}^R \mathbf{S}^{G^{-1}} \mathbf{p}_A^G = \mathbf{b}^R - \mathbf{S}^R \mathbf{S}^{G^{-1}} \mathbf{b}^G. \end{aligned} \quad (8)$$

Si on utilise $\mathbf{S}^R = (\mathbf{S}^R - \mathbf{S}^G) + \mathbf{S}^G$, on peut réécrire le problème sous la forme suivante, où on introduit le résidu \mathbf{r} :

$$\text{Trouver } \mathbf{p}_A^G \text{ tel que } \mathbf{r} := - \left(\mathbf{p}_A^G - (\mathbf{S}^G - \mathbf{S}^R) \mathbf{S}^{G^{-1}} \mathbf{p}_A^G - (\mathbf{b}^R - \mathbf{S}^R \mathbf{S}^{G^{-1}} \mathbf{b}^G) \right) = 0 \quad (9)$$

Cette équation invite à utiliser un algorithme itératif stationnaire, le plus simple étant la méthode de Richardson $\mathbf{p}_{A,j+1}^G = \mathbf{p}_{A,j}^G + \mathbf{r}_j$, mais de nombreuses techniques d'accélération sont possibles et ont été testées avec succès comme Aitken, Quasi-Newton, gradient conjugué (voir [8] pour une rapide revue).

Il est à noter que le résidu possède une interprétation mécanique simple. Partant d'un intereffort $\mathbf{p}_{A,j}$, on calcule le déplacement global $\mathbf{u}_{A,j}$ et l'éventuelle réaction de la zone complémentaire $\boldsymbol{\lambda}_j^{(0),G}$; on applique $\mathbf{J}^{(s)} \mathbf{u}_{A,j}$ sur les modèles fins et on récupère la réaction $\boldsymbol{\lambda}_j^{(s),F}$ associée; le résidu correspond à :

$$\mathbf{r}_j = - \left(\mathbf{A}^{(0)} \boldsymbol{\lambda}_j^{(0),G} + \sum_{s=1}^N \mathbf{A}^{(s)} \mathbf{J}^{(s)T} \boldsymbol{\lambda}_j^{(s),F} \right). \quad (10)$$

Autrement dit, le résidu correspond au déséquilibre des efforts de réaction entre les sous-domaines du problème de référence.

L'algorithme 1 présente la version synchrone du couplage global-local. Le vocabulaire du parallélisme avec échange de messages est employé.

Initialisation de $\mathbf{p}_A^G = 0$ et $(\mathbf{q}^{(s)} = 0)$ pour le model *global* (rank 0)

if Rang 0 **then**

 Réception de tous les $\mathbf{q}^{(s)}$ des autres rangs

 Calcul du résidu : $\mathbf{r}_A = -\sum_{s=0}^N \mathbf{A}^{(s)} \mathbf{q}^{(s)}$

if $\|\mathbf{r}_A\|$ est suffisamment petit (*initialisation exclue*) **then**

 | break

end

 Màj $\mathbf{p}_A = \mathbf{p}_A + \omega \mathbf{r}_A$

 Résolution globale : $\mathbf{u}_A^G = \mathbf{S}^{G^{-1}} (\mathbf{b}^G + \mathbf{p}_A^G)$

 Envoi de $(\mathbf{A}^{(s)})^T \mathbf{u}_A^G$ sur les rangs locaux

 S'il y a un complémentaire, calcul de $\mathbf{q}^{(0)} := \lambda^{(0),G}$.

end

if Rang $s > 0$ (*patches*) **then**

 Réception de $\mathbf{u}_b^{(s),F} := \mathbf{J}^{(s)} \mathbf{A}^{(s)T} \mathbf{u}_A^G$ du Rang 0

 Résolution locale fine $\lambda^{(s),F} = \mathbf{S}^{(s),F} \mathbf{u}_b^{(s),F} - \mathbf{b}^{(s),F}$

 Envoi de $\mathbf{q}^{(s)} := \mathbf{J}^{(s)T} \lambda^{(s),F}$ au Rang 0

end

Algorithm 1: Algorithme du couplage synchrone avec relaxation

3 Version Asynchrone

En asynchrone, on doit présumer que les données disponibles peuvent être en retard par rapport à l'itération courante. Dans la méthode globale-locale, le calcul global joue un rôle centralisateur, et on introduit les fonctions $\sigma^{(s)}(j) \leq j$ pour symboliser le retard dans l'information issue du sous-domaine s disponible à l'itération j . On reprend en asynchrone le schéma itératif de Richardson en employant l'équation (??) pour calculer le résidu avec les informations disponibles :

$$\mathbf{r}_j = -\mathbf{A}^{(0)} \lambda_j^{(0),G} - \sum_{s=1}^N \mathbf{A}^{(s)} \mathbf{J}^{(s)T} \lambda_{\sigma^{(s)}(j)}^{(s),F} \quad (11)$$

Pour illustrer les approches synchrone et asynchrone, nous utilisons le problème de la figure 2 avec 2 patches. Le graphique 3 montre une séquence d'itérations dans le cas synchrone du couplage global-local où nous pouvons voir l'effet de la synchronisation et le temps d'attente nécessaire à chaque processeur avant de passer d'une itération à une autre en attendant de recevoir ou d'envoyer des informations :

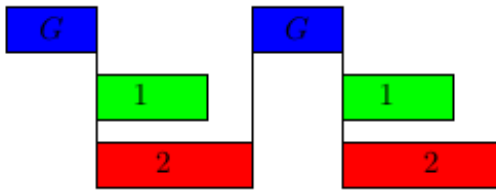


FIGURE 3 – Deux itérations synchrones

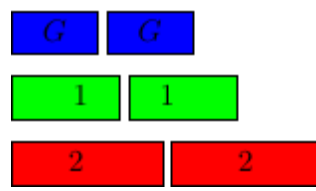


FIGURE 4 – Deux « itérations asynchrones »

Dans le graphique 4 et l'algorithme 2, nous observons la séquence d'itérations dans le cas du couplage asynchrone où chaque processeur avance avec les données disponibles pour continuer, et rend son information aux autres dès qu'il a fini de la calculer. L'algorithme 2 utilise le vocabulaire du parallélisme par accès mémoire à distance (RMA).

La détection de convergence peut se révéler complexe en asynchrone [16, 17]. Un grand avantage du couplage global/local est l'assemblage du résidu sur le modèle grossier, il est donc très simple de calculer un critère d'arrêt. Concernant la preuve théorique de la convergence de la méthode, l'utilisation du cadre des paracontractions [13] permet de démontrer que si un coefficient de relaxation ω permet de faire converger la méthode synchrone, alors ce même coefficient est suffisant pour faire converger en asynchrone.

```

Initialisation  $\mathbf{p}_A^G = 0$  et  $(\mathbf{q}^{(s)} = 0)$  sur le modèle global (rank 0)
if Rang 0 détecte la réception d'une nouvelle valeur  $\mathbf{q}^{(s)}$  d'un rang  $s$  then
    Calcul du residu :  $\mathbf{r}_A = -\sum_{s=0}^N \mathbf{A}^{(s)} \mathbf{q}^{(s)}$ 
    if  $\|\mathbf{r}_A\|$  est suffisamment petit (initialisation exclue) then
        | break
    end
    Màj  $\mathbf{p}_A = \mathbf{p}_A + \omega \mathbf{r}_A$ 
    Résolution globale :  $\mathbf{u}_A^G = \mathbf{S}^{G^{-1}} (\mathbf{b}^G + \mathbf{p}_A^G)$ 
    Global puts  $(\mathbf{A}^{(s)T} \mathbf{u}_A^G)$  sur les rangs locaux
    S'il y a un complémentaire, calcul de  $\mathbf{q}^{(0)} := \boldsymbol{\lambda}^{(0),G}$ .
end
if sous-domaine  $s > 0$  est disponible et détecte l'arrivée d'une nouvelle valeur
     $\mathbf{u}_b^{(s),F} := \mathbf{J}^{(s)} \mathbf{A}^{(s)T} \mathbf{u}_A^G$  then
        Résolution locale fine  $\boldsymbol{\lambda}^{(s),F} = \mathbf{S}^{(s),F} \mathbf{u}_b^{(s),F} - \mathbf{b}^{(s),F}$ 
        Patch  $s$  puts  $\mathbf{q}^{(s)} := \mathbf{J}^{(s)T} \boldsymbol{\lambda}^{(s),F}$  sur le Rang 0
    end

```

Algorithm 2: Algorithme du couplage asynchrone

Du point de vue de l'implémentation, plusieurs techniques existent dans la littérature. Une première repose sur des communications MPI point-à-point non bloquantes comme dans [11]. Une seconde, comme [12], se base sur les communications unilatérales ou MPI- RDMA, qui consistent à autoriser un processeur à directement lire ou écrire le contenu de la mémoire des autres processeurs. Nous avons retenu cette deuxième stratégie dans notre mise en œuvre.

4 Validation sur un cas académique

Pour réaliser notre étude numérique, nous avons construit un cas test permettant d'évaluer l'extensibilité faible de la méthode. Il s'agit d'une « poutre 3D » hétérogène en élasticité linéaire, constituée de la répétition selon les 3 axes d'un sous-domaine cubique. La version fine du sous-domaine comporte une inclusion sphérique de module d'Young 10 fois plus faible que celui de la matrice, alors que la version grossière est homogène constitué uniquement de matrice (le coefficient de Poisson est constant, égal à 0,3), avec un maillage moins fin voir les figures 6, 7 et 8. Pour ce cas test, les patchs fins recouvrent l'intégralité du domaine, il n'y a donc pas de zone complémentaire $\Omega^{(0)}$. Le chargement volumique est identique dans tous les modèles $f = (1, 1, 1)^T$. Un côté de la poutre est encasté, les autres faces sont libres.

La géométrie est générée à l'aide du logiciel gmsh [15] et la bibliothèque getfem [14] est utilisée depuis notre code python pour mettre en place l'approximation élément fini. Les calculs ont été conduits sur le supercalculateur Ruche commun à l'ENS Paris-Saclay et CentraleSupélec.

Pour cette étude, nous avons retenu une géométrie à $2 \times 2 \times 4$ patchs, voir la figure 5. Le modèle global complet (16 cubes) comporte environ 8000 degrés de liberté, chaque modèle fin (1 cube avec inclusion) comporte 3750 degrés de liberté.

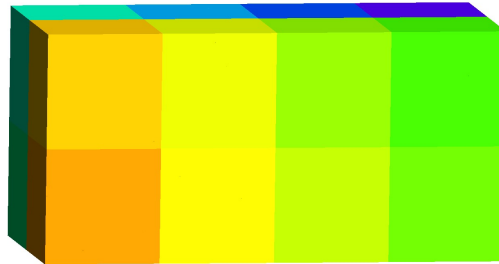


FIGURE 5 – 16 sous-domaines avec $\Omega^0 = \emptyset$

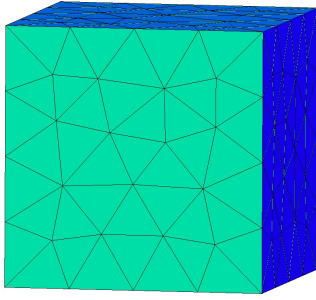


FIGURE 6 – Global $\Omega^{(s),G}$

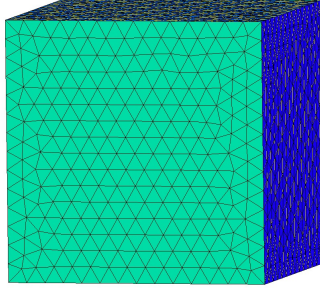


FIGURE 7 – Fin $\Omega^{(s),F}$

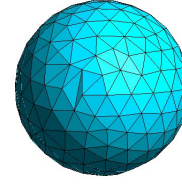


FIGURE 8 – $\Omega_{in}^{(s),F}$: Inclusion dans le fin $\Omega^{(s),F}$

Notre étude consiste à faire varier les ressources allouées au calcul. Le tableau 1 présente les performances en termes d'itérations et la figure 9 en termes de temps de calcul. Nous comparons les versions synchrone et asynchrone pour un même coefficient de relaxation déterminé de manière empirique pour optimiser les performances en asynchrone. Nous présentons également les performances de la méthode Aitken, qui est une variante synchrone où le coefficient de relaxation optimal est estimé à chaque itération par une heuristique faisant intervenir des calculs de produits scalaires.

La méthode Aitken conduit aux meilleures performances pour tous nos indicateurs. Néanmoins, on observe que la méthode asynchrone n'est pas très éloignée en termes de temps de calcul.

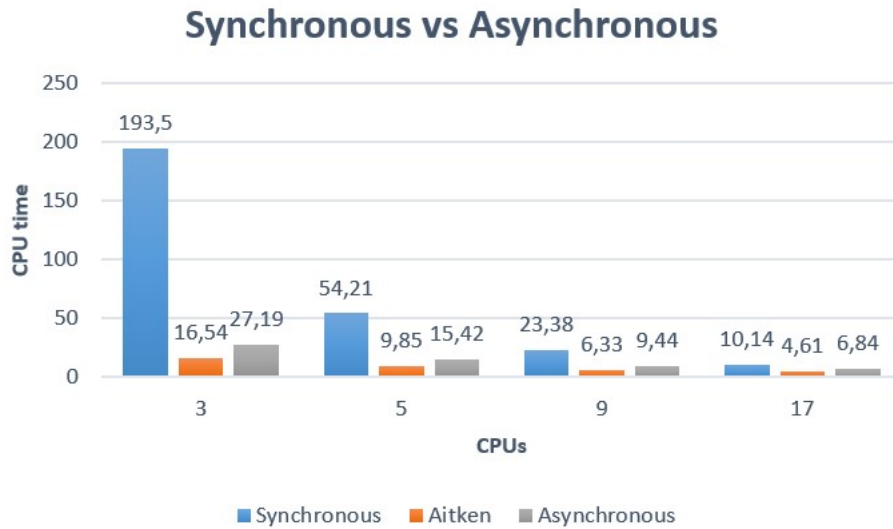


FIGURE 9 – Comparaison des temps de calcul suivant les méthodes et ressources allouée

CPU	Relaxation ω	Synchrone #it	Asynchrone #itglo[#itfin]	Aitken #it (sync)
3 $\{1 + [8, 8]\}$	100	201	197[33 - 35]	16
5 $\{1 + [4, \dots, 4]\}$	200	100	108 [34 - 41]	16
9 $\{1 + [2, \dots, 2]\}$	300	67	61 [41 - 55]	16
17 $\{1 + [1, \dots, 1]\}$	510	39	41 [54 - 75]	16

TABLE 1 – Nombre d'itérations pour les différentes variantes, le coefficient de relaxation est optimisé pour les performances asynchrones. En asynchrone on distingue les itérations sur le modèle global et les patches fins.

Dans la figure ci-dessus, nous pouvons voir le temps de calcul effectué par le modèle asynchrone et celui effectué en synchrone. Nous commençons avec 8 sous-domaines par CPU + un CPU pour calculer le problème global, et en augmentant le nombre de CPUs à chaque fois, nous terminons avec un sous-domaine par CPU. Le résultat montre que globalement le temps de calcul du modèle asynchrone relaxé

avec un coefficient constant se rapproche des temps de calcul du modèle synchrone avec L'Aitken qui est un algorithme d'optimisation de coefficient de relaxation d'une itération à l'autre et qui converge en très peu d'itération, pour le même coefficient de relaxation constant pour les deux modèles on peut voir que le modèle asynchrone est bien plus rapide. Dans le tableau, nous résumons le nombre d'itérations effectuées dans chaque cas en synchrone on fait le même nombre d'itérations partout contrairement à l'asynchrone où chacun fait son propre nombre d'itérations qui diffèrent entre le global et les locaux.

5 Conclusion

Nous avons présenté une version asynchrone de la méthode de calcul global/local non-intrusive, et nous avons proposé une mise en œuvre par des techniques de parallélisation MPI RMA. Le cadre global/local est avantageux pour l'asynchronisme car on peut facilement prouver et contrôler la convergence.

Les performances présentées sont intéressantes : en terme de temps de calcul, la méthode asynchrone se rapproche de la redoutable variante Aitken du calcul synchrone.

Durant la présentation, d'autres exemples académiques avec plus de sous-domaines et un exemple industriel seront présentés. D'autres architectures, moins défavorable qu'un cluster homogène, seront également testées.

Remerciements : ce travail est réalisé dans le cadre du projet ANR ADOM [ANR-18-CE46-0008].

Références

- [1] Lionel Gendre, Olivier Allix, Pierre Gosselet, and François Comte *Non-intrusive and exact global/local techniques for structural problems with local plasticity.*, Computational Mechanics, 44(2) :233-245, 2009.
- [2] F. Hecht, A. Lozinski, and O. Pironneau. *Numerical zoom and the Schwarz algorithm.*, Proceedings of the 18th conference on domain decomposition methods, 2009.
- [3] F Magoules, C Venet. *Asynchronous iterative sub-structuring methods*, Mathematics and Computers in Simulation, 145, 34-49, 2018
- [4] José C Garay, Frédéric Magoules, Daniel B Szyld. *Synchronous and asynchronous optimized Schwarz method for Poisson's equation in rectangular domains*, Research Report 17-10-18, Department of Mathematics, Temple University, 2017.
- [5] F. Magoules, D.B. Szyld, C. Venet. *Asynchronous optimized Schwarz methods with and without overlap*, Numerische Mathematik, 1, 137, 199-227, 2017.
- [6] Christian Glusa, E. Boman, E. Chow, S. Rajamanickam, D. Szyld *Scalable asynchronous domain decomposition solvers*, SIAM Journal on Scientific Computing, 42(6), 384-409, 2020.
- [7] Guillaume Gbikpi-Benissan *Méthodes asynchrones de décomposition de domaine pour le calcul massivement parallèle*, Thèse de l'école CentraleSupélec, 2017.
- [8] Pierre Gosselet, Maxime Blanchard, Olivier Allix, and Guillaume Guguin. *Non-invasive global/local coupling as a Schwarz domain decomposition method : acceleration and generalization*, Advanced Modeling and Simulation in Engineering Sciences, 5(4), 2018.
- [9] Olivier Allix, Pierre Gosselet *Non Intrusive Global/Local Coupling Techniques in Solid Mechanics : An Introduction to Different Coupling Strategies and Acceleration Techniques*, book : Modeling in Engineering Using Innovative Numerical Methods for Solids and Fluids. Springer, Cham, 203-220, 2020.
- [10] Mickael Duval, Jean-Charles Passieux, Michel Salaun and Stéphane Guinard *Non-intrusive coupling : recent advances and scalable nonlinear domain decomposition.*, Archives of Computational Methods in Engineering, 1-22, 2014.
- [11] Frédéric Magoulès, Guillaume Gbikpi-Benissan *Non-intrusive coupling : recent advances and scalable nonlinear domain decomposition.*, Advances in Engineering Software, Elsevier, 116-133, 2018.
- [12] Ichitaro Yamazaki, Edmond Chow, Aurelien Bouteiller, Jack Dongarra *Performance of asynchronous optimized Schwarz with one-sided communication*, Parallel Computing, 86, 66-81, 2019.
- [13] L. Eisner, I. Koltracht, and M. Neumann. *Convergence of sequential and asynchronous nonlinear paracontractions.*, Numerische Mathematik, 62 :305, 319, 1992.
- [14] Yves Renard, Konstantinos Poullos *GetFEM : Automated FE modeling of multiphysics problems based on a generic weak form language.*, 2020.

- [15] Christophe Geuzaine, Jean-François Remacle *Gmsh : a three-dimensional nite element mesh generator with built-in pre- and post-processing facilities.*, International Journal for Numerical Methods in Engineering, 0 :1-24, 2009.
- [16] Frédéric Magoulès, Guillaume Gbikpi-Benissan *Distributed Convergence Detection Based on Global Residual Error Under Asynchronous Iterations*, IEEE transactions on parallel and distributed systems, 29, 2018.
- [17] J.C.Miellou, P.Spiteri, D.El Baz *A new stopping criterion for linear perturbed asynchronous iterations.*, Journal of Computational and Applied Mathematics 219, 471 – 483, 2008