Extension du domaine de la décomposition

Pierre Gosselet

HABILITATION À DIRIGER DES RECHERCHES
DE L'ÉCOLE NORMALE SUPÉRIEURE DE PARIS-SACLAY

Présentée par

Pierre Gosselet

Spécialité

MÉCANIQUE - GÉNIE MÉCANIQUE - GÉNIE CIVIL

Sujet du mémoire

Extension du domaine de la décomposition

Soutenue à Cachan le 10 février 2017 devant le jury composé de :

Paco Chinesta	Président
Pedro Diez	Rapporteur
Anthony Gravouil	Rapporteur
Patrick Laborde	Rapporteur
Olivier Allix	Examinateur, garant
Christian Rey	Examinateur
Martin Vohralík	Examinateur

LMT-Cachan
ENS Paris-Saclay / CNRS / Université Paris-Saclay
61 avenue du Président Wilson, F-94235 Cachan cedex, France

Table des matières

Introduction

Ce mémoire présente ma contribution au calcul de structures mécaniques complexes en vue, à terme, d'être capable d'offrir aux ingénieurs des outils fiables et rapides pour l'aide à la conception. Plus particulièrement, il montre comment les méthodes de décomposition de domaine offrent un cadre propice, non seulement au calcul et à sa vérification, mais aussi à la modélisation des structures.

Les problèmes industriels tels que ceux rencontrés en aéronautique pour la conception de structures composites présentent des caractéristiques qui rendent leur simulation numérique difficile :

— Les phénomènes en jeu ne se modélisent correctement qu'à des échelles très fines comparées à celle de la structure complète. Par exemple un méso-modèle de composite, qui correspond à l'échelle la plus grossière compatible avec une prédiction satisfaisante du délaminage, requiert une représentation séparée des plis (pas d'homogénéisation dans l'épaisseur de la plaque) dont une dimension caractéristique est la centaine de micromètres. Un empilement classique pouvant être constitué de plusieurs dizaines de plis, on comprend que la simulation d'une pièce à l'échelle du décimètre requiert une discrétisation de plusieurs centaines de millions de degrés de liberté, ce qui dépasse les capacités actuelles des solveurs directs tant appréciés par les industriels.

— La conception actuelle cherche de plus en plus à évaluer la tolérance aux défauts, c'est à dire la durée de vie après initiation de phénomènes critiques fortement non-linéaires et régulièrement instables. A l'échelle mésoscopique, le délaminage est modélisé par un endommagement qui est susceptible d'engendrer des ruines rapides. Des flambages locaux sont fréquemment observés sur les panneaux raidis composant la carlingue d'un avion et ne doivent pas en compromettre le fonctionnement. En quasi-statique, la non-linéarité entraîne alors l'ajout d'un solveur non-linéaire itératif et souvent d'un pilotage du chargement. Le nombre de systèmes linéaires à résoudre est grandement accru, d'autant plus qu'il est souvent nécessaire d'évaluer différentes configurations proches (modifications légères des matériaux, de la géométrie ou du chargement).

— Qu'elles soient fonctionnelles ou induites par l'évolution du problème, les hétérogénéités et anisotropies structurelles sont fréquentes : sandwich acier/élastomère, empilements de plis unidirectionnels ou tissés, structures minces, perte locale de raideur suite à une non-linéarité concentrée. Elles entraînent des problèmes de conditionnement et des difficultés numériques.

— L'étape de vérification, indispensable pour adapter le calcul et obtenir des estimations de qualité garantie, est coûteuse en ressources puisqu'il arrive fréquemment qu'elle prenne autant de temps que le calcul direct. Aussi les ingénieurs préfèrent s'appuyer sur des règles métiers et des coefficients de sécurité qui sont malheureusement sous-optimaux dans un contexte compétitif où le dimensionnement au plus juste est souhaité. De plus l'incertitude sur la qualité du calcul empêche d'évaluer réellement les modèles choisis, d'identifier correctement les paramètres.

— Enfin de nombreux phénomènes ont pour support naturel de modélisation des surfaces : interface de contact (avec frottement) entre les pièces d'un assemblage, interface endommageable entre les plis d'un composite. Les techniques usuelles pour leur prise en compte consistent à insérer des éléments surfaciques dédiés (éléments de contact, éléments cohésifs) et à attribuer à ces éléments un comportement. Le problème est ensuite assemblé et fréquemment des difficultés numériques apparaissent.

À ces difficultés imposées par la mécanique des problèmes à résoudre, il convient d'ajouter celles imposées par l'évolution du matériel informatique. Celle-ci contraint la conception des méthodes de calcul et leur programmation. La puissance brute des unités de calcul (liée à la précision de la gravure des processeurs) n'augmentant pratiquement plus, il est nécessaire de mettre en place des stratégies à plusieurs niveaux de parallélisme.

Un dernier problème est dû à l'inertie des pratiques industrielles et des codes de calcul commerciaux qui tardent à adopter, et cela se comprend étant donné les besoins de versatilité et de robustesse des calculs industriels, les pratiques issues de la recherche alors même que le *virtual testing* (remplacement des essais physiques par des essais numériques), qui suppose des calculs massifs, performants et garantis, est un objectif affiché de ces mêmes industriels.

Mon travail a consisté à montrer comment la décomposition du domaine étudié offrait un cadre propice à la résolution de toutes ces difficultés, comment les méthodes établies devaient être adaptées pour traiter ces problèmes et quelles possibilités nouvelles elles offraient. Si on reprend un schéma classique de calcul de structure, tel que présenté sur la figure 0.1, on voit que la décomposition intervient en amont de la résolution et de la vérification, étapes sur lesquelles elle peut impacter naturellement. En observant de plus que la mise en données est elle-même naturellement sous-structurée puisque la CAO d'un assemblage est définie par parties, on voit que la décomposition de domaine peut même intervenir au niveau de la modélisation et du maillage des problèmes.

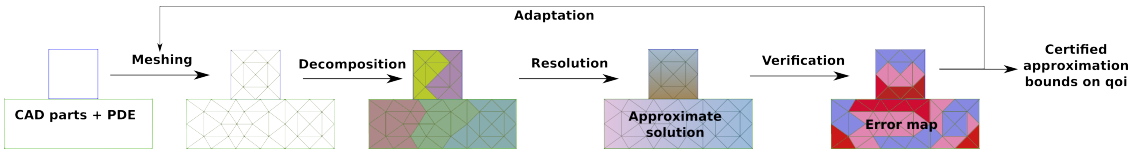


FIGURE 0.1 – Schéma d'une séquence de calcul

Le chapitre 1 montre comment la sous-structuration offre un cadre naturel pour la modélisation des phénomènes mécaniques : en représentant la structure comme un assemblage de parties liées par des interfaces, elle s'adapte à la conception ; en offrant la possibilité d'affecter un comportement aux interfaces (parfaitement collées, en contact avec ou sans frottement, en cohésion) elle offre un support pour prendre en compte les phénomènes de surface.

Le chapitre 2 montre comment la sous-structuration permet d'envisager un traitement plus efficace de la non-linéarité. Schématiquement, il s'agit de jouer sur l'imbrication des

deux boucles de calcul en jeu : celle relative à la non-linéarité (Newton, Uzawa, MAN) et celle relative à la sous-structuration (Schwarz, Schur).

Le chapitre 3 aborde la question de l'accélération des calculs. Fondamentalement les méthodes de décomposition de domaine fonctionnent quand chaque sous-domaine possède une représentation correcte de son environnement. Divers outils peuvent être utilisés à ces fins : conditions mixtes d'interface, préconditionnement, problème grossier (macroscopique).

Le chapitre 4 traite de la possibilité de vérifier les calculs réalisés par décomposition de domaine. On montre comment une technique fortement parallèle permet de diviser le coût de la vérification par un facteur très proche du nombre de processeurs. On montre aussi que la vérification permet de donner des critères objectifs pour arrêter les solveurs itératifs utilisés par la décomposition de domaine.

Le chapitre 5 montre comment des techniques de couplage non-intrusives peuvent être dérivées des méthodes de décomposition de domaine. Ces techniques permettent d'envisager une intégration facilitée des développements issus de la recherche récente dans des environnements de travail familiers des industriels. Elles peuvent être à la base de la définition de modèles de calcul par raffinements imbriqués avec prises en compte de l'ensemble des interactions possibles entre les sous-modèles et les échelles.

À la fin de chaque chapitre, je précise les travaux en cours et les perspectives à court terme. La conclusion donne des perspectives plus globales.

J'ai essayé de concentrer dans ces pages les idées essentielles qui ont guidé mes travaux et les résultats obtenus. Les détails techniques sont donnés dans les annexes constituées par une sélection d'articles publiés ou en cours de finition ; la bibliographie y est bien mieux détaillée que dans le corps de ce document.

L'intégralité de mes publications est accessible à l'adresse suivante : `https://cv.archives-ouvertes.fr/pierre-gosselet`. Dans le texte, les citations renvoyant à des articles dont je suis (co)auteur apparaissent en bleu.

1 Modélisation par décomposition de domaine

Ces travaux visent à simplifier, ou au moins à apporter un peu de souplesse, dans les étapes préalables à un calcul. En effet, dans la pratique, le *preprocessing* d'une pièce nouvelle est souvent une étape plus consommatrice en temps que le calcul lui-même. Il s'agit de vérifier la géométrie (donnée sous la forme d'une CAD), les utiliser pour générer un maillage (dont il faut s'assurer de la correction) et, dans le cas de calculs distribués, le décomposer. Ces étapes, extrêmement techniques, sont souvent sous-traités par les industriels.

L'idée de cette partie est de voir dans quelle mesure introduire la décomposition de domaine très en amont permet de simplifier le travail de l'ingénieur et de formuler des problèmes propices à une résolution efficace en parallèle et/ou sur des logiciels imposés.

Les développements méthodologiques sont détaillés dans les chapitres suivants.

1.1 Assemblage mécanique et décomposition de domaine

La géométrie d'une structure (CAO) est généralement donnée sous la forme de pièces non-recouvrantes reliées par des surfaces fonctionnelles. Il existe donc une sous-structuration naturelle que l'on cherche ici à exploiter, pour obtenir une formulation par décomposition de domaine d'un problème mécanique.

Pour spécifier un problème mécanique à partir d'une CAO, il faut attribuer des comportements aux différentes pièces et à leurs interfaces, et définir des conditions limites et des chargements.

Même si des études récentes cherchent à s'affranchir de l'étape de maillage via l'utilisation d'éléments isogéométriques [Cottrell *et al.*, 2009], y compris en décomposition de domaine [Kleiss *et al.*, 2012], la pratique industrielle consiste encore à mailler les pièces (par des éléments de volume, ou des éléments de structures comme des plaques, des coques ou des poutres) et, suivant les cas, les interfaces également. Concernant les interfaces mécaniques, on retrouve typiquement :

- des éléments mortier [Bernardi *et al.*, 1994] pour les interfaces parfaites mais avec des discrétisations non-conformes,
- des contraintes entre multiples points (MPC) [Curiskis et Valliappan, 1978],
- des éléments de contact (qui utilisent en général une formulation régularisée) [Wriggers, 2006],
- des éléments cohésifs [De Borst, 2003] pour les interfaces susceptibles de se détériorer.

Ces éléments connectés définissent un problème monolithique discret de grande taille, éventuellement non-linéaire, avec des éléments et des comportements très divers, donc des matrices *a priori* mal conditionnées. Ce système peut bien sûr être résolu en parallèle à l'aide de méthodes de décomposition de domaine telles que décrites dans les chapitres suivants.

Utiliser la décomposition de domaine pour prendre en compte le comportement des interfaces est à la base de l'approche LaTIn [Ladevèze, 1999], développée depuis de nom-

breuses années par Pierre Ladevèze. C'est un choix qui impacte fortement la chaîne de résolution, puisqu'il est complexe (quoique faisable) d'envisager d'utiliser des techniques de résolution autres que celles présentées par la suite.

L'article [Ladevèze *et al.*, 2007] décrit les ingrédients fondamentaux de la méthode (au niveau continu), ma contribution y est mineure et ma vision de la méthode a évolué depuis. La méthode, que l'on peut rapprocher de celle des directions alternées de multiplicateurs (voir [Glowinski, 2015] et la bibliographie associée), est fondamentalement mixte, puisque les déplacements et les contraintes sont recherchés simultanément, dans l'ensemble de l'espace-temps ; les volumes comme les interfaces possèdent leurs inconnues, cinématiques et statiques, et leurs équations d'équilibre et de comportement. Dans le cadre des petites perturbations, et sous réserve que le matériau admette une formulation normale, il est possible par un choix astucieux d'inconnues de séparer les équations en deux groupes :

Équations linéaires. Cet ensemble contient notamment toutes les équations différentielles en espace et en temps, les relations de trace entre quantités de volume et quantités d'interface, les lois d'état et les conditions initiales.

Équations locales. Cet ensemble contient notamment les équations non-linéaires associées à la partie dissipative du comportement (élastoviscoplasticité dans le volume, contact frottant aux interfaces, *etc*), et les conditions limites.

Sous réserve que les lois d'évolution soient associées à un opérateur maximal monotone (mécaniquement, un écrouissage positif), il est possible de montrer qu'un schéma à directions alternées permet de converger vers la solution du problème. Ce schéma conduit à résoudre successivement des problèmes linéaires à matrice constante indépendants par sous-domaine et des systèmes non-linéaires ponctuels (en espace et en temps). Notons par ailleurs que ce type de séparation d'équations et les hypothèses formulées sont à la base des travaux de vérification par l'erreur en relation de comportement [Ladevèze et Pelle, 2001] et par l'erreur en dissipation dans le cas non-linéaire [Chamoin et Ladevèze, 2007].

La formulation précédente s'associe naturellement à la *proper generalized decomposition* (anciennement approximation radiale) pour approximer efficacement les champs inconnus [Ladevèze, 1985, 1989; Chinesta *et al.*, 2009; Néron et Ladevèze, 2010; Chinesta *et al.*, 2011].

Au cours des thèses de Pierre Kerfriden et Karin Saavedra, toutes deux dirigées par Olivier Allix, j'ai appliqué la méthode au calcul à l'échelle mésoscopique du délaminage des structures composites stratifiées [Kerfriden *et al.*, 2009; Allix *et al.*, 2010b,c, 2012; Saavedra *et al.*, 2012, 2016]. Au cours de la thèse de Paul Oumaziz, dirigée par Pierre-Alain Boucard, nous nous servons également de cette méthode pour le calcul d'assemblages avec contact et frottement [Oumaziz *et al.*, 2016].

Dans [Kerfriden *et al.*, 2009], nous avons appliqué la méthode à l'étude du délaminage des plaques composites. Ces matériaux sont constitués par des empilements collés de plis unidirectionnels (l'orientation relative des plis permet l'optimisation de la structure), que l'on suppose ici élastiques linéaires orthotropes. Le mode de ruine principal de ces structures est le délaminage, qui correspond à la perte de cohésion entre des plis. Ce

phénomène est initié au niveau des bords des structures (typiquement des trous) ou des zones accidentées (par exemple par des petits chocs). Comme le délaminage est associé à une perte de rigidité de la structure, il peut se propager extrêmement rapidement et entraîner une ruine rapide de la structure. Il faut noter qu'un tel comportement, adoucissant, ne rentre pas dans les hypothèses classiques de la méthode LaTIn.

Afin de bien estimer la contrainte de pelage, il est fondamental de suffisamment mailler les plis dans l'épaisseur (typiquement une dizaine d'éléments de degré 1 dans l'épaisseur). L'originalité de notre approche est d'utiliser les interfaces entre les sous-domaines comme le lieu du délaminage (au lieu d'utiliser des éléments cohésifs). Cela a pour conséquence que chaque sous-domaine n'appartient qu'à un pli. Afin d'éviter les sous-domaines trop distordus, on se retrouve à manipuler des décompositions avec de très nombreux sous-domaines reliés par des lois potentiellement complexes (interface parfaite entre sous-domaines du même pli, interface cohésive entre les plis, contact entre les plis ruinés). Dans ce cas, la décomposition de domaine est guidée bien plus par la modélisation que par le *hardware* sur lequel le calcul sera réalisé (notamment par la mémoire disponible). Il est alors nécessaire de définir un nouveau niveau de calcul (appelé dans ces travaux super-sous-domaines) qui regroupent des sous-domaines connexes pour les distribuer sur un cluster. Cette troisième échelle a des conséquences sur le calcul qui seront détaillées dans les autres parties.

Les résultats de cette modélisation sont tout à fait satisfaisants. On arrive à reproduire des profils de délaminage classiques. L'extrême distribution des calculs au sein de petits sous-domaines peut être propice à des mises en œuvre efficaces sur des architectures de calcul hybrides, mais ce dernier point, extrêmement technique, n'a pas encore été suffisamment poussé.

Les figures 1.1, 1.2 et 1.3, issues de [Allix *et al.*, 2010a] présentent la modélisation d'une liaison boulonnée avec des interfaces de type parfaites (sous-domaines d'un même pli), cohésives (sous-domaines de plis voisins) et contact (au niveau des boulons), la sous-structuration et la super-sous-structuration, ainsi que des résultats mécaniques.

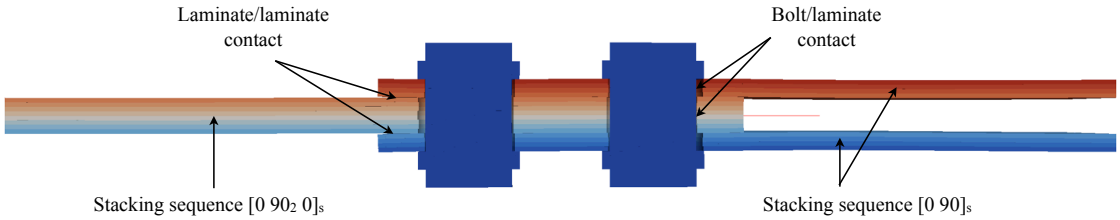


FIGURE 1.1 – Liaison boulonnée entre plaques composites (16 plis).

Dans [Saavedra *et al.*, 2012], nous avons étudié l'interaction entre flambage et délaminage dans ces mêmes composites stratifiés. Il s'agit de mettre en relation deux phénomènes extrêmement brutaux qui généralement collaborent à la ruine des structures : le délaminage entraîne une perte de raideur qui facilite le déclenchement du flambage, ce dernier pouvant fortement augmenter les contraintes transverses qui participent au délaminage.

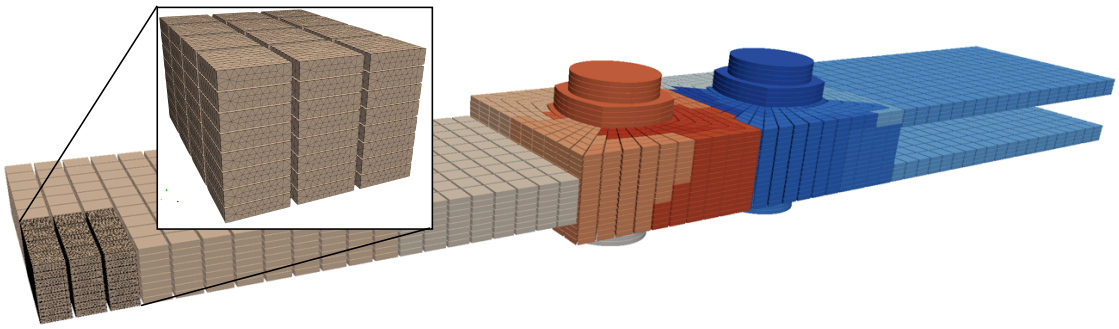


FIGURE 1.2 – Discrétisation ($12\,10^6$ ddl), décomposition en 10 600 sous-structures et assignation aux processeurs (29 CPUs)

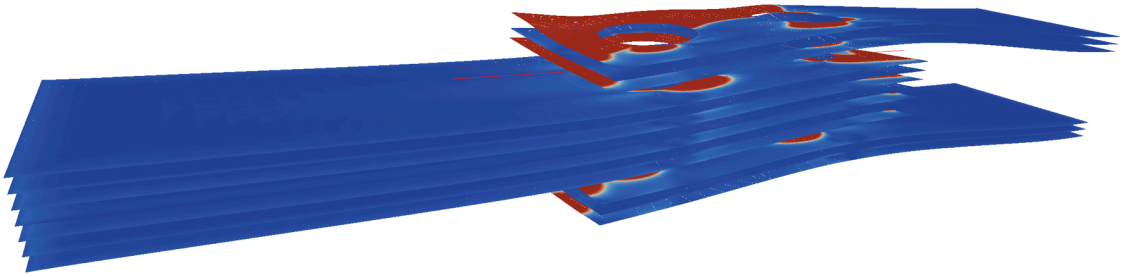


FIGURE 1.3 – Carte d'endommagement des interfaces cohésives après le $70^{\text{ème}}$ incrément.

Nous avons adopté une formulation lagrangienne totale pour la description du mouvement des sous-domaines, avec en première approximation une loi constitutive de type Kirchhoff–Saint-Venant. Nous avons supposé que la loi cohésive écrite dans le repère initial restait valable et, afin de simplifier l'étude du contact, que celui-ci n'intervenait qu'entre des nœuds initialement en vis-à-vis (pas de grand glissement). Là encore, les adaptations de la méthode de calcul seront présentées ultérieurement. Il est à noter que nous avons employé une version pragmatique de la prise en compte des grands déplacements, qui s'écarte de l'élégance de l'approche [Boucard *et al.*, 1997], ce qui nous a conduit à renoncer à la séparation entre équations linéaires et équations locales.

Afin d'illustrer les capacités de la méthode, on reproduit des résultats de [Saavedra *et al.*, 2016] auquel le lecteur est renvoyé pour une description complète du problème. Il s'agit d'une plaque lamifiée $[0°/45°/90°/-45°]_S$ trouée en compression illustrée figure 1.4. La discrétisation utilise 2,5 millions d'éléments prismatiques regroupés en 864 sous-domaines, un léger effort normal réparti permet de rompre la symétrie.

La figure 1.4 montre l'effort compressif en fonction du déplacement axial maximal. Le délaminage autour du trou commence à se propager rapidement après le premier incrément. Cela permet à un flambage local de se déclencher au $6^{\text{ème}}$ incrément, comme on peut le voir sur la figure 1.5 (gauche). Ensuite la plaque flambe en mode 1 quand la charge critique est atteinte (le front de délaminage à cet instant est représenté sur la figure 1.6). Après cela, le délaminage continue de se propager, ce qui assouplit la plaque et progressi-

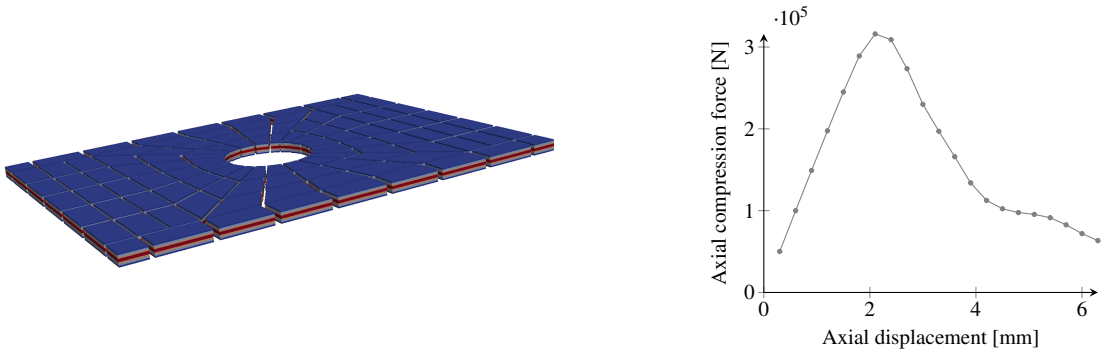


FIGURE 1.4 – Test de compression sur plaque lamifiée trouée. Partition sur domaine (gauche). Courbe effort/déplacement (droite).

vement conduit à une configuration de mode 2, visible sur la figure 1.5 (droite). Au dernier incrément (21$^{\text{ème}}$ pas), la surface délaminée représente plus de 80% des interfaces.

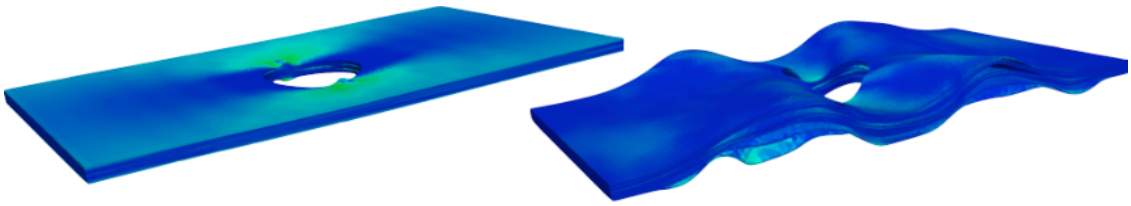
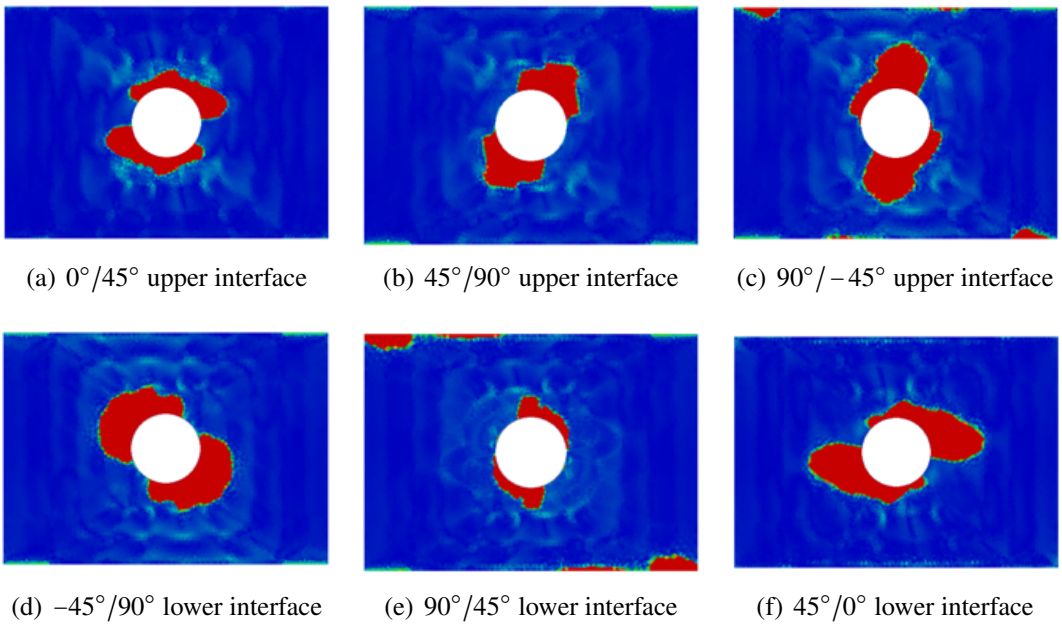


FIGURE 1.5 – Contraintes et déformations aux 6$^{\text{ème}}$ (gauche) et 21$^{\text{ème}}$ incréments (droite).

Il est donc possible d'utiliser la décomposition de domaine sans recouvrement comme le support de la modélisation d'un problème mécanique, notamment en attribuant un comportement aux interfaces. Il me semble qu'il serait particulièrement intéressant d'essayer d'appliquer cette philosophie le plus tôt possible dans la chaîne de calcul, et donc de travailler au niveau de la CAO à l'aide d'éléments isogéométriques. Par ailleurs, il serait très intéressant de reprendre la description des grandes transformations dans [Boucard *et al.*, 1997] et de la développer dans le cadre sous-structuré. Enfin, utiliser en parallèle des analyses duales, pour avoir une approche réellement mixte des problèmes non-linéaires peut sembler séduisant, en particulier si on ajoute des techniques de vérification et d'adaptation.

1.2 Modélisation par patchs recouvrants

Les techniques avec recouvrement sont *a priori* moins populaires en ingénierie, où la matière est en général attachée à un modèle et donc naturellement à un seul sous-domaine, qu'en mathématiques où la possibilité d'appliquer des techniques de stationnarité est particulièrement appréciée.



(a) $0°/45°$ upper interface (b) $45°/90°$ upper interface (c) $90°/-45°$ upper interface

(d) $-45°/90°$ lower interface (e) $90°/45°$ lower interface (f) $45°/0°$ lower interface

FIGURE 1.6 – Front de délaminage pour chaque interface (bleu = pas d'endommagement, rouge = endommagement total).

Un cas d'application est celui du zoom structural, où justement des modèles de finesse croissante se superposent afin de préciser localement une représentation globale grossière d'une structure. Cette technique est particulièrement intéressante car il est possible de l'appliquer très simplement dans des codes de calcul y compris commerciaux relativement fermés. Pour cela on parle de méthode non-intrusive, une terminologie empruntée aux stochasticiens qui cherchaient à utiliser les codes déterministes en boîte noire pour obtenir des résultats probabilistes.

Ces travaux, lancés à l'initiative d'Olivier Allix, étaient motivés par l'idée d'apporter de l'exactitude à la technique industrielle du *submodeling*. Le problème cible est illustré sur la figure 1.7 : les structures industrielles sont souvent définies par une hiérarchie de modèles imbriqués. En pratique les échanges sont limités à des messages « descendant » (du modèle global à un modèle local fin). L'objectif de nos travaux est de faire communiquer entre eux tous les modèles pour permettre la prise en compte de toutes les interactions, en particulier les transferts de charge résultant de la détérioration d'un modèle localisé.

Les modèles locaux peuvent présenter de nombreux enrichissements par rapport au global : détails topologiques ajoutés (trous) et maillage adapté, comportements matériaux complexes, ou même modèle 3D inséré dans un modèle plaque ou coque.

Ces travaux ce sont en particulier déroulés autour des thèses co-encadrées avec Olivier Allix de Lionel Gendre, pour la mise au point de la méthode, Guillaume Guguin, pour le couplage plaque/3D, et de Maxime Blanchard, sur les modèles entièrement 3D et les

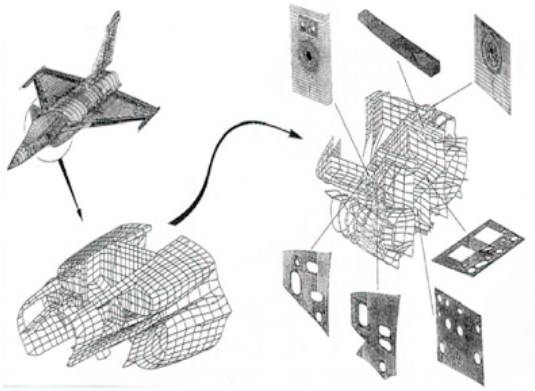


FIGURE 1.7 – Structure définie par des modèles gigognes (Dassault aviation)

études sur de nombreux pas de temps.

Dans [Gendre *et al.*, 2009], nous nous sommes intéressés à la prise en compte de la plasticité localisée au pied d'une aube de turbine. Il s'agissait de montrer à notre partenaire industriel que le *submodeling* conduisait à une mauvaise estimation de la plasticité, et que notre approche itérative était capable de corriger ce problème.

La figure 1.8 montre un modèle d'aube banalisé et la zone sur laquelle la plasticité se développe. Dans ce cas, la seule modification apportée à la zone locale est le comportement (élastoplastique au lieu d'élastique linéaire). La figure 1.9 montre que le *submodeling* sous-estime de 60% la plasticité cumulée qui est pourtant une grandeur essentielle au dimensionnement à la durée de vie. Notre approche (détaillée dans le chapitre 5), mise en oeuvre dans Abaqus, permet de converger très précisément vers la grandeur de référence.

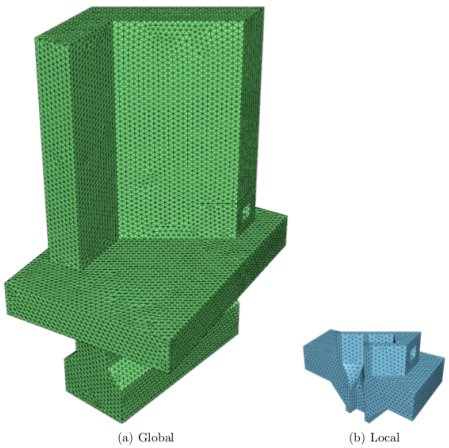
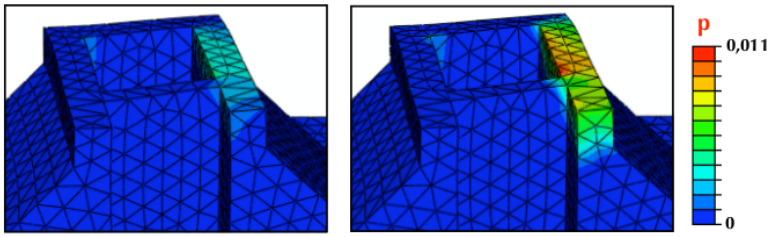


FIGURE 1.8 – Modèle d'aube banalisé et zone d'intérêt

Nous avons ensuite cherché à proposer une stratégie pour insérer des détails 3D dans des modèles de plaques stratifiées. En effet les industriels utilisent massivement des mo-



(a) Zoom descendant (max 0,0045) (b) Référence (max 0,011)

FIGURE 1.9 – Mauvaise estimation par *submodeling* de la déformation plastique cumulée.

dèles plaques ou coques pour leurs structures fines. Ceux-ci sont tout à fait valides pour calculer les grands flux d'efforts au cœur des structures, malheureusement ils sont bien moins corrects près des bords et des accidents géométriques. De plus ils sont généralement associés à des modèles connecteurs 1D très difficiles à identifier et dont le domaine de validité est assez réduit. Notre objectif était donc de venir substituer des modèles 3D partout où les modèles « simplifiées » étaient en défaut.

Pour cela nous avons donc, dans [Guguin *et al.*, 2014], proposé une technique permettant de réaliser la transition entre modèle plaque et modèle 3D. Afin de rester aussi peu intrusif que possible, notre méthode s'appuie sur un relèvement numérique des champs plaques obtenus par une étude préalable sur un volume élémentaire 3D. Par ailleurs, nous avons proposé une technique originale de recollement entre le patch et le modèle global, qui libère un peu les contraintes de maillage et limite la propagation d'effet de bords artificiels dans le modèle 3D.

Ces travaux ont été appliqués dans [Guguin *et al.*, 2016] au calcul d'une liaison boulonnée, illustrée à la figure 1.10. Au cours de cette étude où il était possible de calculer une référence entièrement 3D, nous avons pu montrer que le modèle obtenu par notre approche était bien plus précis que le modèle plaque+connecteur ou que le modèle obtenu par submodeling, sur des quantités d'intérêt locales et globales (notamment contraintes, glissement, dissipation).

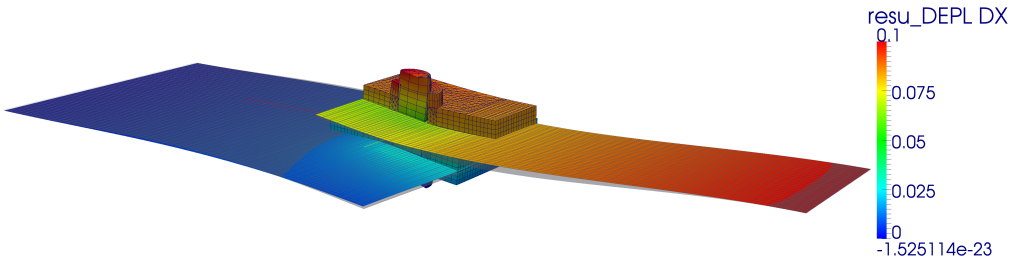


FIGURE 1.10 – Déformée du système couplé plaques/3D.

La modélisation d'une structure par une hiérarchie de modèles recouvrants est donc une alternative aux représentations monolithiques et sous-structurées. On verra au cha-

pitre 5 qu'il est possible de conduire les calculs efficacement, sans bien sûr pouvoir prétendre aux performances des méthodes dédiées au calcul haute performance (chapitre 3). Le grand intérêt de ces méthodes est de pouvoir s'insérer facilement dans les chaînes de calcul des partenaires industriels. Le projet ANR Icare, qui regroupe les partenaires industriels Airbus, EDF et Distène, et les académiques de Toulouse (ICA), Nantes (GeM) et Cachan (LMT), y est d'ailleurs consacré.

Les perspectives actuelles, concernent l'utilisation de modèles globaux non-linéaires (cas de l'élastoviscoplasticité généralisée dans les structures chaudes), et les études sur de très nombreux cycles de chargement pour le dimensionnement à la fatigue.

2 Traitement des problèmes non-linéaires

En vue d'augmenter les performances des structures conçues, il est de plus en plus fréquent de devoir les dimensionner en prenant en compte des phénomènes non-linéaires, que ce soit leur régime normal (comme une structure chaude aéronautique dont le comportement de routine est élastoviscoplastique) ou un régime exceptionnel qu'elles doivent être capables d'affronter (tolérance aux défauts ou aux incidents). Souvent la non-linéarité est associée à une petite longueur caractéristique (épaisseur du pli pour le délaminage, longueur d'onde pour le flambage) qui vient piloter la finesse du maillage (par exemple une dizaine d'éléments par longueur caractéristique) et les incréments de chargement, ce qui conduit à des modèles discrets de très grande dimension. Que la non-linéarité se diffuse ou qu'elle reste localisée, elle est, dans tous les cas, susceptible de modifier significativement l'état de la structure (redistribution des contraintes, grands déplacements suite à une rotule plastique, *etc*). Il est donc nécessaire de réaliser le calcul sur l'intégralité de la structure.

Pour référence, la technique la plus répandue en ingénierie mécanique pour traiter les problèmes non-linéaires est la méthode de Newton-Raphson. Elle consiste à résoudre une succession de problèmes tangents, en général avec un solveur direct. La parallélisation naturelle par décomposition de domaine consiste à faire calculer à chaque sous-domaine son opérateur tangent et à résoudre le système linéaire distribué par un solveur itératif tel que décrit dans le chapitre 3. On parle de méthode Newton-Schur-Krylov. Dans ce cas la seule « optimisation » possible est d'éviter aux sous-domaines restant dans leur domaine élastique de recalculer et re-factoriser leur matrice tangente ; le solveur global reste piloté par la non-linéarité même localisée. Cette technique a été employée dans [Germain *et al.*, 2007] pour le calcul de la propagation d'un endommagement non-local, et dans [Gosselet *et al.*, 2002] pour le calcul du flambage d'une butée flexible aéronautique.

On cherche dans cette partie à voir dans quelle mesure la décomposition de domaine peut aider à résoudre des problèmes non-linéaires, en essayant de traiter à l'échelle qui est la mieux adaptée les phénomènes non-linéaires. On étudie pour cela deux approches : (i) les méthodes de « condensation non-linéaire » qui sont une extension au non-linéaire des solveurs linéaires classiques FETI, BDD et leurs diverses versions ; (ii) la méthode LaTIn qui est fondamentalement un solveur non-linéaire auquel le formalisme de la décomposition de domaine a été ajouté. L'approche non-intrusive du chapitre 5 peut être également vue comme une technique de calcul non-linéaire ; néanmoins elle possède suffisamment de spécificités pour être décrite dans son propre chapitre, de plus elle ne prétend pas au calcul haute performance comme les méthodes décrites ici peuvent le faire.

2.1 Approche par « condensation non-linéaire »

L'objectif de ces travaux est de transposer la philosophie des méthodes de Schur au cas non-linéaire : on formule le problème comme la recherche du zéro d'un système d'équations (non-contractant mais généralement symétrique défini positif) et on emploie

le solveur adapté (Krylov en linéaire, Newton dans la suite de cette section).

Ces travaux ont débuté avec la thèse de Julien Pebrel et se poursuivent avec la thèse de Camille Negrello (toutes deux sous la direction de Christian Rey) [Pebrel *et al.*, 2008; Negrello *et al.*, 2016]. Ils sont apparentés aux travaux sur la relocalisation non-linéaire [Cresta *et al.*, 2007; Hinojosa *et al.*, 2014] mais avec un point de vue assez différent. Des travaux proches pour FETIDP et BDDC peuvent être trouvés dans [Klawonn *et al.*, 2014a] et dans [Dolean *et al.*, 2015a] pour la méthode de Schwarz additive restreinte.

On donne ici une brève description des travaux détaillés dans [Negrello *et al.*, 2016], que l'on peut trouver dans l'annexe C de ce document. On se place dans le cas d'un problème monolithique discrétisé. On se retrouve alors à manipuler des interfaces parfaites où les relations d'interface sont linéaires : égalité des déplacements et équilibre des efforts aux interfaces. La non-linéarité est alors contenue dans les sous-domaines, et l'outil fondamental est la condensation non-linéaire, qui permet de manipuler un sous-domaine comme une « boîte noire » interagissant via son interface.

On considère une approche primale et une décomposition en deux sous-domaines. On définit le pendant non-linéaire du complément de Schur (un opérateur Dirichlet-vers-Neumann) pour chaque sous-domaine. On peut formellement exprimer la réaction nodale λ^s d'un sous-domaine soumis à un déplacement imposé u sur son interface et à un effort donné f^s de la façon suivante :

$$\lambda^s = S^s(u; f^s) \tag{1}$$

L'existence de l'opérateur S est assurée au moins localement dans de nombreux contextes et globalement dans le cas des matériaux à écrouissage positif. L'équilibre de l'interface s'écrit $\lambda^1 + \lambda^2 = 0$. La linéarisation de ce système conduit aux itérations suivantes (indice i) :

$$\sum(\partial S^s)(u_{i+1} - u_i) = -\sum(S^s(u_i; f^s)) = -(\lambda_i^1 + \lambda_i^2) \tag{2}$$

On reconnaît dans le membre de gauche la linéarisation des opérateurs condensés. On montre qu'un tel opérateur est calculable comme la condensation des matrices tangentes par sous-domaines. Autrement dit, le membre de gauche a exactement la structure d'un opérateur tangent en sous-structuré (que l'on résout avec un préconditionnement de type BDD [Mandel, 1993; Le Tallec, 1994]).

Le membre de droite, quant-à lui, correspond au déséquilibre à l'interface entre deux sous-domaines voisins soumis à un même déplacement u_i sur leur interface ; c'est un terme non-linéaire évalué à l'aide de solveurs de Newton locaux. On a schématiquement permuté la linéarisation et la condensation, on parle donc de méthode Schur-Newton.

On se retrouve donc avec un solveur de Newton global sur le système condensé, qui encapsule un solveur tangent de type BDD et des solveurs de Newton locaux. On ajuste les précisions à l'aide de méthodes de Newton inexactes [Eisenstat et Walker, 1996; Dembo *et al.*, 1982].

L'intérêt de la méthode est de permettre à chaque sous-domaine d'évoluer non-linéairement pour un déplacement d'interface donné (on retrouverait la méthode classique si lors de l'évaluation des réactions $S^s(u_i; f^s)$ on forçait les solveurs de Newton locaux à ne faire qu'une itération). Cela permet de faire progresser plus rapidement la non-linéarité

et de limiter le nombre d'itérations globales, voire de diminuer le nombre d'incréments globaux. Ce faisant on limite le nombre de communications entre les sous-domaines, ce qui est la priorité pour la parallélisation efficace des calculs.

	Incréments	u_e	$2u_e$	$5u_e$	$5.75u_e$	$6.5u_e$
Global Newton	Primal	1	1	0.77	0.82	0.83
	Dual	1	1	1.46	2.12	3.35
	Mixte, $\mathbf{Q} = \mathbf{K}_{bb}$	1	1	0.77	0.82	0.78
	Mixte, $\mathbf{Q}$ opti	1	1	0.62	0.65	0.70
Krylov	Primal	1	1	0.76	0.82	0.82
	Dual	0.95	0.95	1.36	1.96	3.08
	Mixte, $\mathbf{Q} = \mathbf{K}_{bb}$	1	1	0.76	0.82	0.78
	Mixte, $\mathbf{Q}$ opti	1	1	0.61	0.64	0.69
Local Newton	Primal	1	2	2.62	2.88	3.04
	Dual	1	2	5.69	9.24	16.74
	Mixte, $\mathbf{Q} = \mathbf{K}_{bb}$	1	2	2.69	2.94	3
	Mixte, $\mathbf{Q}$ opti	1	2	1.85	2.12	2.43

TABLEAU 2.1 – Rapports du nombre d'itérations cumulées entre les méthodes de condensation non-linéaire et l'approche classique.

On vérifie ce fonctionnement dans le tableau 2.1 extrait de [Negrello *et al.*, 2016], pour un cas d'élastoplasticité 2D décomposé en 8 sous-domaines ; il est à noter que pour ce cas la plasticité est presque parfaite (très faible écrouissage). Le premier incrément u_e amène le domaine à la limite élastique, la non-linéarité se propage à partir des incréments suivants. Les lignes « primal » comparent l'approche décrite précédemment avec l'approche classique : on voit que pour un niveau de non-linéarité suffisant, on réalise moins d'itérations globales et donc moins d'itérations du solveur de Krylov associé au problème global tangent (donc moins de communications), au prix de quelques itérations locales (indépendantes) en plus.

Une spécificité des méthodes localement non-linéaires est qu'elles sont très sensibles au choix de l'inconnue d'interface : il est possible de dériver des versions duales (sous l'hypothèse des petites perturbations) ou mixtes de ces approches. Leurs performances peuvent varier énormément. Cela contraste avec les solveurs linéaires où il est très difficile

de justifier une préférence entre la méthode duale FETI, la méthode primale BDD (voir [Klawonn et Widlund, 2001], [Gosselet *et al.*, 2003; Gosselet et Rey, 2007] pour des études de la proximité des méthodes) et la méthode mixte FETI2LM [Roux, 2009]. Ce phénomène se visualise dans le tableau 2.1 où l'approche duale fonctionne extrêmement mal dans ce cas. Cela correspond au fait que les problèmes de Neumann sont mal posés en plasticité parfaite.

Les approches mixtes nécessitent de choisir un paramètre d'impédance aux interfaces. Cette question est développée à la section 2.3.

Parmi les perspectives de ces travaux, un problème important consiste à piloter les solveurs dans le cas de problèmes affaiblissants (flambage, délaminage,...). L'objectif est de s'assurer que la résolution n'active pas des instabilités locales non-réalistes qui la pénaliseraient. Une autre question que nous traitons actuellement est de savoir s'il n'est pas possible d'augmenter encore le nombre de résolutions non-linéaires locales en s'inspirant de techniques de composition non-linéaire [Brune *et al.*, 2015].

2.2 La méthode LaTIn

La méthode LaTIn a été initialement conçue comme une méthode de résolution pour des non-linéarités matérielles [Ladevèze, 1985]. Par un schéma à directions alternées (dites directions de recherches montantes et descendantes), elle permet de traiter ponctuellement (au sens espace-temps) les non-linéarités et de résoudre des problèmes globaux linéaires à matrice constante. Elle a été étendue aux non-linéarités géométriques [Boucard *et al.*, 1997].

Concernant la décomposition de domaine, la première adaptation de la LaTIn conduit à la version dite monoéchelle, où les calculs linéaires sont maintenant supportés indépendamment par les sous-domaines, d'où un parallélisme excellent mais une convergence lente. Les mécanismes qui conduisent à accélérer la méthode sont détaillés au chapitre 3. Par ailleurs, une partie des directions de recherche s'interprète comme une condition mixte dont il sera question à la section 2.3.

Ce paragraphe couvre essentiellement les contributions [Allix *et al.*, 2010b,c; Saavedra *et al.*, 2012] où des travaux spécifiques à la non-linéarité ont été effectués. Il s'agissait du traitement, en quasi-statique, de non-linéarités d'interface de type délaminage, couplées à un traitement des non-linéarités géométriques dans [Saavedra *et al.*, 2012]. Dans tous les cas, l'aspect « grand incrément » de la LaTIn, qui requiert l'usage de la PGD pour être réellement efficace, n'était pas utilisé. Au moment de passer les instabilités, un solveur de type RIKS était enclenché (avec contrôle local sur l'accroissement maximal d'endommagement).

Comme l'intégration des variables internes est une question cruciale, nous avons complété l'approche, dans [Allix *et al.*, 2010b], par une technique d'adaptation du pas de temps permettant de garantir l'intégration correcte de l'évolution. Le problème est illustré à la figure 2.2 pour le cas simple de DCB de la figure 2.1 : si les pas de temps sont trop espacés, l'évolution est mal estimée et des bandes d'endommagement apparaissent.

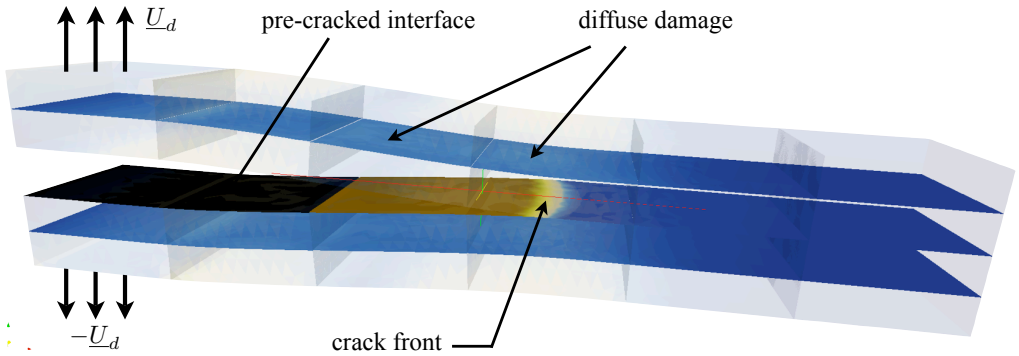


FIGURE 2.1 – Définition du problème DCB à quatre plis

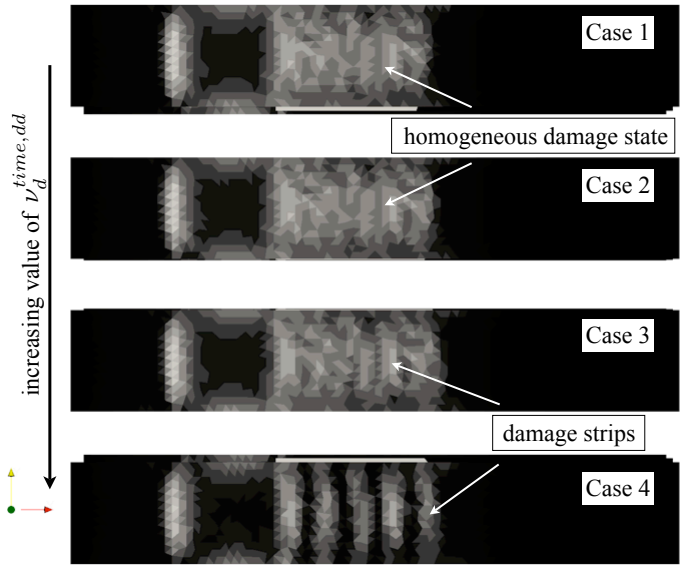


FIGURE 2.2 – Influence de la taille des incréments sur la répartition de l'endommagement sur un calcul DCB.

Dans [Allix *et al.*, 2010c], nous avons compris que le traitement de la non-linéarité par des calculs ponctuels, qui peut sembler une propriété attrayante de la méthode LaTIn, était finalement assez pénalisant pour ce genre de problèmes de propagation quasi-statique où il est indispensable de bien faire converger un état avant de pouvoir passer à l'incrément suivant : le front de délaminage doit avancer d'une seule rangée de points de Gauss à chaque incrément, et le reste de la structure est finalement peu impactée par cette simple avancée. Ce genre de problèmes a motivé les méthodes de condensation non-linéaire de la section précédente.

Nous avons donc proposé d'inclure le front de délaminage dans une « boîte de relocalisation », qui permet de calculer à une échelle intermédiaire l'évolution du front de délaminage avant « d'informer » le reste de la structure de cette évolution. Cette idée est illustrée sur la figure 2.3. Le nombre d'itérations est fortement décru comme illustré par

la figure 2.4. La taille de la boîte de relocalisation devient un paramètre de la méthode, dont l'influence est illustrée par la figure 2.5 : le raccord avec le reste de la structure se fait d'autant mieux que la boîte est grande, ce qui diminue le besoin d'itérer ; par contre le coût des calculs dans la boîte augmente.

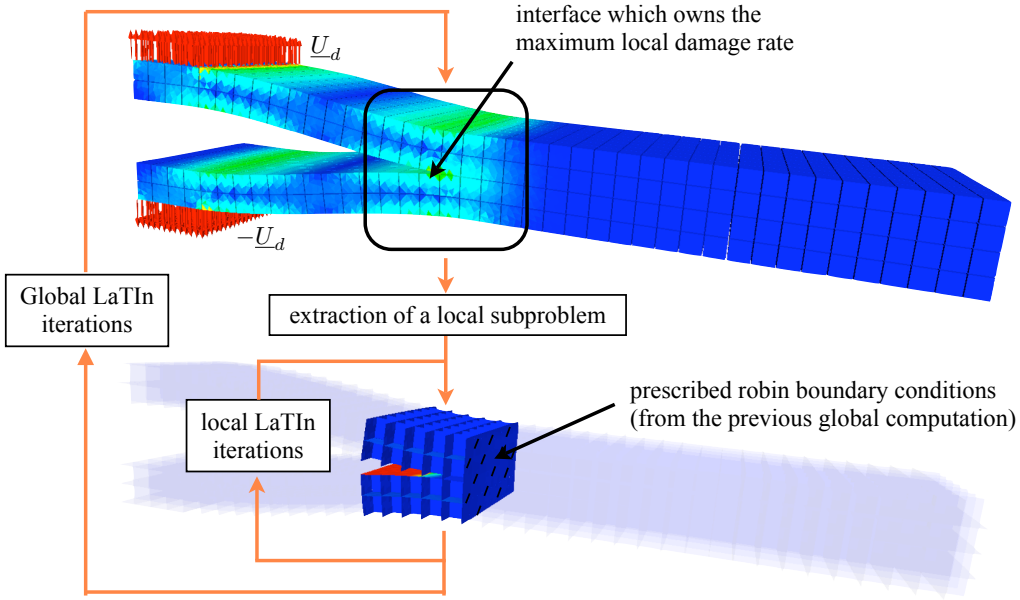


FIGURE 2.3 – Illustration de la technique de sous-résolution

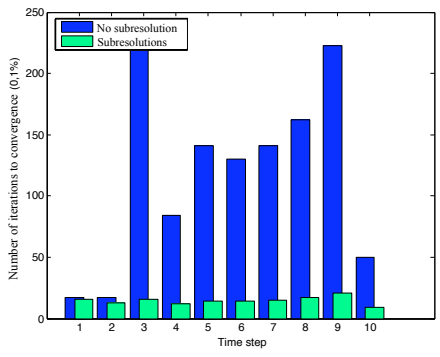


FIGURE 2.4 – Résultat des sous-résolutions

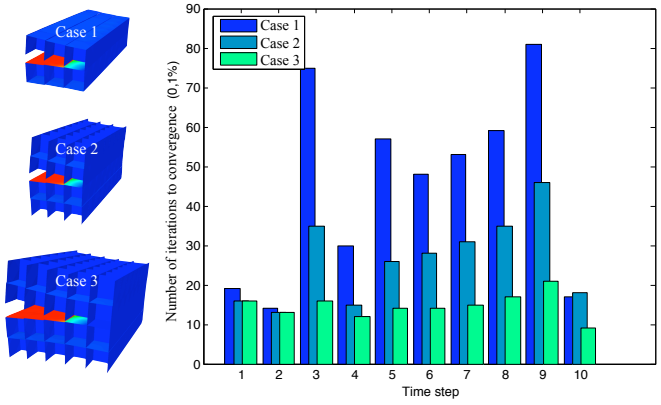


FIGURE 2.5 – Influence de la taille de la boîte

2.3 Construction d'une condition d'interface mixte

Dans la section 2.1 nous avons évoqué la possibilité d'utiliser une formulation mixte aux interfaces. Dans la méthode LaTIn (section 2.2), les directions de descente s'interprètent également comme des conditions mixtes. De fait, si le choix d'une condition

d'interface semble assez peu sensible en linéaire où des préconditionnements efficaces rendent toutes les méthodes pratiquement équivalentes, l'évolution non-linéaire des sous-domaines est bien plus sensible à la représentation qu'ils se font de leur environnement au travers des conditions limites qu'ils subissent.

La question de la représentation de l'environnement d'un sous domaine est assez classique. Elle est même au cœur des méthodes de Schwarz optimisées [Gander, 2006]. C'est une question qui généralise la recherche de la meilleure combinaison de conditions primales et duales que l'on retrouve dans [Gosselet *et al.*, 2003; Gosselet et Rey, 2007], [V. Dolean, 2009]. Il s'agit d'approcher l'opérateur Dirichlet-vers-Neumann du complémentaire du sous-domaine [Magoulès *et al.*, 2006]. Cette question fondamentale a notamment été abordée, avec différents points de vue, dans les thèses de Karin Saavedra, Geoffrey Desmeure, Lionel Gendre, Paul Oumaziz, Camille Negrello. La méthode non-intrusive, qui peut s'interpréter dans le cadre des approches mixtes, est plus précisément étudiée au chapitre 5.

On se contente dans un premier temps de chercher une approximation linéaire de cet opérateur. La façon la plus classique est d'utiliser une condition de Robin (ou Fourier), autrement dit utiliser un simple scalaire. C'est ce qui est fait classiquement dans la méthode LaTIn où le même scalaire est utilisé pour toutes les interfaces. Dans [Saavedra *et al.*, 2012], nous avons montré que ce choix était un peu trop frustre pour des structures élancées, où la forte anisotropie structurelle devait *a minima* se traduire par des scalaires différents en traction et en flexion. Dans [Saavedra *et al.*, 2016], on voit qu'il peut être particulièrement intéressant de pré-calculer une raideur équivalente sur un modèle grossier de la structure pour obtenir de bonnes performances lors du calcul.

Pour améliorer les conditions de Robin, certains utilisent des conditions de Ventcell [Hoang *et al.*, 2014]. L'usage d'un opérateur de Laplace-Beltrami permet en effet de mieux coupler les degrés de bord, approchant un peu mieux la nature pseudo-différentielle de l'opérateur de Stecklov-Poincaré. Dans [Desmeure *et al.*, 2011a,b], nous avons proposé d'améliorer la représentation de ce couplage en utilisant une condition de Robin scalaire entre le déplacement de bord et l'image par isomorphisme de Riesz dans $H^{1/2}(\partial\Omega)$ des efforts de bord (cette idée nous a été inspirée par [Bernardi *et al.*, 2008]).

Une condition de Robin s'écrit classiquement :

$$\alpha(u-u_0)+(q-q_0)=0 \qquad \text{sur } \partial\Omega \tag{3}$$

avec $(u,u_0)\in H^{1/2}(\partial\Omega)^2$ des déplacements de bord et $(q,q_0)\in H^{-1/2}(\partial\Omega)^2$ des réactions au bord (à chaque fois respectivement inconnus et donnés) et $\alpha\in\mathbb{R}$ un scalaire. Nous avons essayé d'utiliser les conditions suivantes :

$$\alpha(u-u_0)+J(q-q_0)=0 \qquad \text{sur } \partial\Omega \tag{4}$$

avec $J : H^{-1/2}\to H^{1/2}$ l'isomorphisme de Riesz. Après calibration de α, les performances obtenues étaient très satisfaisantes en terme d'itérations mais au prix du calcul de J (ou d'une bonne approximation) et au prix de conditions d'interface « pleines » qui couplent tous les degrés de liberté de bord entre eux et détruisent le profil creux des opérateurs.

Ces performances en nombre d'itérations sont illustrées sur la figure 2.6 issue de [Desmeure, 2015] où l'on montre, pour un problème académique d'élasticité 2D (carré découpé en 4×4 sous-domaines), la décroissance du résidu (mesuré en déséquilibre d'effort en saut de déplacement) pour une approche de type Robin classique (notée $q \in L^2$) et l'approche avec isomorphisme (notée $Jq \in H^{1/2}$). La distinction mono/multi échelle est discutée dans le chapitre 3, il faut noter ici le meilleur comportement de la nouvelle approche notamment en multiéchelle.

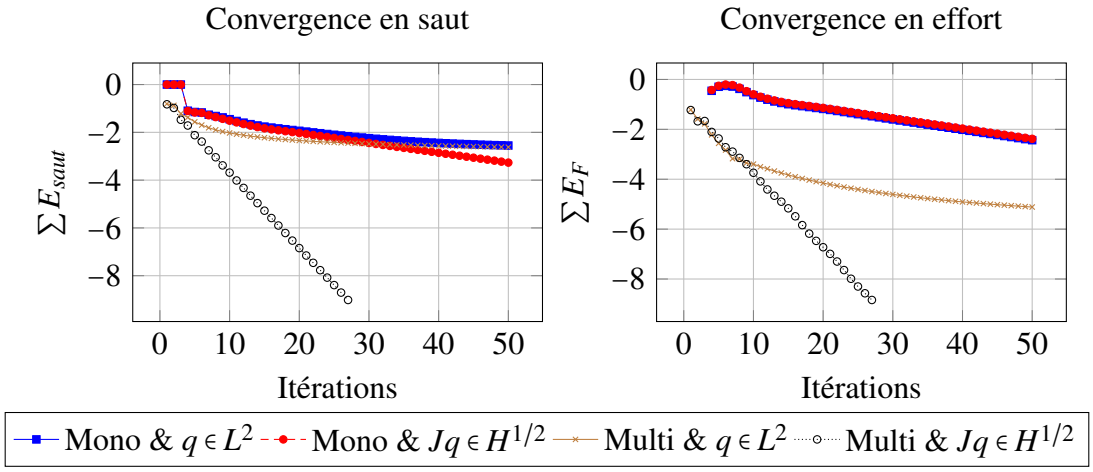


FIGURE 2.6 – Comparaison des convergences pour les versions monoéchelle et multiéchelle, chacune avec interefforts classiques et par représentation dans $H^{1/2}$.

Les conditions mixtes apparaissent également dans la méthode FETI2LM, en particulier dans sa version non-linéaire de la section 2.1. Les résultats du tableau 2.1 montrent les performances de ces deux approches mixtes par rapport à la condensation non-linéaire primale. La condition idéale serait sans doute le complément de Schur tangent du complémentaire du sous-domaine à la fin de l'incrément considéré. Nous avons testé une version quasi optimale mais incalculable sur des cas réalistes (complément de Schur tangent du complémentaire du sous-domaine au début de l'incrément considéré) et au profil trop plein, et une version réaliste basée sur la raideur du bord des voisins (à la façon d'un opérateur *lumped* de la méthode FETI). On voit que la condition optimale permet des gains significativement meilleurs que l'approche primale. L'approche *lumped* est trop raide et se distingue peu de l'approche primale.

Nous avons également proposé créer une condition mixte en ajoutant de la matière sur l'interface des sous-domaines [Oumaziz *et al.*, 2016]. La proximité avec les approches de Schwarz restreintes [Cai et Sarkis, 1999] est très grande, ici la matière dans le recouvrement n'a pas à coïncider avec les voisins, c'est un paramètre de la méthode (l'autre « bord » de la matière ajoutée est encastré). L'intérêt de cette approche est que l'on retrouve le couplage global entre les degrés de liberté de bord et que, d'un point de vue calcul, on travaille toujours sur une matrice creuse. Cette approche a permis d'implan-

ter la méthode LaTIn dans code_aster qui ne dispose pourtant pas de conditions de bord mixtes.

Nous travaillons actuellement à de nouvelles heuristiques pour le calcul de conditions mixtes efficaces, partiellement inspirées des approches multiéchelles de la LaTIn présentées au chapitre 3 et des travaux réalisés dans le cadre non-intrusif [Gendre *et al.*, 2011].

3 Calcul multiéchelle et robuste

On s'intéresse dans ce chapitre à la question de l'extensibilité et de la robustesse des méthodes de décomposition de domaine. En effet, pour qu'il soit pertinent d'utiliser une de ces méthodes, il faut qu'elle soit capable de fonctionner efficacement indépendamment du nombre de sous-domaines et de leur forme. Les découpeurs automatiques en sous-domaines fonctionnent actuellement uniquement à partir des données topologiques du maillage (connectivité entre nœuds) et non à partir de la géométrie du domaine, sans parler de la composition matérielle de celui-ci. Il est donc inévitable de travailler avec des sous-domaines irréguliers et des interfaces alignées n'importe comment par rapport aux hétérogénéités, ce qui entraîne de mauvaises propriétés numériques.

Les méthodes de décomposition de domaine reposent sur des solveurs itératifs (de type stationnarité ou de Krylov) où à chaque itération des résolutions locales indépendantes sont réalisées. Dans leurs versions les plus sommaires, les méthodes de décomposition de domaine font propager l'information d'un sous-domaine à ses voisins à chaque itération, ce qui rend la convergence fortement dépendante du diamètre du graphe de la décomposition. Il est donc nécessaire de rajouter un mécanisme de communication globale dans ces méthodes. Par analogie avec les méthodes multigrilles, cet ingrédient prend le nom de grille grossière (ou problème grossier).

Un guide fondamental concernant le problème grossier est le principe de Saint-Venant. Ce dernier spécifie qu'une distribution d'effort à torseur nul n'a pas d'effet à grande distance. Il existe donc une séparation d'échelle naturelle en effort, et le problème grossier doit au moins transmettre l'information relative aux torseurs des efforts sur la frontière d'un sous-domaine (dite information macro). L'information restante (dite micro) reste cantonnée à proximité du sous-domaine, elle dépend finement de l'allure de la distribution d'efforts et doit être trouvée par des itérations entre voisins.

Bien sûr, s'assurer que le problème grossier transfère les torseurs d'efforts au bord des sous-domaines revient à s'assurer que l'on résout des problèmes de Neumann bien posés sur les sous-domaines. Cela correspond bien aux problèmes grossiers usuels de FETI et BDD.

Le principe de Saint-Venant est illustré sur la figure 3.1. On résout un problème élastique linéaire isotrope en contrainte plane, avec condition de Dirichlet nulle sur tout le bord. On applique diverses distributions d'effort sur une frontière interne représentant le bord d'un petit sous-domaine carré et on représente la contrainte de Von Mises. Les deux premières rangées correspondent à des distributions d'effort donnant la même résultante en effort. On voit que le sous-domaine a dans ce cas une influence sur toute la structure, à vue d'œil ces effets au loin sont assez indépendants de la distribution précise de l'effort. La rangée du bas correspond à des distributions d'effort à résultante nulle, la propagation est bien plus limitée (les contraintes sont pratiquement nulles au delà de 5 fois la longueur caractéristique du sous-domaine).

Je présente dans ce chapitre mes travaux autour de la méthode LaTIn, puis ceux autour des méthodes BDD et FETI qui conduisent à des développements assez généraux autour

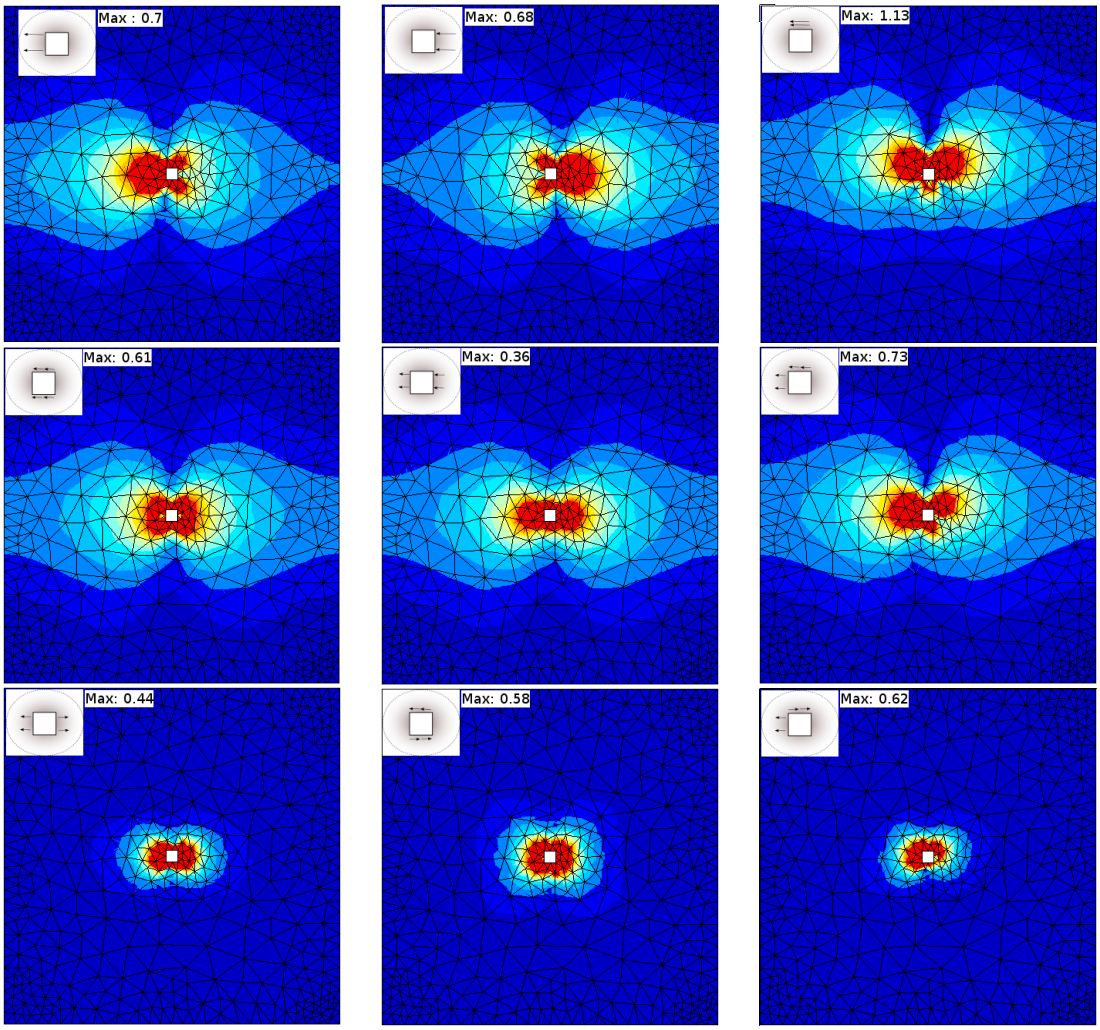


FIGURE 3.1 – Illustration du principe de Saint-Venant (la couleur correspond à la contrainte de Von Mises, même échelle), la résultante est identique (non-nulle) pour les deux premières rangées, elle est nulle sur la rangée du bas.

des solveurs de Krylov. Je termine par une courte référence à des travaux sur la recherche d'un problème macroscopique pour des techniques de Galerkin discontinues apparentées aux décompositions de domaine pour les calculs en moyenne fréquence.

3.1 Cas de la méthode LaTIn

La méthode LaTIn travaille avant tout sur des décompositions « naturelles » en sous-domaines issues de la modélisation du problème (cf. chapitre 1). On manipule donc généralement des interfaces de forme régulière (essentiellement des plans, des cylindres ou des sphères). Pour cela l'essentiel des travaux sur la méthode a porté sur des géométries simples.

3.1.1 Espace macroscopique classique

Dans la version monoéchelle de la LaTIn, c'est à dire la méthode telle que décrite jusqu'à présent, l'étape linéaire correspond à la résolution indépendamment par sous-domaine de problèmes de Robin, et l'étape locale correspond à une recombinaison des déplacements et efforts d'interface entre voisins.

L'équilibre des interfaces est une propriété locale (prise en compte à l'étape du même nom), linéaire, vérifiée par tous les comportements envisagés (liaison parfaite, contact, frottement). Il est donc possible de transférer une partie de sa vérification à l'étape linéaire, ce qui permet de transférer une information à l'ensemble de la structure.

Pour ce faire on impose au déséquilibre sur chaque interface d'être orthogonal à un sous-espace de déplacements (dits macroscopiques) donné. Afin de satisfaire le principe de Saint-Venant, l'espace macroscopique doit être tel qu'il contienne les déplacements de solide rigide de chaque sous-domaine. En pratique, il est bien plus grand puisque il contient des déplacements de solide rigide par interface. Cet espace est même complété par des déformations du premier degré, qui correspondent aux théories usuelles de l'homogénéisation, voire du second ou du troisième degré. Afin de permettre la satisfaction de cet équilibre macroscopique, la direction de recherche est modifiée.

Ces espaces macroscopiques ont été appliqués dans [Kerfriden *et al.*, 2009; Saavedra *et al.*, 2016] à des calculs de composites lamifiés tels que présentés au chapitre 1.

Plus précisément, nous avons exploité dans [Kerfriden *et al.*, 2009] la structure du problème macroscopique pour proposer de le résoudre en parallèle (en regroupant les sous-domaines au sein de super-sous-structures) par une méthode de type BDD. Ceci permet une meilleure distribution des calculs que dans les mises en œuvres précédentes. L'équilibre des super-sous-structures (vis-à-vis de leurs modes rigides) entraîne la formation d'un problème super-macroscopique qui permet de faire transiter une information globale pertinente même en très peu d'itérations : un résidu relatif de 10^{-1} semble suffire pour obtenir la convergence usuelle de la LaTIn. Ces résultats confirment que le problème macroscopique usuel de la méthode LaTIn est bien trop riche, et que l'on pourrait se contenter du haut du spectre de l'opérateur macroscopique.

Dans [Saavedra *et al.*, 2016] des tests assez systématiques ont été conduits sur l'utilité des espaces macroscopiques enrichis pour le calcul des plaques composites. La conclusion qui s'en dégage est qu'il est primordial avant toute chose de bien choisir les paramètres de Robin (directions de descente) de la méthode (voir section 2.3), et que le problème macro du premier degré suffit dans la majorité des cas (sauf sous-structures particulièrement épaisses).

3.1.2 Réinterprétation de l'espace macroscopique

Dans la thèse de Paul Oumaziz, dirigée par Pierre-Alain Boucard, nous essayons de revisiter la méthode LaTIn, de manière à pouvoir la mettre en œuvre dans un code industriel (code_aster) tout en conservant ses ingrédients principaux.

Tout d'abord, comme toutes les méthodes mixtes, la discrétisation de cette méthode pose problème : dans sa version la plus stricte, il faudrait combiner inconnues de dépla-

cement et d'effort avec les éléments finis adaptés (Lagrange vs Raviart-Thomas). Pour éviter de manipuler autant d'inconnues, on privilégie souvent les déplacements. Dans le volume cela signifie que les contraintes ne sont plus un « champ » mais uniquement définies aux points de Gauss. Sur les interfaces, il est souvent choisi de discrétiser les efforts, ce qui pose de nombreux problèmes de raccord.

Dans [Oumaziz *et al.*, 2016] (voir l'annexe B), on montre qu'il est plus naturel de travailler avec les efforts d'interface définis par dualité discrète, autrement dit de travailler avec des réactions nodales. De plus, nous montrons que les espaces macroscopiques en effort introduits *a priori* dans la majorité des articles relatifs à la méthode n'ont pas d'utilité. De tels espaces apparaissent comme l'image de l'espace en déplacement par la direction de recherche ; ils ne sont donc pas en général identiques entre voisins (sauf direction de recherche commune entre voisins).

Par ailleurs, on montre que le problème macroscopique s'interprète comme un assouplissement de rang faible k^M de la condition de Robin initiale k^m (direction de recherche monoéchelle). Schématiquement, on a :

$$k = k^m - k^M \qquad \text{avec } k^M = FQ^{-1}F^T > 0 \tag{5}$$

F est une base d'efforts et Q est une rigidité macroscopique, elles dépendent toutes deux de l'espace macro en déplacement et de la direction de recherche monoéchelle k^m. Autrement dit, si on part d'une représentation trop souple de l'environnement, rajouter un problème grossier risque d'empirer la résolution. De plus, le problème grossier ne s'appuie que sur les directions de recherche monoéchelles, il n'apporte pas d'information sur la structure réelle.

La perturbation est illustrée sur le problème modèle de la figure 3.2. On voit sur le profil de la figure 3.3 que k^M crée un couplage entre les sous-domaines ; l'étape linéaire n'est plus bloc-diagonale.

Pour résoudre l'étape linéaire de la méthode LaTIn, on applique k sur la rigidité K des sous-structures. Une façon efficace de dérouler le calcul est d'utiliser la formule de Shermann-Morrisson-Woodbury, en écrivant le problème comme la matrice monoéchelle $(K + k^m)$ soumise à une perturbation de rang faible. Ces calculs coïncident avec ceux faits usuellement. On voit que le second problème micro introduit usuellement n'est que l'apport de l'information macroscopique et ne nécessite pas de réelle résolution.

On détaille par ailleurs la possibilité d'introduire des contraintes macroscopiques sur le déplacement dans le cas d'interfaces parfaites. On voit qu'elles répondent à un mécanisme différent puisqu'elles ne semblent pas pouvoir s'interpréter comme une correction de la rigidité. Le choix de contraintes macro en déplacement pourra s'inspirer des nombreux travaux sur les méthodes FETI-DP et BDDC.

On s'interroge également sur la possibilité d'exhiber *a priori* un problème macroscopique optimisé, à l'instar des approches GENEO [Spillane et Rixen, 2013]. Le cas des méthodes de Schwarz optimisées a été traité dans [Haferssas *et al.*, 2015], le cas de la LaTIn doit en découler.

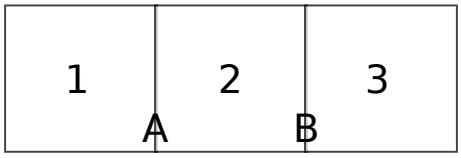


FIGURE 3.2 – Décomposition modèle

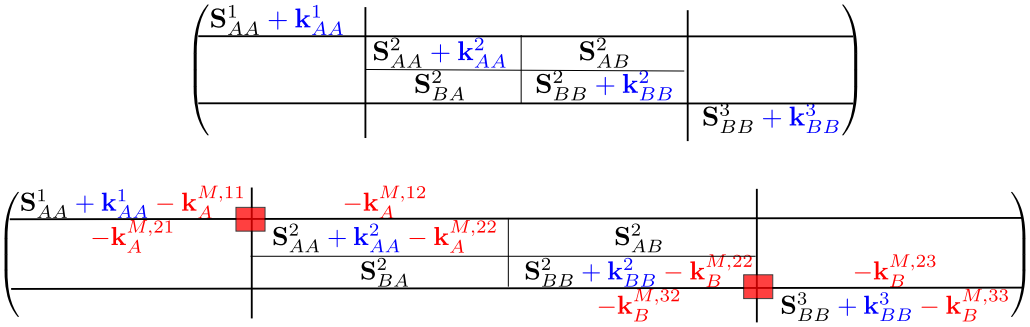


FIGURE 3.3 – Profil des matrices blocs pour la décomposition 3.2, en monoéchelle (haut) et multiéchelle (bas), les termes k^M sont positifs de rang faible.

3.2 Accélération des solveurs de Krylov

Une partie significative de mes travaux a traité des méthodes de décomposition de domaine de type Schur. Après m'être intéressé à une mise en œuvre générique de ces méthodes (primales, duales, mixtes, recondensées) [Gosselet et Rey, 2007] au sein du code Zset, j'ai travaillé à leur accélération.

C'est une question qui a fait d'importants progrès récemment. L'objectif était principalement de rendre ces méthodes robustes vis à vis des hétérogénéités, des interfaces irrégulières, de l'élancement des sous-domaines, de l'incompressibilité... Chacune de ces questions a soulevé de nombreux travaux, à chaque fois les auteurs proposaient d'améliorer des ingrédients dans le préconditionneur (notamment le *scaling*) ou dans les problèmes grossiers (ou plus globalement dans les contraintes de continuité imposées au système dans BDDC et FETI-DP) [Farhat et Rixen, 1995; Rixen et Farhat, 1999a; Klawonn et Rheinbach, 2007; Klawonn *et al.*, 2008; Dryja *et al.*, 1996; Mandel et Brezina, 1996; Dohrmann et Widlund, 2009, 2010; Beirão da Veiga *et al.*, 2014; Spillane *et al.*, 2014; Klawonn *et al.*, 2014b].

Dans [Gosselet et Rey, 2002; Gosselet *et al.*, 2013], on propose de profiter d'une succession de résolutions de systèmes linéaires « proches » pour anticiper et éliminer les difficultés de convergence : à la fin de chaque système on réalise à coût nul une recherche des valeurs propres du système aux valeurs propres généralisées formé par un opérateur et son préconditionneur. Pour ces méthodes (FETI, BDD...), on sait que le spectre est minoré par 1 et que seules posent problème les valeurs propres hautes isolées. Par un critère en stagnation, on arrive à sélectionner ces valeurs propres et on se sert des vecteurs propres associés comme augmentation pour les systèmes suivants.

Dans [Gosselet *et al.*, 2013], on applique cette idée à la résolution de problèmes non-

linéaires par un solveur Newton-Raphson et à l'exploration de plans d'expérience pour l'optimisation des composants d'un matériau composite ou d'un matériau plastique. Le tableau 3.1 résume les performances de cette approche pour le calcul de la plasticité dans une plaque trouée en traction telle que présentée à la figure 3.4. On suppose que les comportements suivent une loi normale avec un écart type de 10% et on réalise 100 tirages. On compare, pour FETI muni de différents préconditionneurs, une résolution classique (cg), la réutilisation de la totalité de l'espace de Krylov (trks) ou d'une sélection (srks), en termes de nombre moyen d'itérations par système linéaire, de taille moyenne ou maximale de l'espace d'augmentation, de temps CPU et humain (WCT). On voit que la sélection des vecteurs problématiques donne un bon compromis, en particulier pour le préconditionneur *lumped* : le nombre d'itérations est divisé par deux et le temps de calcul décroît de 36%.

Dans [Kerfriden *et al.*, 2011], on essaie d'utiliser de conserve la réduction de modèle (par la POD) et les sous-espaces de Krylov. La méthode conçue, à rapprocher de l'hyper-réduction [Ryckelynck, 2008], donne des résultats intéressants sur des problèmes complexes de fissuration. Les coûts de calcul restent néanmoins assez importants, et l'accolement des modes POD (issus d'une SVD) et des vecteurs de Ritz (issus d'un problème aux valeurs propres généralisées) n'est pas conceptuellement satisfaisant en l'état.

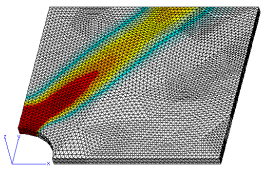


FIGURE 3.4 – Déformation plastique

precond.	method	avg. # it	avg. n_c	max n_c	avg. CPU	avg. WCT
Dirichlet	cg	25.6	–	–	1.21	3.03
Dirichlet	trks*	1.4	358	492	2.66	9.83
Dirichlet	srks14	**16.1**	17	19	**0.98**	**2.35**
lumped	cg	41.4	–	–	1.03	3.24
lumped	trks*	1.2	520	695	4.72	6.98
lumped	srks14	**19.1**	43	45	**0.87**	**2.08**

* calcul trop lent, arrêté avant la fin

TABLEAU 3.1 – Plaque trouée, résumé des performances

Récemment, il a été compris qu'il était également possible d'anticiper les modes problématiques en relocalisant le problème aux valeurs propres généralisées sur chaque sous-domaine et ses voisins [Spillane et Rixen, 2013; Dolean *et al.*, 2015b]. Il est alors proposé de résoudre initialement ces problèmes aux valeurs propres en parallèle à l'aide d'itérations de Lanczos, d'assembler un problème grossier puis de résoudre le problème initial par un gradient conjugué augmenté. C'est une approche extrêmement robuste et complètement algébrique (inutile de connaître l'équation aux dérivées partielles initiale). Un seul aspect peut sembler décevant : étant donné la proximité entre solveur de Lanczos et gradient conjugué, on se retrouve pratiquement à résoudre une première fois le système (certes dans une version distribuée) pour trouver ce qui pose problème et le résoudre une seconde fois en ayant éliminé les difficultés.

Dans [Gosselet *et al.*, 2015] nous avons proposé de généraliser la méthode S-FETI proposé par Daniel Rixen dans sa thèse. Cette méthode laisse, à chaque itération du gradient conjugué, chaque sous-domaine contribuer à la solution : on recombine optimale-

ment le bloc de directions générées. Cela revient à définir un *scaling* variable à chaque itération et donc à adapter le préconditionneur. Cette méthode s'interprète d'ailleurs comme un multipréconditionneur [Bridson et Greif, 2006]. Les performances obtenues sont extrêmement satisfaisantes, le solveur est robuste vis à vis des hétérogénéités, des découpages irréguliers et des sous-domaines élancés. De plus ces méthodes s'étendent assez facilement au cas non-symétrique [Bovet *et al.*, 2016], ce qui n'est pas encore le cas des techniques basées sur une analyse préalable.

Le coût de ces approches est néanmoins important, une campagne d'essais est actuellement menée sur un code optimisé pour évaluer jusqu'à combien de sous-domaines cette méthode reste valable (en incluant des techniques d'adaptations proposées par Nicole Spillane [Spillane, 2016]). Notre objectif est d'atteindre des performances satisfaisantes sur des clusters de taille modérée tels que l'on en rencontre chez les équipes d'ingénierie (quelques dizaines de milliers de cœurs).

En complément de l'annexe E, on illustre à la figure 3.6 l'intérêt de la méthode sur une décomposition irrégulière donnée à la figure 3.5 (haut). Il s'agit d'un cas simple de structure homogène en traction mais décomposée automatiquement, résolue avec BDD. Un sous-domaine possède une « péninsule », celle-ci est artificiellement souple et engendre à l'initialisation une contribution importante à la direction de recherche visible sur la figure 3.5 (bas) (la couleur correspond à l'amplitude du déplacement). Une combinaison classique crée une direction de recherche inadaptée, alors que recombiner les directions permet de limiter les dégâts causés par cette mauvaise géométrie.

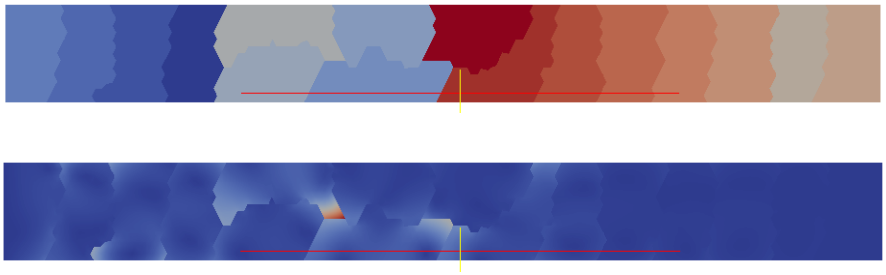


FIGURE 3.5 – Contributions locales à la direction de recherche (bas) pour une décomposition irrégulière (haut)

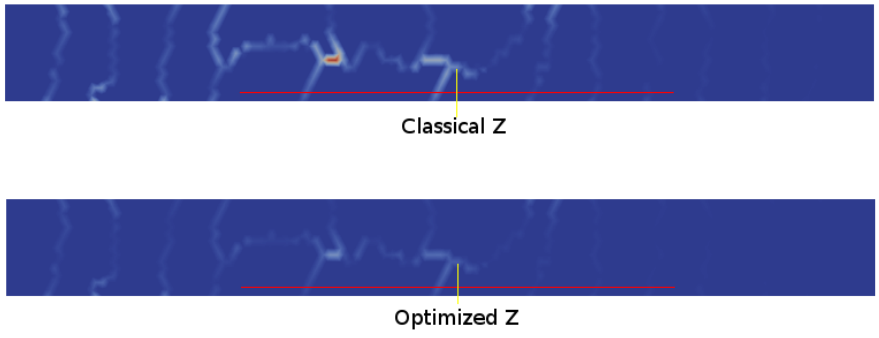


FIGURE 3.6 – Direction de recherche classique vs direction optimisée

À noter que l'idée de multiplier les directions de recherche avait déjà été exploitée pour la résolution de problèmes exhibant des motifs répétés (ou une périodicité partielle, ce qui est très fréquent en ingénierie) [Gosselet *et al.*, 2009], avec là encore des performances très satisfaisantes.

3.3 Contribution au calcul en moyenne fréquence

Enfin je termine ce chapitre en citant ma collaboration avec Louis Kovalevsky qui a conduit à la rédaction de l'article [Kovalevsky et Gosselet, 2016], où l'on propose la construction d'un problème grossier quasi-optimal pour résoudre un problème d'acoustique moyenne fréquence (équation de Helmholtz) par la TVRC [Ladevèze, 1996].

La TVRC est une méthode de Galerkin-discontinue [Di Pietro et Ern, 2012] anti-hermitienne couplée à une approximation de Trefftz indirecte [Weck, 2004; Gittelson et Hiptmair, 2014]. Dans cet article, on remonte à la source du mauvais conditionnement (la compacité de l'opérateur de Herglotz [Colton et Kress, 2001]) et par une procédure parallélisée sur les sous-domaines, on montre que l'on peut contrôler le conditionnement du problème dans un sous-espace au prix d'une légère perte de précision qui peut être comblée si le sous-espace sert d'augmentation à un solveur de Krylov.

Sur les cas étudiés, on montre qu'en se fixant l'objectif de capturer 99,99999% de l'énergie du problème, on restaure le conditionnement du problème (on passe de 10^{20} à 10^{4}) et la solution dans le sous-espace est de très bonne qualité (mesurée par le résidu). En quelques itérations d'un solveur LSQR [Paige et Saunders, 1982], on obtient une qualité supérieure à ce que donne un solveur direct sur le problème initial. De plus avec notre procédure nous stabilisons la représentation des pressions dans l'espace des amplitudes, ce qui ouvre la possibilité d'exploiter cette information (normalement extrêmement bruitée) pour l'ingénierie.

4 Vérification des calculs par décomposition de domaine

La vérification des calculs de mécanique est un point fort du laboratoire, notamment grâce aux nombreuses contributions de Pierre Ladevèze autour de l'erreur en relation de comportement, régulièrement enrichies depuis les premiers travaux [Ladevèze, 1975].

L'objectif est de donner des bornes supérieures et inférieures calculables de l'erreur $e = |||u - u_h|||$ où u est la solution exacte (inconnue) du problème mécanique, u_h une solution approchée (typiquement obtenue par élément fini), et $|||\cdot|||$ est une norme bien choisie. On peut également s'intéresser à l'erreur sur des quantités d'intérêt, comme par exemple la contrainte moyenne sur une zone ou un facteur d'intensité de contrainte en pointe de fissure [Panetier *et al.*, 2009].

S'il existe de nombreuses heuristiques permettant d'obtenir à moindre coût des estimations efficaces de l'erreur, notamment à l'aide d'estimateur en régularité de contraintes [Zienkiewicz et Zhu, 1987; Zhu et Zienkiewicz, 1988], les bornes rigoureuses s'appuient sur la construction de champs duaux qui satisfont les équations d'équilibres. Suivant les communautés, on parle de champs statiquement admissibles ou de résidus équilibrés. Outre un calcul dual direct (à l'aide d'une formulation dédiée et d'un élément fini adapté à H_{div}), ces champs équilibrés peuvent être obtenus par un post-processing de la solution élément fini en déplacement.

Le post-processing de champ de contrainte admissible implique d'assurer l'équilibre des tractions entre éléments et l'équilibre interne des éléments. Traditionnellement, le premier point se traite à l'échelle d'un patch d'éléments partageant un noeud (*star-patch*) et le second se traite par des calculs raffinés locaux sur le même patch [Parés *et al.*, 2006; Cottereau *et al.*, 2009; Parés *et al.*, 2009] ou sur les éléments [Ladevèze et Leguillon, 1983; Ladevèze *et al.*, 2012b]. Bien que ces calculs aient des supports limités, leur grand nombre, associé à la structure assez peu naturelle du *star-patch* (du point de vue logiciel), conduit à des coûts de calcul non-négligeables, de l'ordre de ceux du calcul direct qui a donné l'approximation dont on souhaite vérifier la qualité.

La première partie de mes travaux a donc consisté à montrer qu'il était possible, dans un calcul par décomposition de domaine, de paralléliser l'étape du calcul de champ statiquement admissible. Ceci a permis d'obtenir une première borne supérieure de l'erreur. Ensuite les travaux se sont portés sur l'amélioration de la borne. Ces études se sont développées au cours des thèses d'Augustin Parret-Fréaud et Valentine Rey dirigées par Christian Rey,

4.1 Reconstruction parallèle de champs admissibles

Nous avons étudié dans quelle mesure il était possible de reconstruire des champs globalement admissibles à partir des champs calculés naturellement au cours d'une méthode de décomposition de domaine. Plus précisément, il s'agit de construire un champ de déplacement $\hat{u} \in \mathrm{KA}(\Omega)$ et un champ de contrainte $\hat{\sigma} \in \mathrm{SA}(\Omega)$ où Ω est le domaine global d'étude, $\mathrm{KA}(\Omega)$ est l'espace des champs ayant la régularité $H^1(\Omega)$ et satisfaisant les conditions limites de Dirichlet et $\mathrm{SA}(\Omega)$ est l'espace des contraintes satisfaisant

l'équilibre au sens faible, c'est un sous-espace de $H_{div(\Omega)}$.

Pour une partition de Ω en sous-domaines non-recouvrants (Ω^s), les propriétés classiques des espaces cités montrent que l'admissibilité globale (sur Ω) est équivalente à l'admissibilité locale (sur chacun des Ω^s) sous réserve de continuité des traces et des flux normaux sur les interfaces.

Parmi les méthodes de décomposition de domaine, la construction de tels champs est particulièrement simple pour BDD [Mandel, 1993; Le Tallec, 1994] et pour FETI [Farhat et Roux, 1994]. En fait, il s'agit d'exploiter l'analyse faite par Daniel Rixen sur le rôle du *scaling* dans les préconditioneurs [Rixen et Farhat, 1999b] dans un contexte différent.

Fondamentalement, il s'agit d'interpréter BDD et FETI comme la recherche d'un point fixe où l'on alterne résolution de Dirichlet à déplacement (continu) imposé sur l'interface et résolution de Neumann à effort (équilibré) imposé sur les interfaces. La subtilité est que ce point fixe n'est pas contractant, un solveur de Krylov doit être utilisé. Suivant l'approche (BDD ou FETI) les champs continus ou équilibrés ne sont pas tous directement accessibles mais ils peuvent tous être reconstruits à coût nul. Cela est illustré sur la figure 4.1 pour la méthode primale (BDD).

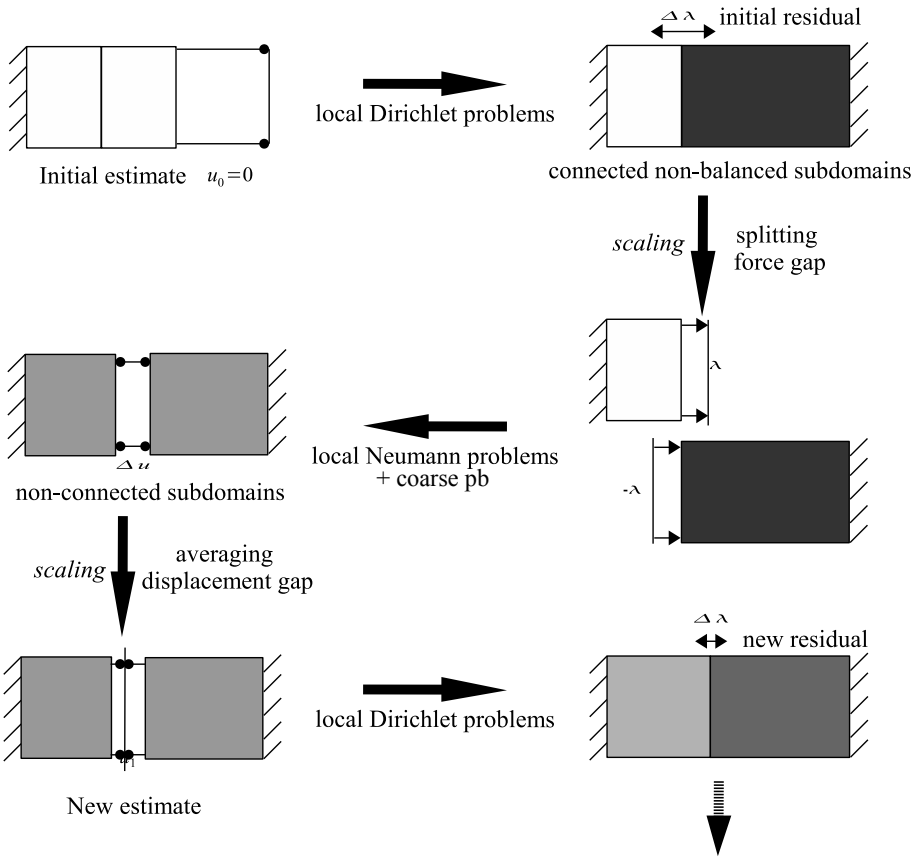


FIGURE 4.1 – Illustration de la méthode BDD par un point fixe (non-contractant), en réalité un solveur de Krylov est utilisé. La couleur correspond à l'état de contrainte.

En pratique, il faut prendre en compte le fait que les efforts d'interface ne sont connus qu'au travers des réactions nodales. Afin d'appliquer sur chaque sous-domaine une technique de reconstruction séquentielle de champ admissible, on intercale une phase de reconstruction d'un champ continu d'intereffort générant le même travail. Tous les détails, y compris la prise en compte des sous-structures flottantes, sont donnés dans [Parret-Fréaud *et al.*, 2010]. Malheureusement, cet article restait imprécis sur les précautions à prendre en présence de points multiples, cela a été amélioré dans [Parret-Fréaud *et al.*, 2016] où l'on montre qu'il est nécessaire d'effectuer un échange entre voisins partageant un point multiple avant de pouvoir reconstruire des champs équilibrés. Ce résultat permet d'étendre les résultats de nos études aux méthodes avec contraintes primales FETI-DP [Farhat *et al.*, 2001] et BDDC [Dohrmann, 2003]. À l'occasion de cet article, on montre également qu'il est possible d'améliorer le champ statiquement admissible en fonction des différentes rigidités en présence.

Pour conclure cette section, nous avons montré que pour une famille de méthodes de décomposition de domaine (BDD, FETI, BDDC, FETIDP), il était possible à tout instant de calculer sans coût supplémentaire les champs suivants :

— u_D un champ de déplacement admissible sur l'ensemble de la structure, et en particulier continu aux interfaces ;

— u_N un champ de déplacement dont la contrainte associée σ_N est en équilibre (au sens élément fini) sur l'ensemble de la structure.

σ_N peut être utilisée comme donnée d'entrée à un algorithme de reconstruction de champ équilibré $\hat{\sigma}_N$ [Ladevèze et Leguillon, 1983; Ladevèze *et al.*, 1993; Parés *et al.*, 2006; Pled *et al.*, 2011] exécuté indépendamment sur chacun des sous-domaines.

À cet occasion, notons une contribution personnelle au calcul de champs admissibles en séquentiel : la méthode STARFLEET [Rey *et al.*, 2014b] qui revisite la méthode d'équilibrage par élément [Ladevèze et Leguillon, 1983], en cherchant une mise en œuvre efficace et en se laissant l'opportunité d'optimiser le champ reconstruit.

Les principales questions ouvertes concernant la reconstruction de champs en parallèle concernent l'extension des méthodes précédentes à d'autres méthodes de décomposition de domaine comme les approches hybrides primales-duales [Gosselet et Rey, 2007] ou les méthodes mixtes (Schwarz-optimisées).

4.2 Bornes sur l'erreur globale

À partir des champs reconstruits dans la section précédente, en utilisant directement les résultats sur l'erreur en relation de comportement [Ladevèze, 1975; Coffignal et Ladevèze, 1983], on obtient, dans le cadre de l'élasticité linéaire en petites perturbations, une borne de la forme [Parret-Fréaud *et al.*, 2010] :

$$|||u - u_D||| \leqslant \sqrt{\sum_s e_{cr}^2(u_D^s, \hat{\sigma}_N^s)} \tag{6}$$

où la norme utilisée est celle de l'énergie, et où e_{cr} est l'erreur en relation de comportement (calculable) :

$$e_{cr}^2(u_D^s, \hat{\sigma}_N^s) = \int_{\Omega^s} (\mathbb{H} : \varepsilon(u_D^s) - \sigma_N^s) : \mathbb{H}^{-1} : (\mathbb{H} : \varepsilon(u_D^s) - \sigma_N^s)\, dx$$

avec $\mathbb{H}$ le tenseur de Hooke et $\varepsilon(u)$ la partie symétrique du gradient de u.

Cette borne est calculable (en parallèle) à chaque instant de la résolution du problème par décomposition de domaine, néanmoins son coût de calcul est loin d'être négligeable. On a montré dans [Parret-Fréaud *et al.*, 2010] qu'à convergence du solveur, elle était très peu sensible à la sous-structuration et donnait une borne équivalente à celle obtenue à partir d'un calcul séquentiel. On a même observé des cas de structures hétérogènes où la reconstruction parallèle donnait une meilleure estimation que la séquentielle. Cela a conduit à optimiser les reconstructions parallèles et séquentielles dans [Parret-Fréaud *et al.*, 2016].

Par ailleurs cette borne mélange les contributions de la discrétisation et celles du solveur : au cours des itérations la contribution du solveur décroît et la borne stagne à un niveau égal à l'estimation qui aurait été calculée en séquentiel. Nous avons cherché à améliorer cette borne et, en nous inspirons de [Jiránek *et al.*, 2010; Vohralík, 2007], obtenu des bornes où la contribution du solveur est mise en exergue [Rey *et al.*, 2014a, 2015] :

$$\begin{aligned} |||u - u_D||| &\leqslant \sqrt{\sum_s e_{cr}^2(u_N^s, \hat{\sigma}_N^s)} + |r| \\ |||u - u_N||| &\leqslant \sqrt{\sum_s e_{cr}^2(u_N^s, \hat{\sigma}_N^s)} + |r| \end{aligned} \tag{7}$$

où $|r|$ est la norme du résidu du solveur au sens du préconditionneur, c'est un terme naturellement calculé au cours des itérations que l'on peut rendre arbitrairement petit en laissant le solveur itérer. Les expériences numériques ont montré que le terme en erreur en relation de comportement était très majoritairement causé par la discrétisation et qu'il était très peu sensible à la convergence du solveur : il est très rare qu'il évolue à partir de la deuxième itération. De plus les contributions locales à l'erreur en relation de comportement, qui correspondent à l'intégration de l'erreur sur chacun des éléments et qui fournissent un guide utile à l'adaptation du maillage, stagnent rapidement. Ces résultats permettent d'arrêter le solveur dès que l'erreur en résidu devient négligeable par rapport à l'erreur de discrétisation estimée lors des premières itérations.

On illustre ces résultats à partir des résultats de [Rey *et al.*, 2014a]. On considère le cas classique (en vérification) de la structure pré-fissurée de la figure 4.2 décomposée en 16 sous-domaines. Le graphe 4.3 montre l'évolution des bornes supérieures de l'erreur par rapport à l'erreur séquentielle, ainsi que l'évolution de la contribution de chacun des termes. On observe bien la stagnation de l'erreur de discrétisation dès la deuxième itération. La contribution du solveur est négligeable dès la $7^{\text{ème}}$ itération. On montre figure 4.4 que les contributions locales ont également convergé en les comparant à celles calculées à la $21^{\text{ème}}$ itération, qui aurait été celle atteinte en utilisant un critère usuel sur le résidu relatif.

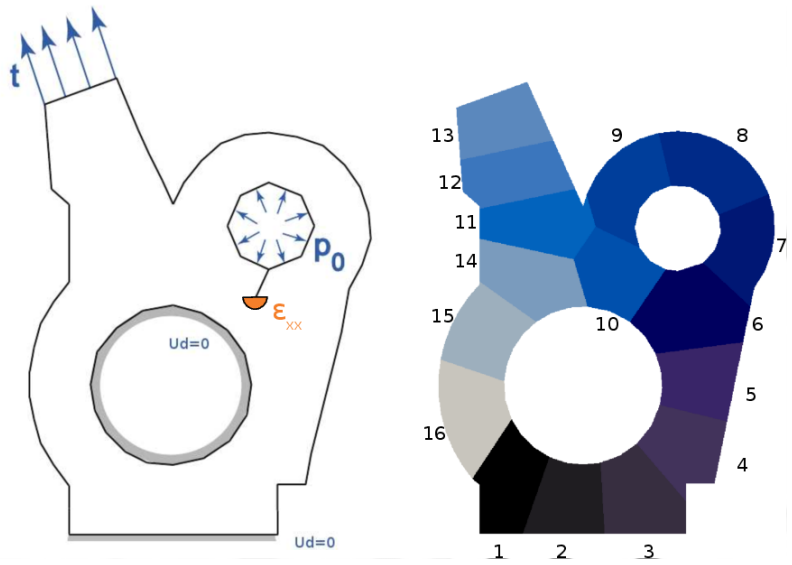


FIGURE 4.2 – Structure pré-fissurée : chargement direct (bleu) et adjoint (orange), décomposition en 16 sous-domaines

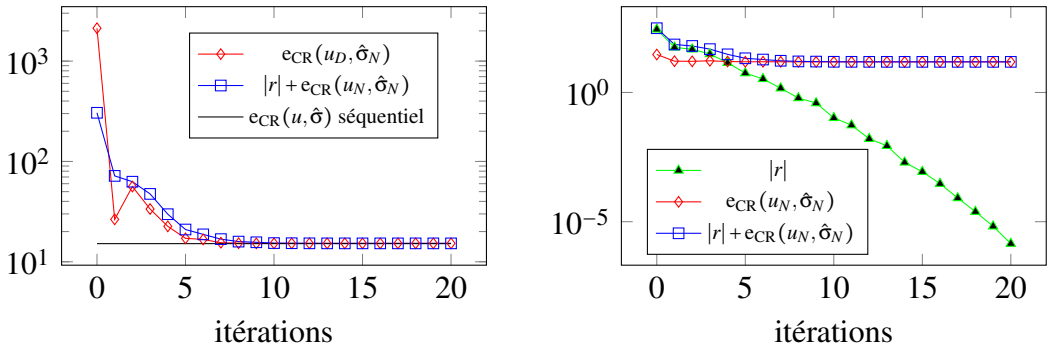


FIGURE 4.3 – Évolution des estimateurs au cours des itérations.

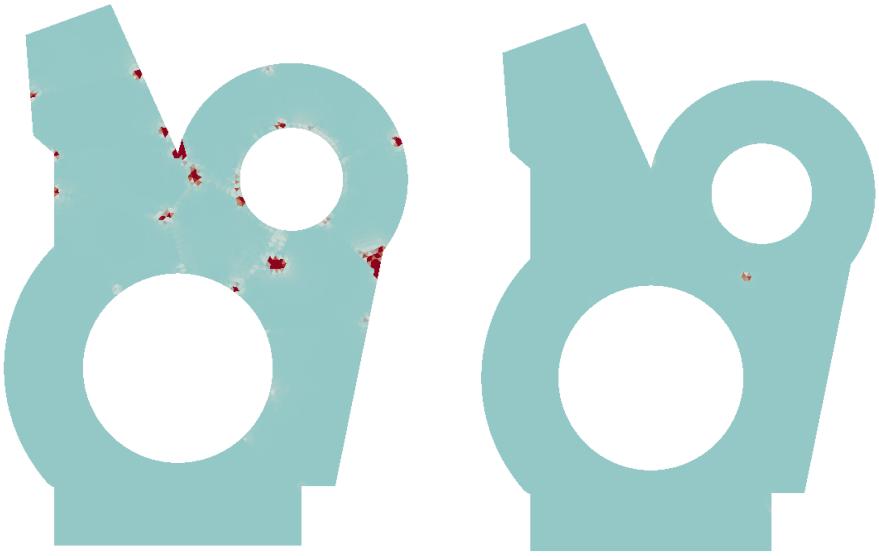
Ces résultats sur l'arrêt du solveur itératif ont pu être précisé dans [Rey *et al.*, 2016] par l'obtention d'une borne inférieure en étendant au cadre sous-structuré le résultat fondamental de [Prudhomme et Oden, 1999] :

$$|||u-u_h||| \geqslant \frac{R_h(w)}{|||w|||}, \ \forall w \in \mathrm{KA}^0(\Omega) \setminus \{0\} \tag{8}$$

où KA^0 est l'espace vectoriel associé à KA (sous-espace des déplacements nuls sur le bord de Dirichlet) et $R_h(w)$ est le résidu du calcul élément fini, facilement calculable et nul pour tout w dans la discrétisation élément fini.

Pour la décomposition de domaine, cela se traduit par la borne suivante :

$$|||u-u_D||| \geqslant \frac{R_N(w)}{|||w|||} - |r| \tag{9}$$



(a) $'e_{CR_\Omega}(\underline{\hat{u}}_N, \underline{\underline{\hat{\sigma}}})_{it=1} - e_{CR_\Omega}(\underline{\hat{u}}_N, \underline{\underline{\hat{\sigma}}})_{it=20}'$ (b) $'e_{CR_\Omega}(\underline{\hat{u}}_N, \underline{\underline{\hat{\sigma}}})_{it=7} - e_{CR_\Omega}(\underline{\hat{u}}_N, \underline{\underline{\hat{\sigma}}})_{it=20}'$

FIGURE 4.4 – Convergence des contributions élémentaires à l'estimation d'erreur

où R_N est le résidu du calcul élément fini pour un problème de Neumann. Une borne plus précise, mais plus longue à détailler, est donnée dans [Rey *et al.*, 2016]. Dans ce même papier, on montre qu'un choix particulièrement simple pour w, inspiré de [Díez *et al.*, 2003], permet d'obtenir des bornes inférieures de qualité. Ces résultats sont illustrés figure 4.5 où θ_{discr} et ρ_{discr} représentent les termes en discrétisation dans les bornes supérieures et inférieures. On observe l'évolution comparée de ces termes par rapport au résidu au cours des itérations. On voit que dès la quatrième itération les bornes encadrant l'erreur ne vont pratiquement plus évoluer, ce qui représente encore un gain d'itérations par rapport à l'étude de la borne supérieure seule.

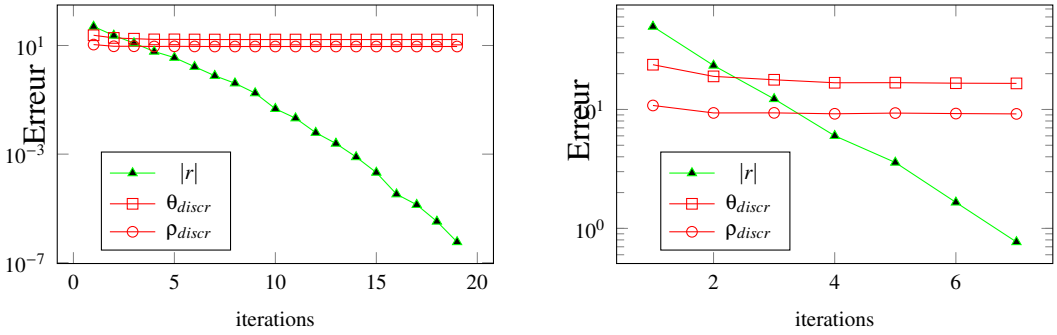


FIGURE 4.5 – Enveloppe de l'erreur due à la discrétisation et évolution du résidu

Les perspectives de ces travaux concernent essentiellement l'étude des problèmes non-linéaires et l'obtention de bornes à contributions séparées pour l'erreur en dissipation

[Ladevèze, 2006; Chamoin et Ladevèze, 2007].

4.3 Bornes sur des quantités d'intérêt linéaires

Connaître une borne d'erreur globale en énergie n'est pas d'un grand intérêt pratique pour l'ingénieur. En effet, il semble plus utile au dimensionnement de connaître avec précision certaines grandeurs comme la contrainte maximale, le déplacement en certains points, les facteurs de concentration de contrainte autour d'une fissure... Même si l'erreur globale contrôle toutes les autres (au travers de constantes difficiles à estimer), il n'est pas raisonnable de chercher à la décroître à l'excès en raffinant globalement le maillage.

Dans le cas où la quantité d'intérêt est une fonction linéaire du déplacement inconnu, une stratégie efficace pour la calculer avec ses bornes d'erreur est d'utiliser un problème adjoint [Becker et Rannacher, 1996]. Le problème adjoint (noté avec des tildes) diffère du problème direct par le second membre qui correspond à la forme linéaire permettant d'extraire la quantité d'intérêt. On dispose alors de bornes faisant intervenir l'estimation d'erreur sur le problème direct et sur le problème adjoint [Ladevèze, 2006, 2008; Prudhomme et Oden, 1999].

S'il est possible de résoudre le problème adjoint sur un maillage différent de celui du problème direct, et donc adapté à la quantité d'intérêt, il nous a semblé plus pertinent d'un point de vue calcul d'utiliser un seul maillage [Rey *et al.*, 2015]. Dans ce cas il est possible de résoudre les deux problèmes simultanément à l'aide d'un solveur Krylov par bloc [Saad, 2000]. On choisit d'arrêter le solveur quand le résidu du problème direct et celui du problème adjoint sont négligeables devant les erreurs de discrétisation associées estimées lors de la première itération. On calcule alors les bornes supérieures et inférieures de l'erreur qui permettent de préciser l'estimation de la quantité d'intérêt [Rey *et al.*, 2016].

Pour illustrer les résultats, on reprend le problème de la structure pré-fissurée 4.2, et on considère comme quantité d'intérêt une contrainte moyenne en bout de fissure (ce calcul est censé être représentatif d'un calcul de facteur d'intensité de contrainte, voir [Panetier *et al.*, 2010] pour un traitement précis de cette question).

Le tableau 4.1 (extrait de [Rey *et al.*, 2016]) donne l'évolution des résidus direct et adjoint au cours des itérations, ainsi que l'estimation des erreurs de discrétisation (sup et inf) calculée à la première itération et recalculée à la sixième itération. La table 4.2 donne l'estimation d'erreur sur la quantité d'intérêt, la seconde ligne exploite les bornes inférieures pour augmenter la précision de l'estimation. On voit qu'*in fine* on peut garantir une estimation à 22% près.

De la même manière que pour la section précédente, la perspective la plus directe de ces travaux concerne la prise en compte de quantités d'intérêt pour des problèmes non-linéaires, il s'agit de voir dans quelle mesure la résolution du problème miroir peut s'entremêler avec celle du problème direct [Ladevèze *et al.*, 2012a].

Iteration	θ_{discr}	ρ_{discr}	$\widetilde{\theta}_{discr}$	$\widetilde{\rho}_{discr}$	$\lvert r\rvert$	$\lvert\widetilde{r}\rvert$
0					260.07	0.26041
1	11.45	7.0008	0.12867	$1.5457\ 10^{-3}$	48.339	0.14931
2					35.991	$8.5664\ 10^{-2}$
3					16.649	$4.8205\ 10^{-2}$
4					5.5966	$1.2462\ 10^{-2}$
5					3.5765	$8.4996\ 10^{-3}$
6	9.9004	6.7105	0.12682	$1.5893\ 10^{-3}$	0.94805	$1.5156\ 10^{-3}$

TABLEAU 4.1 – Structure pré-fissurée : bornes et résidus des problèmes direct et adjoint au cours des itérations

	I_{ex}^{+}	I_{ex}^{-}	width	precision
Intervalle de base	3.8347	2.5791	1.2555	39.852 %
Intervalle optimisé	3.5825	2.8824	0.7001	22.224 %

TABLEAU 4.2 – Structure pré-fissurée : bornes sur la quantité d'intérêt

4.4 Adaptation de maillage

Si on imagine qu'à l'avenir un ingénieur spécifiera un problème en donnant des objectifs de précision sur des quantités d'intérêt, le résultat sera obtenu par une succession de calculs/estimations d'erreur/remaillages adaptatifs. Il faut donc s'interroger sur comment la décomposition de domaine peut intégrer l'adaptation de maillage et la résolution d'une succession de calculs basés sur la même équation.

Le traitement des maillages est une spécialité en soi, en particulier quand il s'agit d'optimiser le maillage en fonction de plusieurs objectifs ou de l'adapter en parallèle [Verfürth, 1996; Díez et Huerta, 1999; Oden et Prudhomme, 2001; Bellenger et Coorevits, 2005; Díez *et al.*, 2007]. Nous avons voulu tester des stratégies simples qui conservent une sous-structuration constante de façon à pouvoir réutiliser l'information des calculs précédents pour accélérer le calcul courant par augmentation du solveur de Krylov (à l'instar de la section 3).

Dans le cas où la contribution à l'erreur est concentrée dans quelques sous-domaines, nous avons proposé de ne raffiner que ces sous-domaines en autorisant le mailleur à rajouter des nœuds sur le bord des sous-domaines raffinés [Rey *et al.*, 2015]. Un raccord par interpolation sur l'interface est utilisé ; nous avons vérifié numériquement que l'incompatibilité n'est pas dommageable sauf dans le cas d'une quantité d'intérêt extrêmement proche de l'interface.

Cela conduit à des résultats tels que ceux extraits de [Rey *et al.*, 2016] : après un premier calcul sur un maillage grossier (on travaille toujours sur la structure pré-fissurée décomposée en 16 sous-domaines), on affiche la contribution à l'erreur par sous-domaine, figure 4.6. On décide dans un premier temps de réduire l'erreur directe, comme celle-ci est relativement bien distribuée sur la structure on utilise un raffinement adaptatif. À la suite

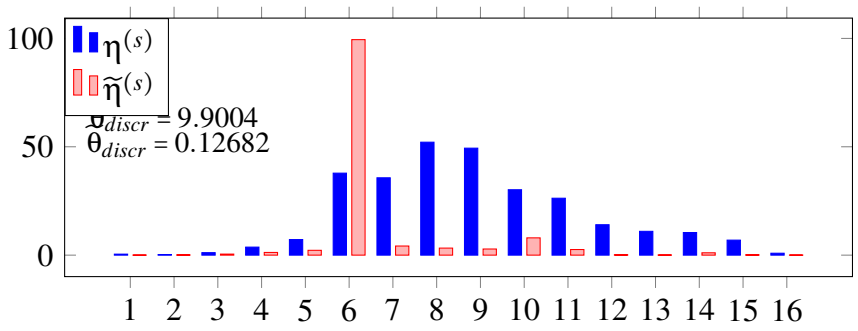


FIGURE 4.6 – Distribution des estimateurs d'erreur dans les sous-domaines

du deuxième calcul, on raffine spécifiquement le sous-domaine numéro 6 qui concentre la quantité d'intérêt (et donc l'erreur adjointe). La figure 4.7 montre les itérations des trois systèmes et les bornes calculées sur les erreurs. On arrête le calcul dès que les deux résidus sont plus faibles que leur borne inférieure. La figure 4.8 montre l'évolution de l'estimation de la quantité d'intérêt et de son intervalle d'erreur.

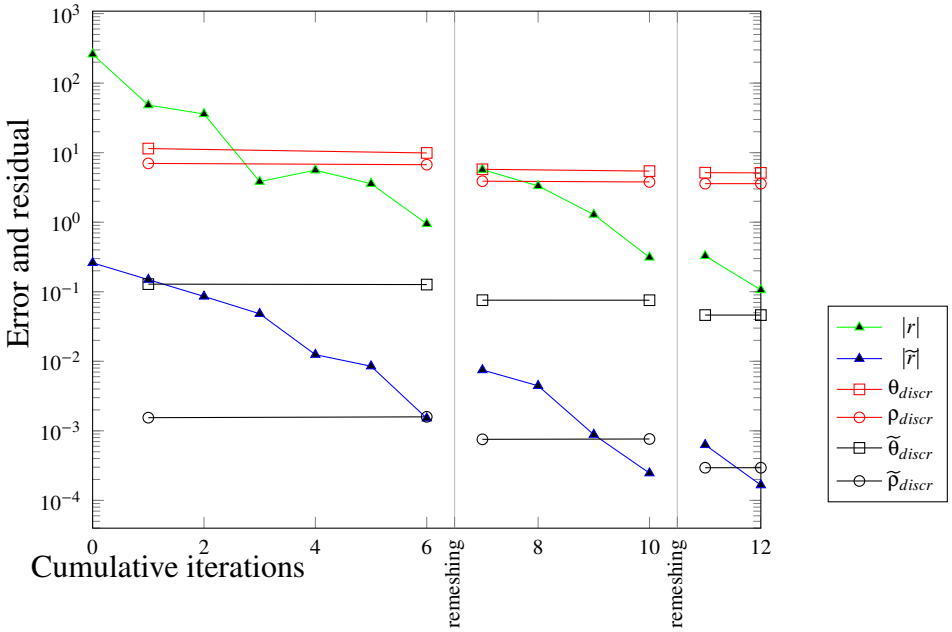


FIGURE 4.7 – Erreurs de discrétisation et résidus au cours des itérations

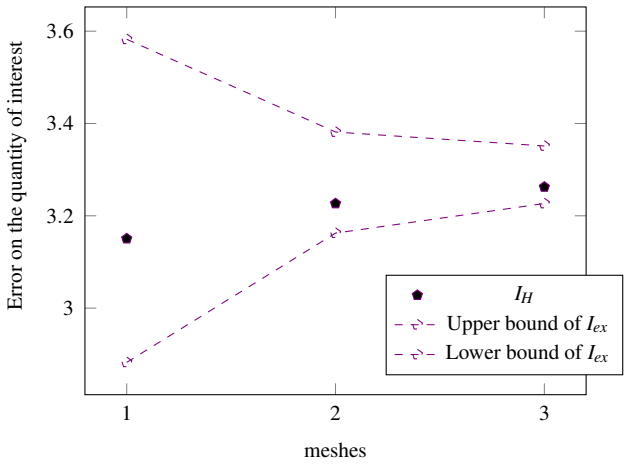


FIGURE 4.8 – Évolution de l'estimation de la quantité d'intérêt et des bornes associées

5 Approche non-intrusive

Les approches non-intrusives s'appuient sur la modélisation par patchs décrite à la section 1.2 et illustrée par la figure 5.1 : la représentation de référence d'un problème est constituée d'une zone d'intérêt et d'une zone complémentaire, en pratique on dispose séparément d'un modèle fin de la zone d'intérêt et d'un modèle global où la zone d'intérêt est représentée par un modèle grossier (dit auxiliaire).

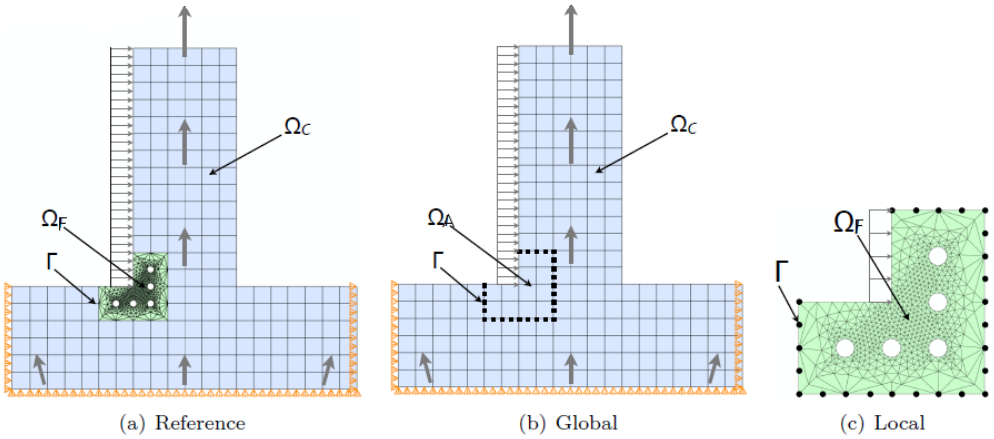


FIGURE 5.1 – Modèles de référence (complémentaire + fin), modèle global (complémentaire + auxiliaire) et modèle local (fin)

L'objectif est de converger vers la solution de référence à partir de calculs globaux grossiers et locaux fins. On s'impose de ne pas intervenir de manière intrusive sur le calcul grossier, autrement dit, on s'autorise uniquement à imposer de nouveaux efforts sur le modèle global.

Il est difficile d'attribuer une paternité à ces méthodes que l'on rencontre dans de nombreux domaines sous des vocables différents, entre autres méthode globale/locale [Mao et Sun, 1991], méthode Chimère [Brezzi *et al.*, 2001], méthode de patch [Jara-Almonte et Knight, 1988; Whitcomb, 1991], méthode de zoom [Hirai *et al.*, 1984; Hirai, 1985; Hecht *et al.*, 2009], méthode semi-Schwarz Lagrange [Chevreuil *et al.*, 2013], *submodeling* [Cormier *et al.*, 1999], méthode de raffinement hiérarchique [Rank et Krause, 1997], et qui peuvent être obtenues comme des cas particuliers de grandes classes de méthodes comme les multigrilles localisées [Passieux *et al.*, 2013] ou les techniques d'*operator-splitting* [Glowinski, 2015].

Ces méthodes connaissent actuellement une grande popularité, en France le projet ANR Icare rassemble de grands groupes industriels comme Airbus et EDF, une PME (Distene) et des grands laboratoires universitaires (ICA Toulouse, GeM Nantes et LMT-Cachan) pour étendre leur cadre d'application (fissuration, stochastique, modèle global simplifié) et leurs performance, voir par exemple[Chevreuil *et al.*, 2013; Duval *et al.*, 2014; Bouclier *et al.*, 2015].

Mes travaux sur ce sujet ont été conduit en collaboration étroite avec Olivier Allix qui a dirigé les thèses de Lionel Gendre, Guillaume Guguin et Maxime Blanchard.

Nous avons montré dans [Gendre *et al.*, 2009] comment appliquer la méthode comme correction du *submodeling* dans Abaqus. La méthode peut s'interpréter comme un point fixe dont la convergence est assurée sous réserve que le modèle auxiliaire est plus rigide que le modèle fin, sinon une relaxation est requise. Dans cette première étude le modèle global était linéaire et la méthode pouvait s'interpréter comme un solveur de Newton modifié, d'où la possibilité d'utiliser des méthodes de quasi-Newton (de type SR1 [Khalfan *et al.*, 1993]) pour accélérer la résolution. Nous avons également essayé d'améliorer la pertinence du modèle fin en enrichissant ses conditions aux limites par une condition mixte construite à l'aide d'une approximation à deux échelles du complément de Schur du modèle complémentaire [Gendre *et al.*, 2011] : les effets lointains sont estimés par le calcul de quelques problèmes dont le chargement excite les torseurs d'interface, les effets proches sont approximés par le complément de Schur de quelques bandes d'éléments (de 0 à 4).

On présente sur la figure 5.2 la convergence des itérations globales/locales sur le cas 3D de la figure 1.8. On voit que la méthode est capable de converger très rapidement pour un tel problème de complexité industrielle : 2 itérations pour l'approche mixte accélérée et 4 itérations pour l'approche primale accélérée suffisent pour diviser l'erreur par plus de 100.

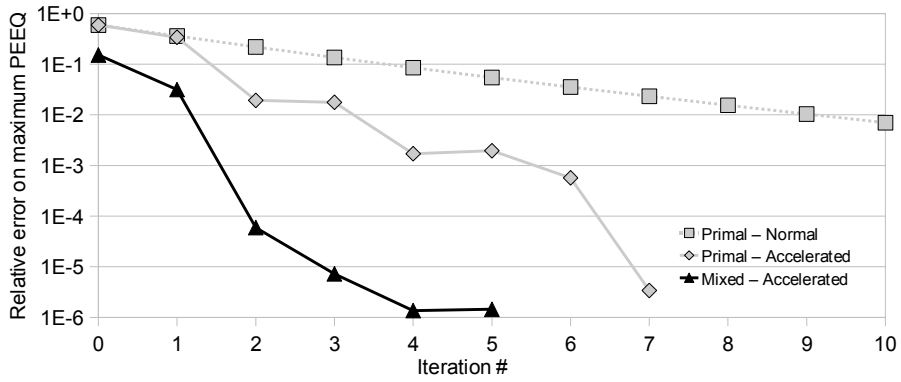


FIGURE 5.2 – Convergence des itérations non-intrusives pour le modèle d'aube banalisé de la figure 1.8

La méthode a été ensuite appliquée à l'inclusion de modèles 3D (avec contact frottant) dans des modèles plaques [Guguin *et al.*, 2014, 2016]. Nous avons ici utilisé code_aster pour manier le modèle global, ce code a l'avantage de permettre la conservation de l'opérateur factorisé en mémoire au cours des itérations. Pour traiter les problèmes fins nous avons utilisé le logiciel Cofast qui est une spécialisation de la LaTIn sur des problèmes de frottement et s'est révélé extrêmement efficace [Champaney *et al.*, 1999].

D'un point de vue méthodologique, une accélération Krylov a été mise en œuvre dans le cas de problèmes entièrement linéaires, pour obtenir une convergence inconditionnelle

et superlinéaire. De plus il a été proposé d'utiliser un recouvrement entre les modèles : on distingue alors une interface de descente (là où les déplacements globaux sont relevés et appliqués au bord du modèle fin) et une interface de remontée (incluse dans le modèle fin) où l'on vient converger vers l'équilibre des efforts. Contrairement à la méthode Arlequin [Ben Dhia, 1998], le raccord se fait sur le bord de la zone de collage et non en volume. Cette double interface trouve un intérêt mécanique dans notre raccord plaque/3D pour éviter certains effets de bord [Guguin *et al.*, 2014] et permet également de traiter les maillages complètement incompatibles sous réserve de pouvoir réaliser quelques intégrations un peu complexes, qui ont été mises en œuvre dans code_aster [Guguin, 2015].

On illustre ces développements par la figure 5.3 où l'on effectue un couplage entre une plaque et la description 3D d'un trou. Le modèle auxiliaire (modèle plaque recouvert par le domaine 3D) est un paramètre de la méthode, il est possible de représenter le trou ou pas (ce dernier choix est bien sûr beaucoup plus simple pour l'ingénieur). La figure 5.4 présente la convergence de la méthode de point fixe et celle du solveur de Krylov en fonction du modèle auxiliaire. On voit l'intérêt d'un modèle auxiliaire le plus fidèle possible au modèle fin, et l'on observe également la convergence rapide du solveur de Krylov.

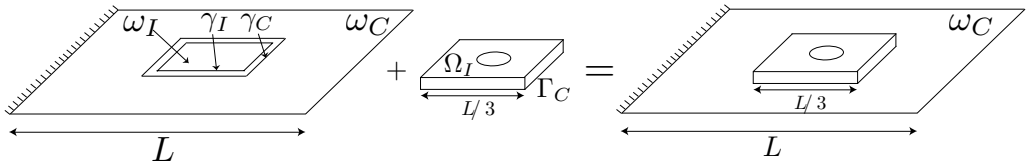


FIGURE 5.3 – Le couplage des modèles de gauche définit le modèle hybride plaque/3D de droite

.

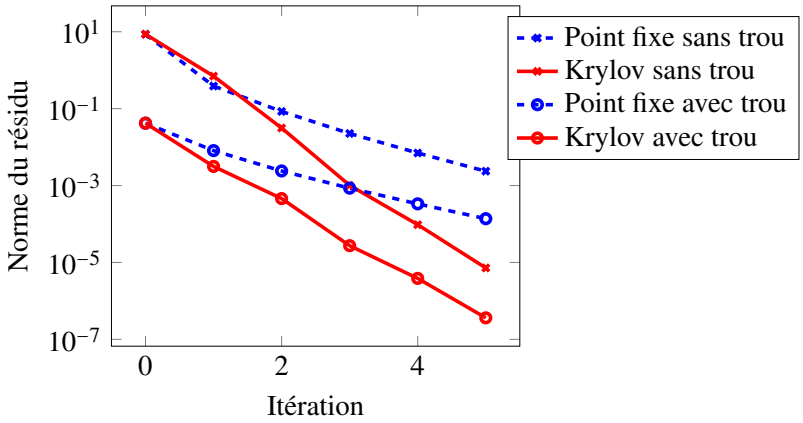


FIGURE 5.4 – Évolution du résidu au cours des itérations en fonction de la finesse du modèle plaque

On trouvera en annexe H le brouillon d'un article en cours de rédaction où je développe la méthode et ses principales propriétés. Il me semble que le cadre le plus propice

pour l'analyse de la méthode (dans le cas d'un seul niveau de patch) est celui de la méthode de Schwarz alternée optimisée, appliquée au modèle fin et au modèle complémentaire : pour le modèle fin on applique une impédance infinie (condition de Dirichlet), pour le modèle complémentaire, on approche l'impédance optimale (complément de Schur du modèle fin) par le complément de Schur du modèle auxiliaire ; de plus les chargements sur le modèle auxiliaire permettent d'initialiser efficacement la méthode.

Tous les développements récents de la méthode peuvent s'inscrire comme des résultats classiques dans ce cadre :

— Résultats sur la convergence (utilisation d'une itération de Richardson modifiée) en linéaire et en non-linéaire (en particulier dans le cas d'opérateurs maximaux monotones),
— Possibilité d'appliquer une accélération de Krylov dans le cas d'un problème tout linéaire, et plus précisément d'un gradient conjugué préconditionné à droite (testé dans [Guguin *et al.*, 2014]),
— Possibilité d'utiliser un recouvrement, en particulier pour la gestion des maillages incompatibles [Guguin *et al.*, 2014],
— Possibilité d'améliorer l'impédance de la zone d'intérêt, avec par exemple une approximation à deux échelles du complément de Schur [Gendre *et al.*, 2011].

Nous travaillons actuellement à l'utilisation de la méthode pour des modèles globaux non-linéaires (élastoviscoplastiques). Le grand challenge est de s'assurer que la méthode converge dans des temps compatibles avec l'ingénierie ; pour cela nous travaillons sur des techniques d'accélération non-intrusives de type Aitken (appliquées dans le cadre non-intrusif avec problème global linéaire dans [Liu *et al.*, 2014]) ou quasi-Newton et sur des méthodes de parallélisme en temps [Gander et Neumüller, 2016] et de découplage d'intégrateur par sous-domaines [Gravouil et Combescure, 2003; Mahjoubi *et al.*, 2009].

L'utilisation d'une condition mixte sur le calcul local a montré sa puissance. La version actuellement proposée, basée sur une approximation à deux échelles, est assez peu efficace en terme de temps de calcul ; à partir des idées du chapitre 2.3, nous serons probablement capables de proposer de nouvelles approximations aussi efficaces en terme d'itérations mais bien plus rapides à construire et faciles à utiliser.

Une autre perspective extrêmement intéressante de ces travaux serait d'apporter des techniques de vérification de manière non-intrusive : ce besoin est très important pour les industriels mais les codes commerciaux et les pratiques des ingénieurs tardent à incorporer de tels outils.

Conclusion

J'ai essayé, dans un format concis, de faire un panoramique de mes recherches depuis une dizaine d'années au LMT-Cachan. Les annexes qui suivent rassemblent une sélection d'articles pour détailler une partie du contenu technique de mon travail.

J'espère avoir convaincu le lecteur de l'intérêt de la décomposition de domaine dans la conception en ingénierie afin de modéliser les problèmes, résoudre efficacement les systèmes associés sur des calculateurs parallèles ou sur des logiciels imposés partiellement fermés, et donner rapidement des bornes d'erreur sur les quantités d'intérêt.

J'ai au cours du document donné des perspectives à court terme à chacune de mes contributions. D'une manière plus générale, il me semble que des gains importants sont à espérer de l'intégration au plus tôt de la sous-structuration dans la chaîne de calcul.

Un premier challenge réside dans le découpage même : même si les méthodes sont bien plus robustes maintenant, de nombreuses difficultés peuvent être évitées en utilisant des sous-structurations adaptées au problème qui engendrent des sous-domaines massifs avec des interfaces régulières et bien positionnées par rapport aux hétérogénéités. Pour cela, une piste consiste à travailler sur la CAO et non sur le maillage, éventuellement avec des éléments isogéométriques. De plus, conserver l'information géométrique est utile au remaillage adaptatif qu'il faut envisager pour garantir la qualité des résultats.

Se pose alors la question du remaillage parallèle, et en particulier de l'équilibrage de charge entre les sous-domaines. C'est une question difficile, les stratégies actuelles sont souvent itératives et basées sur des transferts d'éléments entre voisins, ce qui « déforme » les sous-domaines et rend complexe la réutilisation de l'information numérique générée au cours des systèmes précédents dont on a vu le potentiel, typiquement via le recyclage des sous-espaces de Krylov.

Les progrès récents des solveurs, les évolutions des matériels (avec millions de cœurs disponibles), la volonté des industriels de pratiquer le *virtual testing*, tout concourt à viser des calculs plus complexes et plus gros. Clairement, mes efforts doivent se porter sur l'amélioration de la mise en œuvre des méthodes au niveau informatique.

Dans le projet ANR Semafor, en collaboration avec Safran Tech et l'Onera, nous essayons de pousser les méthodes de multipréconditionnement dans leurs retranchements tant du point de vue de la complexité mécanique, en travaillant sur de vrais problèmes industriels, qu'au niveau informatique, en visant de rester efficace sur des clusters de plusieurs milliers de cœurs. Ce faisant, nous espérons délimiter la classe de problèmes pour laquelle nos partenaires pourront avoir confiance en nos solveurs et déterminer les nouveaux axes de recherche pour faire évoluer nos solveurs.

Les méthodes non-intrusives arrivent maintenant à un certain point de maturité, leurs fondations théoriques sont solides et les applications peuvent se multiplier (dynamique, homogénéisation, patchs mobiles, *etc*). Ces méthodes sont généralement mises en œuvre sous la forme d'un script *python* qui pilote des logiciels (commerciaux ou non). Il me semble que nous sommes maintenant en mesure de proposer une interface de programmation permettant d'encapsuler les logiciels et de réaliser une bonne partie de la program-

mation de manière abstraite (sans référence aux logiciels employés). Cela permettrait de diffuser encore plus facilement nos développements, y compris les solveurs spécialisés codés dans les laboratoires.

Enfin, au delà de la modélisation et du calcul, d'autres problématiques de l'ingénieur peuvent bénéficier des méthodes de décomposition de domaine. Je pense en particulier à l'identification et la validation des modèles. La littérature est déjà importante dans ce domaine, que ce soit d'un point de vue algorithmique où des méthodes de complétion de données de bord s'apparentent fortement aux approches de Schwarz ou de Schur, ou d'un point de vue modélisation des incertitudes où il est fréquent de globaliser les méconnaissances par sous-domaines. Il sera intéressant de voir dans quelle mesure les développements présentés dans ce mémoire peuvent se porter à ces nouvelles applications.

Pour dire « décomposer », le latin utilise *solvo*, dont découle bien-sûr « résoudre ». Ramener un problème à des parties simples, dont on possède une appréhension directe, et analyser leurs interactions, voilà une des bases de l'apprentissage et de la compréhension. Les méthodes de décomposition de domaine ne sont donc que l'application d'un mécanisme fondamental à l'étude des équations aux dérivées partielles. Leur transfert vers l'enseignement est d'ailleurs fulgurant : comprendre une EDP, c'est comprendre comment elle se décompose et, comme le faisait remarquer le professeur Farhat, les sujets d'articles de recherche des années 90 sont devenus en 20 ans des sujets d'examens de master 1. Nul doute que les travaux présentés ici, ou au moins le sous-ensemble (que j'espère non-vide !) de ceux de valeur, connaîtront le même sort.

Ce mémoire se clôt comme il a commencé, par un détournement de Michel Houellebecq :

> Ce n'est pas aussi [insoluble] qu'on le raconte, les relations [entre sous-domaines] : c'est souvent [compliqué], mais c'est rarement [insoluble]. [1]

1. Citation originale : « Ce n'est pas aussi compliqué qu'on le raconte, les relations humaines : c'est souvent insoluble, mais c'est rarement compliqué », *Plateforme* (2001), Michel Houellebecq.

Références

ALLIX, O., GOSSELET, P. et KERFRIDEN, P. (2010a). *Improved multiscale computational strategies for delamination*, volume 55 de *Lecture Notes in Applied and Computational Mechanics*, pages 261–279. Springer.

ALLIX, O., GOSSELET, P., KERFRIDEN, P. et SAAVEDRA, K. (2012). Virtual delamination testing through non-linear multi-scale computational methods : Some recent progress. *CMC : Computers, Materials, & Continua*, 32(2):107–132.

ALLIX, O., KERFRIDEN, P. et GOSSELET, P. (2010b). On the control of the load increments for a proper description of multiple delamination in a domain decomposition framework. *International Journal for Numerical Methods in Engineering*, 83(11):1518 – 1540.

ALLIX, O., KERFRIDEN, P. et GOSSELET, P. (2010c). A relocalization technique for the multiscale computation of delamination in composite structures. *Computer Modeling in Engineering and Sciences*, 55(3):271–292.

BECKER, R. et RANNACHER, R. (1996). A feed-back approach to error control in finite element methods : Basic analysis and examples. *Journal of Numerical Mathematics*, 4:237–264.

Beirão da VEIGA, L., PAVARINO, L. F., SCACCHI, S., WIDLUND, O. B. et ZAMPINI, S. (2014). Isogeometric BDDC preconditioners with deluxe scaling. *SIAM J. Sci. Comput.*, 36(3):A1118–A1139.

BELLENGER, E. et COOREVITS, P. (2005). Adaptive mesh refinement for the control of cost and quality in finite element analysis. *Finite Elements in Analysis and Design*, 41(15):1413 – 1440.

BEN DHIA, H. (1998). Multiscale mechanical problems : the Arlequin method. *Comptes Rendus de l'Academie des Sciences Series IIB Mechanics Physics Astronomy*, 326(12): 899–904.

BERNARDI, C., CHACON REBOLLO, T. et CHACON VERA, E. (2008). A feti method with a mesh independent condition number for the iteration matrix. *Computer Methods in Applied Mechanics and Engineering*, 197(13-16):1410–1429.

BERNARDI, C., MADAY, Y. et PATERA, A. T. (1994). A new nonconforming approach to domain decomposition : the mortar element method. *In* LIONS, H. B. . J.-L., éditeur : *Collège de France Seminar XI*, volume XI, page 13. Pitman.

BOUCARD, P., LADEVÈZE, P., POSS, M. et ROUGÉE, P. (1997). A non-incremental approach for large displacement problems. *Computers & Structures*, 64:499–508.

BOUCLIER, R., PASSIEUX, J.-C., SALAÜN, M. et LABORDE, P. (2015). Enrichissement local de modèles NURBS par couplage non intrusif avec des éléments finis classiques. *In CSMA 2015 - 12ème Colloque National en Calcul des Structures*, Giens (France).

BOVET, C., GOSSELET, P. et SPILLANE, N. (2016). Multipreconditioning for nonsymmetric problems : the case of orthomin and biCG. *submitted to Comptes rendus de l'académie des sciences (mathématiques)*.

BREZZI, F., LIONS, J.-L. et PIRONNEAU, O. (2001). Analysis of a chimera method. *Comptes Rendus de l'Académie des Sciences - Series I - Mathematics*, 332(7):655–660.

BRIDSON, R. et GREIF, C. (2006). A multipreconditioned conjugate gradient algorithm. *SIAM J. Matrix Anal. Appl.*, 27(4):1056–1068 (electronic).

BRUNE, P. R., KNEPLEY, M. G., SMITH, B. F. et TU, X. (2015). Composing scalable nonlinear algebraic solvers. *SIAM Review*, 57(4):535–565.

CAI, X.-C. et SARKIS, M. (1999). A restricted additive schwarz preconditioner for general sparse linear systems. *Siam journal on scientific computing*, 21(2):792–797.

CHAMOIN, L. et LADEVÈZE, P. (2007). Bounds on history-dependent or independent local quantities in viscoelasticity problems solved by approximate methods. *International Journal for Numerical Methods in Engineering*, 71(12):1387–1411.

CHAMPANEY, L., COGNARD, J. et LADÈVEZE, P. (1999). Modular analysis of assemblages of three-dimensional structures with unilateral contact conditions. *Computers and Structures*, 73(1–5):249 – 266.

CHEVREUIL, M., NOUY, A. et SAFATLY, E. (2013). A multiscale method with patch for the solution of stochastic partial differential equations with localized uncertainties. *Computer Methods in Applied Mechanics and Engineering*, 255(0):255 – 274.

CHINESTA, F., AMMAR, A. et CUETO, E. (2009). Recent advances and new challenges in the use of the proper generalized decomposition for solving multidimensional models. *Archives of Computational Methods in Engineering*, 17(4):327–350.

CHINESTA, F., LADEVÈZE, P. et CUETO, E. (2011). A short review on model order reduction based on proper generalized decomposition. *Archives of Computational Methods in Engineering*, 18:395–404. 10.1007/s11831-011-9064-7.

COFFIGNAL, G. et LADEVÈZE, P. (1983). Error computation and optimal mesh in elasticity and elastoplasticity. *In Proc. SMIRT*, 7(L):177–182.

COLTON, D. et KRESS, R. (2001). On the denseness of Herglotz wave functions and electromagnetic Herglotz pairs in Sobolev spaces. *Mathematical Methods in the Applied Sciences*, 24:1289–1303.

CORMIER, N. G., SMALLWOOD, B. S., SINCLAIR, G. B. et MEDA, G. (1999). Aggressive submodelling of stress concentrations. *International Journal for Numerical Methods in Engineering*, 46(6):889–909.

COTTEREAU, R., DÍEZ, P. et HUERTA, A. (2009). Strict error bounds for linear solid mechanics problems using a subdomain-based flux-free method. *Computational Mechanics*, 44(4):533–547.

COTTRELL, J. A., HUGHES, T. J. R. et BAZILEVS, Y. (2009). *Isogeometric Analysis : Toward Integration of CAD and FEA*. Wiley.

CRESTA, P., ALLIX, O., REY, C. et GUINARD, S. (2007). Nonlinear localization strategies for domain decomposition methods : Application to post-buckling analyses. *Computer Methods in Applied Mechanics and Engineering*, 196(8):1436–1446.

CURISKIS, J. I. et VALLIAPPAN, S. (1978). A solution algorithm for linear constraint equations in finite element analysis. *Computers and Structures*, 8:117–124.

DE BORST, R. (2003). Numerical aspects of cohesive-zone models. *Engineering Fracture Mechanics*, 70(14):1743–1757.

DEMBO, R. S., EISENSTAT, S. C. et STEIHAUG, T. (1982). Inexact Newton methods. *SIAM Journal on Numerical analysis*, 19(2):400–408.

DESMEURE, G. (2015). *Une stratégie de décomposition de domaine mixte et multiéchelle pour le calcul des assemblages*. Thèse de doctorat, École normale supérieure de Cachan.

DESMEURE, G., GOSSELET, P., REY, C. et CRESTA, P. (2011a). Etude de différentes représentations des interefforts dans une stratégie de décomposition de domaines mixte. *In Actes du* 10^{eme} *colloque national en calcul des structures*, Giens (Var).

DESMEURE, G., GOSSELET, P., REY, C. et CRESTA, P. (2011b). Traitement des quantités duales d'interface par une représentation dans $H^1/2$ pour une méthode de décomposition de domaine mixte. *In Actes du* 20^{eme} *congrès français de mécanique*, Besançon.

DI PIETRO, D. A. et ERN, A. (2012). *Mathematical Aspects of Discontinuous Galerkin Methods*, volume 69 de *Mathématiques et Applications*. Springer.

DÍEZ, P. et HUERTA, A. (1999). A unified approach to remeshing strategies for finite element h-adaptivity. *Computer Methods in Applied Mechanics and Engineering*, 176(1-4):215–229.

DÍEZ, P., JOSÉ RÓDENAS, J. et ZIENKIEWICZ, O. C. (2007). Equilibrated patch recovery error estimates : simple and accurate upper bounds of the error. *International Journal for Numerical Methods in Engineering*, 69(10):2075–2098.

DÍEZ, P., PARÉS, N. et HUERTA, A. (2003). Recovering lower bounds of the error by postprocessing implicit residual a posteriori error estimates. *International Journal for Numerical Methods in Engineering*, 56(10):1465–1488.

DOHRMANN, C. R. (2003). A preconditionner for substructuring based on constrained energy minimization. *SIAM Journal for Scientific Computing*, 25:246.

DOHRMANN, C. R. et WIDLUND, O. B. (2009). An overlapping Schwarz algorithm for almost incompressible elasticity. *SIAM J. Numer. Anal.*, 47(4):2897–2923.

DOHRMANN, C. R. et WIDLUND, O. B. (2010). Hybrid domain decomposition algorithms for compressible and almost incompressible elasticity. *Internat. J. Numer. Methods Engrg.*, 82(2):157–183.

DOLEAN, V., GANDER, M. J., KHERIJI, W., KWOK, F. et MASSON, R. (2015a). Nonlinear preconditioning : how to use a nonlinear schwarz method to precondition newton's method. working paper or preprint.

DOLEAN, V., JOLIVET, P. et NATAF, F. (2015b). *An introduction to domain decomposition methods*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA. Algorithms, theory, and parallel implementation.

DRYJA, M., SARKIS, M. V. et WIDLUND, O. B. (1996). Multilevel Schwarz methods for elliptic problems with discontinuous coefficients in three dimensions. *Numer. Math.*, 72(3):313–348.

DUVAL, M., PASSIEUX, J.-C., SALAÜN, M. et GUINARD, S. (2014). Non-intrusive coupling : recent advances and scalable nonlinear domain decomposition. *Archives of Computational Methods in Engineering*, pages 1–22.

EISENSTAT, S. C. et WALKER, H. F. (1996). Choosing the forcing terms in an inexact Newton method. *SIAM Journal on Scientific Computing*, 17(1):16–32.

FARHAT, C., LESOINNE, M., LETALLEC, P., PIERSON, K. et RIXEN, D. (2001). FETI-DP : a dual-primal unified FETI method - part i : a faster alternative to the two-level FETI method. *Int. J. Num. Meth . Eng.*, 50(7):1523–1544.

FARHAT, C. et RIXEN, D. (1995). A new coarsening operator for the optimal preconditioning of the dual and primal domain decomposition methods : application to problems with severe coefficient jumps. *In* N. DUANE MELSON, T. A. Manteuffel, S. F. M. et DOUGLASM, C. C., éditeurs : *Proceedings of the Seventh Copper Mountain Conference on Multigrid Methods*, pages 301–316.

FARHAT, C. et ROUX, F. X. (1994). Implicit parallel processing in structural mechanics. *Computational Mechanics Advances*, 2(1):1–124. North-Holland.

GANDER, M. J. (2006). Optimized schwarz methods. *SIAM Journal on Numerical Analysis*, 44(2):699–731.

GANDER, M. J. et NEUMÜLLER, M. (2016). Analysis of a new space-time parallel multigrid algorithm for parabolic problems. *SIAM Journal on Scientific Computing*, 38(4):A2173–A2208.

GENDRE, L., ALLIX, O. et GOSSELET, P. (2011). A two-scale approximation of the Schur complement and its use for non-intrusive coupling. *International Journal for Numerical Methods in Engineering*, 87(9):889–905.

GENDRE, L., ALLIX, O., GOSSELET, P. et COMTE, F. (2009). Non-intrusive and exact global/local techniques for structural problems with local plasticity. *Computational Mechanics*, 44(2):233–245.

GERMAIN, N., BESSON, J., FEYEL, F. et GOSSELET, P. (2007). High performance parallel simulation of structure degradation using nonlocal damage models. *International Journal for Numerical Methods in Engineering*, 71(3):253–276.

GITTELSON, C. J. et HIPTMAIR, R. (2014). Dispersion analysis of plane wave discontinuous Galerkin methods. *International Journal for Numerical Methods in Engineering*, 98(5):313–323.

GLOWINSKI, R. (2015). *Variational Methods for the Numerical Solution of Nonlinear Elliptic Problems*. Society for Industrial and Applied Mathematics, Philadelphia, PA.

GOSSELET, P. et REY, C. (2002). On a selective reuse of Krylov subspaces in Newton-Krylov approaches for nonlinear elasticity. *In Proc. of DD14 - 14th International Conference on Domain Decomposition Methods*, pages 419–426, Cocoyoc, Mexico.

GOSSELET, P. et REY, C. (2007). Non-overlapping domain decomposition methods in structural mechanics. *Archives of computational methods in engineering*, 13(4):515–572.

GOSSELET, P., REY, C., LÉNÉ, F. et DASSET, P. (2002). A domain decomposition method for quasi-incompressible formulation with discontinuous pressure fields. *Revue Européenne des Eléments Finis*, 11:363–378.

GOSSELET, P., REY, C. et PEBREL, J. (2013). Total and selective reuse of Krylov subspaces for the solution to a sequence of nonlinear structural problems. *International Journal for Numerical Methods in Engineering*, 94(1):60–83.

GOSSELET, P., REY, C. et RIXEN, D. (2003). On the initial estimate of interface forces in FETI methods. *Computer Methods in Applied Mechanics and Engineering*, 192:2749–64.

GOSSELET, P., RIXEN, D. et REY, C. (2009). A domain decomposition strategy to efficiently solve structures containing repeated patterns. *International Journal for Numerical Methods in Engineering*, 78(7):828–842.

GOSSELET, P., RIXEN, D., ROUX, F.-X. et SPILLANE, N. (2015). Simultaneous-FETI and Block-FETI : robust domain decomposition with multiple search directions. *International Journal for Numerical Methods in Engineering*, 104(10):905–927.

GRAVOUIL, A. et COMBESCURE, A. (2003). Multi-time-step and two-scale domain decomposition method for non-linear structural dynamics. *International Journal for Numerical Methods in Engineering*, 58(10):1545–1569.

GUGUIN, G. (2015). *Stratégie non-intrusive de couplage plaque/3D pour l'application aux plaques composites stratifiées*. Thèse de doctorat, École normale supérieure de Cachan.

GUGUIN, G., ALLIX, O., GOSSELET, P. et GUINARD, S. (2014). Non intrusive coupling between 3d and 2d laminated composite models based on finite element 3d recovery. *International Journal for Numerical Methods in Engineering*, 98(5):324–343.

GUGUIN, G., ALLIX, O., GOSSELET, P. et GUINARD, S. (2016). On the computation of plate assemblies using realistic 3d joint model : a non-intrusive approach. *Advanced Modeling and Simulation in Engineering Sciences*, 3(16).

HAFERSSAS, R., JOLIVET, P. et NATAF, F. (2015). A robust coarse space for optimized schwarz methods : Soras-geneo-2. *Comptes Rendus Mathematique*, 353(10):959 – 963.

HECHT, F., LOZINSKI, A. et PIRONNEAU, O. (2009). Numerical zoom and the schwarz algorithm. *In Proceedings of the 18th conference on domain decomposition methods*.

HINOJOSA, J., ALLIX, O., GUIDAULT, P.-A. et CRESTA, P. (2014). Domain decomposition methods with nonlinear localization for the buckling and post-buckling analyses of large structures. *Advances in Engineering Software*, 70:13 – 24.

HIRAI, I. (1985). An exact zooming method for finite element analysis. *Finite element analysis and design*, 1:61–69.

HIRAI, I., WANG, B. P. et PILKEY, W. D. (1984). An efficient zooming method for finite element analysis. *International Journal for Numerical Methods in Engineering*, 20(1671-1683):10.

HOANG, T. P., JAPHET, C., KERN, M. et ROBERTS, J. (2014). Ventcell conditions with mixed formulations for flow in porous media. *In* DICKOPF, T., GANDER, M., HALPERN, L., KRAUSE, R. et PAVARINO, L., éditeurs : *Domain Decomposition Methods in Science and Engineering XXII*, pages 531–540, Lugano (Switzerland).

JARA-ALMONTE, C. C. et KNIGHT, C. E. (1988). The specified boundary stiffness/force SBSF method for finite element subregion analysis. *International Journal for Numerical Methods in Engineering*, 26(7):1567–1578.

JIRÁNEK, P., STRAKOŠ, Z. et VOHRALÍK, M. (2010). A posteriori error estimates including algebraic error and stopping criteria for iterative solvers. *SIAM Journal on Scientific Computing*, 32(3):1567–1590.

KERFRIDEN, P., ALLIX, O. et GOSSELET, P. (2009). A three-scale domain decomposition method for the 3D analysis of debonding in laminates. *Computational Mechanics*, 44(3):343–362.

KERFRIDEN, P., GOSSELET, P., ADHIKARI, S. et BORDAS, S. (2011). Bridging proper orthogonal decomposition methods and augmented Newton-Krylov algorithms : an adaptive model order reduction for highly nonlinear mechanical problems. *Computer Methods in Applied Mechanics and Engineering*, 200(5-8):850–866.

KHALFAN, H. F., BYRD, R. H. et SCHNABEL, R. B. (1993). A theoretical and experimental study of the symmetric rank-one update. *SIAM Journal on Optimization*, 3(1):1–24.

KLAWONN, A., LANSER, M. et RHEINBACH, O. (2014a). Nonlinear FETI-DP and BDDC methods. *SIAM J. Sci. Comput.*, 36(2):A737–A765.

KLAWONN, A., RADTKE, P. et RHEINBACH, O. (2014b). Adaptive coarse spaces for BDDC with a transformation of basis. *In Twenty Second International Conference on Domain Decomposition Methods*.

KLAWONN, A. et RHEINBACH, O. (2007). Robust FETI-DP methods for heterogeneous three dimensional elasticity problems. *Computer Methods in Applied Mechanics and EngineerinComputer Methods in Applied Mechanics and Engineering*, 196(8):1400–1414.

KLAWONN, A., RHEINBACH, O. et WIDLUND, O. B. (2008). An analysis of a FETI–DP algorithm on irregular subdomains in the plane. *SIAM J. Numer. Anal.*, 46(5):2484–2504.

KLAWONN, A. et WIDLUND, O. (2001). Feti and neumann-neumann iterative substructuring methods : Connections and new results. *Communications on Pure and Applied Mathematics*, 54(1):57–90.

KLEISS, S. K., PECHSTEIN, C., JÜTTLER, B. et TOMAR, S. (2012). IETI - isogeometric tearing and interconnecting. *Computer Methods in Applied Mechanics and Engineering*, 247-248:201 – 215.

KOVALEVSKY, L. et GOSSELET, P. (2016). A quasi-optimal coarse problem and an augmented Krylov solver for the variational theory of complex rays. *International Journal for Numerical Methods in Engineering*, online.

LADEVÈZE, P. (1975). *Comparaison de modèles de milieux continus*. Thèse de doctorat, Université P. et M. Curie.

LADEVÈZE, P. (1985). Sur une famille d'algorithmes en mécanique des structures. *Comptes Rendus Académie des Sciences - Mécanique, Paris*, 300(2):41–45.

LADEVÈZE, P. (1989). The large time increments method for analysis of structures with nonlinear behavior described by internal variables (in french). *Comptes Rendus Académie des Sciences - Mécanique, Paris*, 309(2):1095.

LADEVÈZE, P. (1996). A new computational approach for structure vibrations in the medium frequency range,. *Comptes Rendus Académie des Sciences Paris, série II*, pages 849–856.

LADEVÈZE, P. (1999). *Nonlinear Computational Structural Mechanics – New Approaches and Non-Incremental Methods of Calculation*. Springer Verlag.

LADEVÈZE, P. (2006). Upper error bounds on calculated outputs of interest for linear and nonlinear structural problems. *Comptes Rendus Académie des Sciences - Mécanique, Paris*, 334(7):399–407.

LADEVÈZE, P. (2008). Strict upper error bounds on computed outputs of interest in computational structural mechanics. *Computational Mechanics*, 42(2):271–286.

LADEVÈZE, P., BLAYSAT, B. et FLORENTIN, E. (2012a). Strict upper bounds of the error in calculated outputs of interest for plasticity problems. *Computer Methods in Applied Mechanics and Engineering*, 245–246(0):194–205.

LADEVÈZE, P. et LEGUILLON, D. (1983). Error estimate procedure in the finite element method and application. *SIAM Journal of Numerical Analysis*, 20(3):485–509.

LADEVÈZE, P. et PELLE, J. P. (2001). *La maîtrise du calcul en mécanique linéaire et non linéaire*. Hermes.

LADEVÈZE, P., PELLE, J. P. et ROUGEOT, P. (1993). Error estimation and mesh optimization for classical finite elements. *Engineering Computations*, 8(1):69–80.

LADEVÈZE, P., PLED, F. et CHAMOIN, L. (2012b). New bounding techniques for goal-oriented error estimation applied to linear problems. *International Journal for Numerical Methods in Engineering*.

LADEVÈZE, P., NÉRON, D. et GOSSELET, P. (2007). On a mixed and multiscale domain decomposition method. *Computer Methods in Applied Mechanics and Engineering*, 196(8):1526–1540.

LE TALLEC, P. (1994). Domain decomposition methods in computational mechanics. *Comput. Mech. Adv.*, 1(2):121–220.

LIU, Y., SUN, Q. et FAN, X. (2014). A non-intrusive global/local algorithm with non-matching interface : Derivation and numerical validation. *Computer Methods in Applied Mechanics and Engineering*, 277:81–103.

MAGOULÈS, F., ROUX, F. X. et SERIES, L. (2006). Algebraic approximation of dirichlet-to-neumann maps for the equations of linear elasticity. *Computer Methods in Applied Mechanics and Engineering*, 195(29-32):3742–3759.

MAHJOUBI, N., GRAVOUIL, A. et COMBESCURE, A. (2009). Coupling subdomains with heterogeneous time integrators and incompatible time steps. *Computational mechanics*, 44(6):825–843.

MANDEL, J. (1993). Balancing domain decomposition. *Communications in Numerical Methods in Engineering*, 9(3):233.

MANDEL, J. et BREZINA, M. (1996). Balancing domain decomposition for problems with large jumps in coefficients. *Math. Comp.*, 65(216):1387–1401.

MAO, K. M. et SUN, C. T. (1991). A refined global-local finite element analysis method. *International Journal for Numerical Methods in Engineering*, 32(1):29–43.

NEGRELLO, C., GOSSELET, P., REY, C. et PEBREL, J. (2016). Substructured formulations of nonlinear structure problems — influence of the interface condition. *International Journal for Numerical Methods in Engineering*, online.

NÉRON, D. et LADEVÈZE, P. (2010). Proper generalized decomposition for multiscale and multiphysics problems. *Archives of Computational Methods in Engineering*, 17(4): 351–372.

ODEN, J. T. et PRUDHOMME, S. (2001). Goal-oriented error estimation and adaptivity for the finite element method. *Computers & Mathematics with Applications*, 41(5-6):735–756.

OUMAZIZ, P., GOSSELET, P. et BOUCARD, P.-A. (2016). A non-intrusive implementation of a mixed domain decomposition method for frictional contact problems. *submitted to Computational Mechanics*.

PAIGE, C. C. et SAUNDERS, M. A. (1982). LSQR : An algorithm for sparse linear equations and sparse least squares. *ACM Transactions on Mathematical Software*, 8(1):43–71.

PANETIER, J., LADEVÈZE, P. et CHAMOIN, L. (2010). Strict and effective bounds in goal-oriented error estimation applied to fracture mechanics problems solved with xfem. *International Journal for Numerical Methods in Engineering*, 81(6):671–700.

PANETIER, J., LADEVÈZE, P. et LOUF, F. (2009). Strict bounds for computed stress intensity factors. *Computers and Structures*, 87(15-16):1015–1021.

PARÉS, N., DÍEZ, P. et HUERTA, A. (2006). Subdomain-based flux-free a posteriori error estimators. *Computer Methods in Applied Mechanics and Engineering*, 195(4-6):297–323.

PARÉS, N., SANTOS, H. et DÍEZ, P. (2009). Guaranteed energy error bounds for the poisson equation using a flux-free approach : Solving the local problems in subdomains. *International Journal for Numerical Methods in Engineering*, 79(10):1203–1244.

PARRET-FRÉAUD, A., REY, C., GOSSELET, P. et FEYEL, F. (2010). Fast estimation of discretization error for FE problems solved by domain decomposition. *Computer Methods in Applied Mechanics and Engineering*, 199(49-52):3315–3323.

PARRET-FRÉAUD, A., REY, V., GOSSELET, P. et REY, C. (2016). Improved recovery of admissible stress in domain decomposition methods – application to heterogeneous structures and new error bounds for FETI-DP. *accepted in International Journal for Numerical Methods in Engineering*.

PASSIEUX, J.-C., RÉTHORÉ, J., GRAVOUIL, A. et BAIETTO, M.-C. (2013). Local/global non-intrusive crack propagation simulation using a multigrid x-fem solver. *Computational Mechanics*, 52(6):1381–1393.

PEBREL, J., REY, C. et GOSSELET, P. (2008). A nonlinear dual domain decomposition method : application to structural problems with damage. *International Journal for Multiscale Computational Engineering*, 6(3):251–262.

PLED, F., CHAMOIN, L. et LADEVÈZE, P. (2011). On the techniques for constructing admissible stress fields in model verification : Performances on engineering examples. *International Journal for Numerical Methods in Engineering*, 88(5):409–441.

PRUDHOMME, S. et ODEN, J. T. (1999). On goal-oriented error estimation for elliptic problems : application to the control of pointwise errors. *Computer Methods in Applied Mechanics and Engineering*, 176(1-4):313–331.

RANK, E. et KRAUSE, R. (1997). A multiscale finite-element method. *Computers & Structures*, 64(1–4):139–144.

REY, V., GOSSELET, P. et REY, C. (2014a). Study of the strong prolongation equation for the construction of statically admissible stress fields : implementation and optimization. *Computer Methods in Applied Mechanics and Engineering*, 268(1):82–104.

REY, V., GOSSELET, P. et REY, C. (2015). Strict bounding of quantities of interest in computations based on domain decomposition. *Computer Methods in Applied Mechanics and Engineering*, 287(1):212–228.

REY, V., GOSSELET, P. et REY, C. (2016). Strict lower bounds with separation of sources of error in non-overlapping domain decomposition methods. *International Journal for Numerical Methods in Engineering*, online.

REY, V., REY, C. et GOSSELET, P. (2014b). A strict error bound with separated contributions of the discretization and of the iterative solver in non-overlapping domain decomposition methods. *Computer Methods in Applied Mechanics and Engineering*, 270(1):293–303.

RIXEN, D. et FARHAT, C. (1999a). A simple and efficient extension of a class of substructure based preconditioners to heterogeneous structural mechanics problems. *Internat. J. Num. Meth. Engin.*, 44(4):489–516.

RIXEN, D. J. et FARHAT, C. (1999b). A simple and efficient extension of a class of substructure based preconditioners to heterogeneous structural mechanics problems. *International Journal for Numerical Methods in Engineering*, 44(4):489–516.

ROUX, F.-X. (2009). A feti-2lm method for non-matching grids. *In* BERCOVIER, M., GANDER, M. J., KORNHUBER, R. et WIDLUND, O., éditeurs : *Domain Decomposition Methods in Science and Engineering XVIII*, volume 70 de *Lecture Notes in Computational Science and Engineering*, pages 121–128. Springer Berlin Heidelberg.

RYCKELYNCK, D. (2008). Hyper-reduction of mechanical models involving internal variables. *International Journal for Numerical Methods in Engineering*, 77(1):75 – 89.

SAAD, Y. (2000). *Iterative methods for sparse linear systems*. PWS Publishing Company, 3rd edition édition.

SAAVEDRA, K., ALLIX, O. et GOSSELET, P. (2012). On a multiscale strategy and its optimization for the simulation of combined delamination and buckling. *International Journal for Numerical Methods in Engineering*, 91(7):772–798.

SAAVEDRA, K., ALLIX, O., GOSSELET, P., HINOJOSA, J. et VIARD, A. (2016). An enhanced non-linear multi-scale strategy for the simulation of buckling and delamination on 3d composite plates. *submitted to Computer Methods in Applied Mechanics and Engineering*.

SPILLANE, N. (2016). An Adaptive Multipreconditioned Conjugate Gradient Algorithm. *SIAM J. Sci. Comput.*, 38(3):A1896–A1918.

SPILLANE, N., DOLEAN, V., HAURET, P., NATAF, F., PECHSTEIN, C. et SCHEICHL, R. (2014). Abstract robust coarse spaces for systems of PDEs via generalized eigenproblems in the overlaps. *Numer. Math.*, 126(4):741–770.

SPILLANE, N. et RIXEN, D. J. (2013). Automatic spectral coarse spaces for robust FETI and BDD algorithms. *Internat. J. Num. Meth. Engin.*, 95(11):953–990.

V. DOLEAN, G. Rapin, F. N. (2009). Deriving a new domain decomposition method for the stokes equations using smith factorization. *Mathematics of Computation*, 78:789–814.

VERFÜRTH, R. (1996). *A Review of A Posteriori Error Estimation and Adaptive Mesh-refinement Techniques*. Wiley-Teubner, Stuttgart.

VOHRALÍK, M. (2007). A posteriori error estimates for lowest-order mixed finite element discretizations of convection-diffusion-reaction equations. *SIAM Journal on Numerical Analysis*, 45(4):1570–1599.

WECK, N. (2004). Approximation by Herglotz wave functions. *Mathematical Methods in the Applied Sciences*, 27(2):155–162.

WHITCOMB, J. D. (1991). Iterative global/local finite element analysis. *Computers and structures*, 40(4):1027–1031.

WRIGGERS, P. (2006). *Computational Contact Mechanics*. Springer, Berlin, 2 édition.

ZHU, J. Z. et ZIENKIEWICZ, O. C. (1988). Adaptive techniques in the finite element method. *Communications in Applied Numerical Methods*, 4(2):197–204.

ZIENKIEWICZ, O. C. et ZHU, J. Z. (1987). A simple error estimator and adaptive procedure for practical engineerng analysis. *International Journal for Numerical Methods in Engineering*, 24(2):337–357.

A Article [Allix *et al.*, 2012], méthode Latin pour le calcul du délaminage

Cette article présente un rapide bilan sur les développements réalisés autour de la méthode LaTIn pour la simulation du flambage et du délaminage des composites lamifiés. L'idée fondamentale est d'utiliser les interfaces de la décomposition de domaine pour supporter les comportements mécaniques surfaciques tels que le délaminage et le contact. Ces problèmes très difficiles ont nécessité de nombreuses adaptations quant-au solveur non-linéaire, au choix des directions de recherche, et au problème grossier. La bibliographie de l'article renvoie vers des articles détaillant ces contributions.

 CMC, vol.32, no.2, pp.107-132, 2012

Virtual Delamination Testing through Non-Linear Multi-Scale Computational Methods: Some Recent Progress

O. Allix[1], P. Gosselet[1], P. Kerfriden[2] and K. Saavedra[3]

Abstract: This paper deals with the parallel simulation of delamination problems at the meso-scale by means of multi-scale methods, the aim being the Virtual Delamination Testing of Composite parts. In the non-linear context, Domain Decomposition Methods are mainly used as a solver for the tangent problem to be solved at each iteration of a Newton-Raphson algorithm. In case of strongly non linear and heterogeneous problems, this procedure may lead to severe difficulties. The paper focuses on methods to circumvent these problems, which can now be expressed using a relatively general framework, even though the different ingredients of the strategy have emerged separately. We rely here on the micro-macro framework proposed in (Ladevèze, Loiseau, and Dureisseix, 2001). The method proposed in this paper introduces three additional features: (i) the adaptation of the macro-basis to situations where classical homogenization does not provide a good preconditioner, (ii) the use of non-linear relocalization to decrease the number of global problems to be solved in the case of unevenly distributed non-linearities, (iii) the adaptation of the approximation of the local Schur complement which governs the convergence of the proposed iterative technique. Computations of delamination and delamination-buckling interaction with contact on potentially large delaminated areas are used to illustrate those aspects.

1 Introduction

Despite the many studies since the beginning of the eighties, the prediction of delamination remains a challenge, both from a scientific and from an industrial point of view. Many difficulties encountered in this field are inherent to composite mod-

[1] LMT-Cachan, ENS Cachan/CNRS, 61, avenue du président Wilson, F-94230 Cachan, France
[2] now at Cardiff University, School of Engineering, Queen's Buildings, The Parade, Cardiff CF24 3AA, UK
[3] now at Departamento de Tecnologías Industriales, Universidad de Talca, Los Niches km 1, Curicó, Chile

eling. The interested reader will find a survey of the literature devoted to the modeling of composites in (Herakovich, 2012). Simulating delamination requires to tackle specific difficulties (i) the complex state of stress which leads to the initiation and the propagation of delamination (ii) the size and the complexity of the problem to be solved when dealing with composite structures. One of the industrial objectives is to replace some of the numerous physical tests required to assess the damage tolerance of composites by numerical simulations (Allix and Blanchard, 2006), or in other words to perform Virtual Delamination Testing (VDT).

It is today well accepted at an industrial level that a realistic VDT procedure should rely on a meso-scale description of the laminates. Such a fine-scale description requires a Finite Element discretisation whose characteristic size is at most the thickness of individual plies, which is of the order of a tenth of a millimetre. For the treatment of industrial parts, the size of the resulting model is therefore huge, which leads to considering the applicability of parallel computations to those large and complex non-linear cases. Indeed, these problems potentially involve multiple delamination fronts, buckling or contact, and therefore instabilities.

The paper describes some attempts to deal with such problems within a multi-scale parallel framework when delamination is the only damage mode that is taken into account. Delamination is here modeled by means of a cohesive zone model. Several such models have been proposed in the literature, for example in (Allix and Ladevèze, 1992; Xu and Needleman, 1994; Schellekens and de Borst, 1994; Qiu, Crisfield, and Alfano, 2001; Camanho and Dávila, 2002). In this paper we make use of the model proposed in (Allix and Corigliano, 1996) but similar results could be obtained with other models once calibrated. The main features of the model are briefly presented in Section 2.

In the remainder of the paper, the basics of the multi-scale strategy are described and test cases are presented to demonstrate its numerical efficiency. The contribution of the paper is the definition of a framework that encompasses different aspects of the strategy that have emerged separately in (Ladevèze, Néron, and Gosselet, 2007; Guidault, Allix, Champaney, and Navarro, 2007; Cresta, Allix, Rey, and Guinard, 2007; Kerfriden, Allix, and Gosselet, 2009; Allix, Kerfriden, and Gosselet, 2010b; Saavedra, Allix, and Gosselet, 2012). Indeed, our previous publications focussed on particular issues related to the present topic. It now appears that the set of tools that we have developed in the past can help formulate a general non-linear multi-scale approach that is adapted to the treatment of large laminated and damageable structures taking into account possible buckling and contact. Multi-scale refers here to a computational technique which involves different scales and which aims at finding the solution over the whole structure at the smallest scale of interest. In this domain, one of the main issues to be addressed is the transfer of information

from one scale to another.

A large number of multiscale methods rely on classical homogenisation of heterogeneous media, which was initiated by (Hill, 1963). For linear periodic media the most efficient method was initiated in (Sánchez-Palencia, 1980). Further developments and related computational approaches can be found in (Zohdi, Oden, and Rodin, 1996; Fish, Shek, Pandheeradi, and Shephard, 1997; Pellegrino, Galvanetto, and Schrefler, 1999; Kouznetsova, Geers, and Brekelmans, 2002; Feyel, 2003). An overview of these methods can be found in (Zohdi and Wriggers, 2008). In computational homogenization approaches the resolution of the macro-problem leads to effective values of the unknowns; then, the micro solution is calculated locally based on the macro solution. The fundamental assumption, besides periodicity, is that the ratio of the characteristic length of the small scale to the characteristic length of the large scale must be small. All areas which do not satisfy the hypothesis of scale separation such as boundary zones or vicinity of cracks, where the material cannot be homogenized, require special treatment. Therefore a large amount of recent studies aim at applying enhanced homogenization schemes to localized failure, see for instance (Massart, Peerlings, and Geers, 2007; Belytschko, Loehnert, and Song, 2007; Coenen, Kouznetsova, Bosco, and Geers, 2012). These extension, which are dedicated to overcome some of the difficulties for a given type of problem are not suited yet to deal with edge effects and large cracks and therefore not directly suitable to the treatment of delamination.

The proposed method is based on a Domain Decomposition approach. These methods have been developed initially for large linear problems for which they can now be considered mature. The most well-known methods are FETI (Farhat, Mandel, and Roux, 1994) and BDD (Mandel, 1993) which where designed for perfect interface and extended, in the case of FETI, to contact (with or without friction) (Dostál, Horák, and Vlach, 2007) or prescribed displacement gaps (Kruis and Bittnar, 2007). FETI and BDD rely on a static condensation for each sub-domain. The resulting interface problem, which is large, is solved iteratively using a preconditioner which is associated to the balance of the rigid-body momentum of the sub-domains. In the case of heterogeneous media, the macro-space of rigid body is not necessarily the most appropriate one and a specific deflation can be performed (Efendiev and Galvis, 2011). It is therefore interesting to define the macro-space as the space which is associated with the homogenized response of the subdomains. The use of homogenization in the context of a Domain Decomposition Method has been proposed in the micro-macro approach of (Ladevèze and Dureisseix, 1999).

Homogenization being a natural tool for composites, we base our approach on this micro-macro framework. The decomposition of the structure into subdomains is chosen to be compatible with the mesomodeling of the laminate. The micro-macro

method relies on mixed interface conditions between subdomains, which allows to treat any kind of interfacial constitutive law directly at the interface level (Ladevèze, Néron, and Gosselet, 2007). Three types of interfaces are considered here: damageable interfaces and contact interfaces that are compatible with the meso-modeling, and perfect interfaces within the plies to split the problem into smaller pieces that are suited to parallelism. This leads to a large number of substructures, and therefore to a large macro-problem. A third scale is then introduced to perform a low cost and precise approximation of the homogeneized macro-problem by using a BDD preconditioner (Kerfriden, Allix, and Gosselet, 2009), as described in Section 3. Note that in this paper only unilateral contact is considered but the framework is well suited to friction (Champaney, Boucard, and Guinard, 2008).

In the context of non-linear problems, Domain Decomposition Methods are mainly used as a solver for the tangent problem to be solved at each iteration of a Newton-Raphson based algorithm (Le Tallec, 1994; Germain, Besson, Feyel, and Gosselet, 2007; Klawonn, Rheinbach, and Wohlmuth, 2007), with if required an arc-length control. Several problems can be encountered in this case. One of such problems is that the choice of the macro-space, and the implied homogenized behavior, is in general not adapted to a situation where a crack interacts with a subdomain. In other words, a preconditioner designed for a problem without crack can become ineffective in the presence of a crack. It has been shown in (Guidault, Allix, Champaney, and Navarro, 2007; Guidault, Allix, Champaney, and Cornuault, 2008), that for cracks crossing interfaces of the domain decomposition, a modification of the macro-basis that adds a few degrees of freedom only makes it possible to obtain the same convergence rate for a cracked domain and for an un-cracked one. Unfortunately such an adaptation is not possible in the case of cracks propagating at the interface between subdomains. We have therefore followed another line, the one of non-linear relocalization (Cresta, Allix, Rey, and Guinard, 2007; Pebrel, Rey, and Gosselet, 2008). This approach aims at solving the problem of the strong deterioration of the convergence rate due to localized non-linear effect (Cai and Keyes, 2002). Following this line of thoughts, we have applied a non-linear relocalization approach on the set of subdomains that are connected to the front of delamination, which is described in Section 4 (Allix, Kerfriden, and Gosselet, 2010b). Note that our approach naturally avoids to conduct useless nonlinear computations far from the delamination front which is the goal of the prediction procedure developped in (Lloberas-Valls, Rixen, Simone, and Sluys, 2012) for Newton-Schur strategies.

Having in mind the problem of the tolerance of laminates to compression after impact (de Moura, Goncalves, Marques, and DeCastro, 1997) we have started to work on the question of the interaction between potentially large growing delaminated areas and buckling, taking into account the contact condition (Saavedra, Al-

lix, and Gosselet, 2012). In order to reconnect the micro part of the solution, the iterative technique introduces parameters which are associated to the influence of neighboring subdomains. These so called "search directions" correspond to Robin parameters whose values were optimized for 3D structures. These parameters must be adapted to the aspect ratios of the slender structures by introducing well-chosen anisotropic coefficients. Moreover, in the case of contact between slender structures over large delaminated areas, the search directions should be optimized according to the interface's state (open or closed) in order to obtain a reasonable convergence rate of the iterative solver. These points are discussed in Section 5. However, changing these parameters requires a refactorization of the homogenized operators, which is computationally expensive. A compromise between convergence speed and unit cost of iterations must be found so that the total CPU time is minimised.

2 The interface model

The interface model relates the jump of displacements $\underline{[u]} = \underline{u}_{p'} - \underline{u}_p$ to the normal Cauchy stress $\underline{t} = \underline{\underline{\sigma}}_p.\underline{n} = \underline{\underline{\sigma}}_{p'}.\underline{n}$ on the interface Γ between two plies p and p'. Here, $\underline{n}$ denotes the outer normal to ply p on $\Gamma_{pp'}$. In this problem, large displacements are considered but the jump of displacement is assumed to be small prior to full delamination. The tractions $\underline{t}$ are uniquely defined and related to the displacement discontinuities by means of the following damageable constitutive relation:

$$\underline{t} = \underline{\underline{K}}.\underline{[u]} \tag{1}$$

The expression of the stiffness operator $\underline{\underline{K}}$ can be given in the reference frame $(\underline{N}_1, \underline{N}_2, \underline{n})$ (as depicted in Figure (1) with $\underline{n} = \underline{N}_3$, $p \equiv$ "Ply 1" and $p' \equiv$ "Ply 2"):

$$\begin{pmatrix} (1-d_1)k_1 & 0 & 0 \\ 0 & (1-d_2)k_2 & 0 \\ 0 & 0 & \left(1 - \left\langle \underline{[u]}.\underline{n} \right\rangle_+ d_3\right) k_3 \end{pmatrix} \tag{2}$$

$< >_+$ is here the positive indicator function.

The local damage variables d_i are introduced into the interface model in order to simulate its softening behavior when the structure is loaded. Their values range from 0 (healthy interface point) to 1 (completely damaged interface point). The parameters d_i are related to the local energy release rates Y_i of the interface's degradation modes as follows:

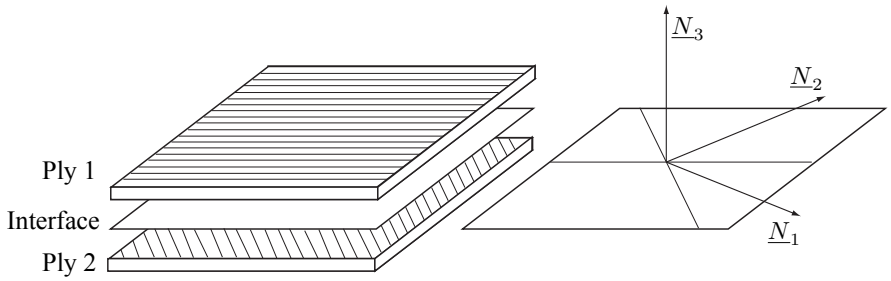


Figure 1: The mesomodel entities

$$Y_i = -\frac{\partial e_d}{\partial d_i} \qquad \text{where} \quad \begin{cases} Y_1 &= \dfrac{1}{2} k_1 [u]_1^2 \\ Y_2 &= \dfrac{1}{2} k_2 [u]_2^2 \\ Y_3 &= \dfrac{1}{2} k_3 \left\langle [u]_3 \right\rangle_+^2 , \end{cases} \tag{3}$$

where e_d is the strain energy of the interface per unit area and $[u]_i$ is the i^{th} component of field $\underline{[u]}$. We assume that the damage variables are functions of a single quantity: the maximum $Y_{|t}$ over time of a combination of the energy release rates $Y_{i|\tau}$, $\tau \leq t$:

$$Y_{|t} = \sup_{\tau \leq t} \left({Y_3}_{|\tau}^{\alpha} + \gamma_1 {Y_1}_{|\tau}^{\alpha} + \gamma_2 {Y_2}_{|\tau}^{\alpha} \right)^{\frac{1}{\alpha}} \tag{4}$$

Thus, the evolution law is defined by:

$$d_1 = d_2 = d_3 = w(Y) \tag{5}$$

where, in general, $\quad w(Y) = \dfrac{n}{n+1} \left(\dfrac{Y}{Y_c} \right)^n .$

This model has the advantage, using a single damage variable, to be able to recover different critical energy release rates for pure mode loading as well as to fit to classical mixed mode criteria. For example, setting $\alpha = 2$ leads to the classical quadratic energy criteria.

3 Main features of the method

As presented in Figure 2, the structure is split into substructures which are connected by interfaces that have mechanical behaviors. In a second stage, the substructures are gathered into "super-substructures" which are assigned to independent processors. In order to ensure the scalability of the method, the interfaces are

the support of a macro (homogenized) problem, the resolution of which requires the solution of a third scale (supermacro) problem on super-interfaces.

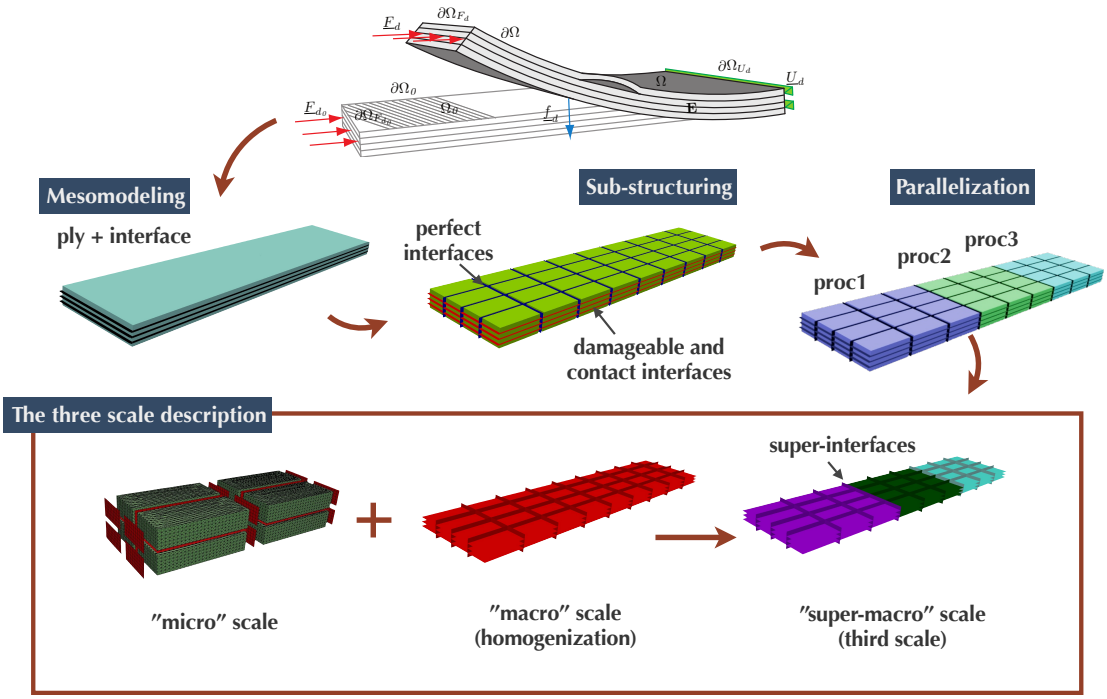


Figure 2: Decomposition of the problem and associated three computational scales

3.1 Micro-macro scale separation

The iterative LaTIn algorithm, which is designed to non-linear problems, is here applied to the solution of the geometrically non-linear substructured reference problem with "material" non-linearities (damage, contact) localized on the interfaces. The finite element method is used to discretize the governing equations. The aim of the method is to find the subdomains fields $\underline{u}_{E_0}$ (displacement), $\underline{\underline{\pi}}_{E_0}$ (Piola-Kirchhoff stress), and the interface fields $\underline{W}_{E_0}$ (displacement), $\underline{F}_{E_0}$ (interforces), where index E ranges over all substructures and index 0 means that the quantity is given in the reference configuration (total Lagrangian approach).

In order to ensure the scalability of the method, a global and linear coarse grid problem is solved. The definition of the macroscopic fields required to construct this problem is done on the interface unknowns only. Whichever the interface behavior, action reaction principle $\underline{F}_{E_0} + \underline{F}_{E'_0} = \underline{0}$ is verified; the principle of the macro-problem is to enforce this equation partially at any time of the iterative solution

process:

$$\forall \underline{W}_{E_0}^{M\star} \in \mathscr{W}_{E_0}^{M}, \quad \int_{\Gamma_{E_0E_0'}} (\underline{F}_{E_0} + \underline{F}_{E_0'}) \cdot \underline{W}_{E_0}^{M\star} d\Gamma_0 = 0, \tag{6}$$

where displacement macrospace $\mathscr{W}_{E_0}^{M}$ is a parameter of the method as well as its dual $\mathscr{F}_{E_0}^{M}$. The definition of the macro-spaces is done through a projector Π, which is chosen L^2-orthogonal (so that the same projector is used by the traction and displacement fields). We then have the following splitting of the interface quantities:

$$\begin{aligned} &\forall (\underline{F}_{E_0}, \underline{W}_{E_0}) \in \mathscr{F}_{E_0} \times \mathscr{W}_{E_0}, \\ &\underline{F}_{E_0} = \underline{F}_{E_0}^{M} + \underline{F}_{E_0}^{m} \text{ with } \underline{F}_{E_0}^{M} = \Pi \underline{F}_{E_0} \text{ and } \underline{W}_{E_0} = \underline{W}_{E_0}^{M} + \underline{W}_{E_0}^{m} \text{ with } \underline{W}_{E_0}^{M} = \Pi \underline{W}_{E_0} \\ &\Rightarrow \int_{\Gamma_{E_0E_0'}} \underline{F}_{E_0} \cdot \underline{W}_{E_0} \, d\Gamma_0 = \int_{\Gamma_{E_0E_0'}} \underline{F}_{E_0}^{M} \cdot \underline{W}_{E_0}^{M} \, d\Gamma_0 + \int_{\Gamma_{E_0E_0'}} \underline{F}_{E_0}^{m} \cdot \underline{W}_{E_0}^{m} \, d\Gamma_0 \end{aligned} \tag{7}$$

Numerical tests showed that in order to ensure the numerical scalability of the method the macro-basis should extract at least the linear part of the interface forces. Indeed, this macro-space contains the part of the interface fields with the highest wavelength. Consequently, according to the Saint-Venant principle, the micro complement only has a local influence.

3.2 Iterative algorithm

It is now possible to split all the equations of the system into two groups:

- non-linear equations in the substructures and macroscopic admissibility of interfaces, whose solutions are elements of the manifold $\mathbf{A_d}$:
  - non-linear kinematic admissibility of the substructures

$$\underline{\underline{E}}_{E_0} = \frac{1}{2}\left(\underline{\underline{\nabla}}_0 \underline{u}_{E_0} + {}^t\underline{\underline{\nabla}}_0 \underline{u}_{E_0} + {}^t\underline{\underline{\nabla}}_0 \underline{u}_{E_0} \, {}^t\underline{\underline{\nabla}}_0 \underline{u}_{E_0}\right), \text{ on } \Omega_{E_0} \tag{8}$$

$$\underline{u}_{E_0|\partial\Omega_{E_0}} = \underline{W}_{E_0|\Gamma_{E_0}}, \text{ on } \Gamma_{E_0E_0'} \tag{9}$$

  - non-linear static admissibility of the substructures

$$\begin{aligned} &\forall (\underline{u}_{E_0}^{\star}, \underline{W}_{E_0}^{\star}) \in \mathscr{U}_{E_0}^{0} \times \mathscr{W}_{E_0}^{0} \quad \textit{such that} \quad \underline{u}_{E_0|\partial\Omega_{E_0}}^{\star} = \underline{W}_{E_0|\Gamma_{E_0}}^{\star}, \\ &\int_{\Omega_{E_0}} \underline{\underline{\pi}}_{E_0} : \underline{\underline{\dot{E}}}(\underline{u}_{E_0}^{\star}) \, d\Omega_0 \\ &\qquad = \int_{\Omega_{E_0}} \rho_{E_0} \, \underline{f}_d \cdot \underline{u}_{E_0}^{\star} \, d\Omega_0 + \int_{\Gamma_{E_0}} \underline{F}_{E_0} \cdot \underline{W}_{E_0}^{\star} \, d\Gamma_0 \end{aligned} \tag{10}$$

- behavior of the substructures

$$\underline{\underline{\pi}}_{E_0} = \frac{\partial \psi}{\partial \underline{\underline{E}}_{E_0}} \text{ , on } \Omega_{E_0} \text{ ,} \tag{11}$$

- macroscopic admissibility of the interfaces (6).

• local (non-linear) equations in the interfaces whose solutions belong to the manifold Γ:

- interface behavior (perfect, contact and cohesive interfaces) and boundary conditions (see Section 2).

The solution $s_{ref} = (\underline{u}_{E_0}, \underline{\underline{\pi}}_{E_0}, \underline{W}_{E_0}, \underline{F}_{E_0})$ is such that:

$$s_{ref} \in \mathbf{A_d} \cap \Gamma \tag{12}$$

Note that in the case of small perturbations, the admissibility equations are linear. In this case, the manifold $\mathbf{A_d}$ is an affine space.

The resolution scheme consists in seeking the solution s_{ref} alternatively in these two manifolds: first, one finds a solution s_n in $\mathbf{A_d}$, then a solution $\widehat{s}_{n+\frac{1}{2}}$ in Γ. In order for the two problems to be well-posed, search directions $\mathbf{E}^+$ and $\mathbf{E}^-$ linking the solutions s and $\widehat{s}$ through the iterative process are introduced (see Figure 3).

• At the global stage, starting from fields $(\underline{\widehat{W}}_{E_0}, \underline{\widehat{F}}_{E_0})$ satisfying the interface equations (manifold Γ), we seek fields $(\underline{W}_{E_0}, \underline{F}_{E_0})$ in $\mathbf{A_d}$ (satisfying the subdomain equations) using the following closing relation:

$$k^-_{E_0}\left(\underline{W}_{E_0} - \underline{\widehat{W}}_{E_0}\right) + \left(\underline{F}_{E_0} - \underline{\widehat{F}}_{E_0}\right) = 0 \tag{13}$$

$k^-_{E_0}$ is the search direction associated to the global stage.

• At the local stage, starting from fields $(\underline{W}_{E_0}, \underline{F}_{E_0})$ satisfying the subdomain equations (belonging to manifold $\mathbf{A_d}$), we seek fields $(\underline{\widehat{W}}_{E_0}, \underline{\widehat{F}}_{E_0})$ satisfying the interface equations (manifold Γ), using the following closing relation:

$$k^+_{E_0}\left(\underline{W}_{E_0} - \underline{\widehat{W}}_{E_0}\right) - \left(\underline{F}_{E_0} - \underline{\widehat{F}}_{E_0}\right) = 0 \tag{14}$$

$k^+_{E_0}$ is the search direction associated to the local stage.

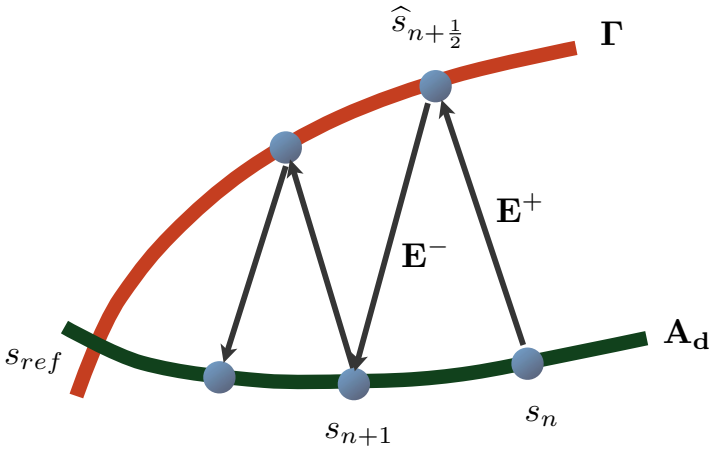


Figure 3: Schematic representation of the iterative LaTIn algorithm

3.3 Third scale

The macro-problem being linear, discrete and sparse, it can be efficiently solved by a balancing domain decomposition algorithm (Mandel, 1993). To do so, substructures are gathered into super-substructures. The macro-problem is condensed at the super-interfaces and this condensed problem is solved by a conjugate gradient algorithm. The balance of super-structures requires the global solution of a third scale problem whose size is the number of rigid body motions of floating super-substructures.

Figure (4) shows the convergence rate of the LaTIn algorithm when the conjugate gradient scheme for the condensed macroproblem is stopped after a fixed number of iterations. The test case is the holed plate under traction loading with the super-substructuring pattern given in Figure (5). This problem is 3.4 10^6 degrees of freedom large, it is distributed in 520 subdomains with $1,350$ interfaces leading to a 12 10^3-unknown macro-problem. 8 super-substructures are employed so that the condensed macro-problem on the super-interfaces has $1,746$ unknowns and the dimension of the third scale (associated to the rigid body motions of the super-substructures) is only 36.

It appears clearly that only very few iterations of the conjugate gradient scheme are required to get the necessary part of the macro-displacement Lagrange multiplier leading to the multiscale convergence rate of the LaTIn algorithm. Typically, the algorithm is stopped when the residual error (normalized by the initial error) falls below 10^{-1}. The admissibility of the macro-forces is thus enforced on a third level, which is sufficient to determine the part of the solution which needs to be transmitted at each iteration of the resolution. In that case the time required to

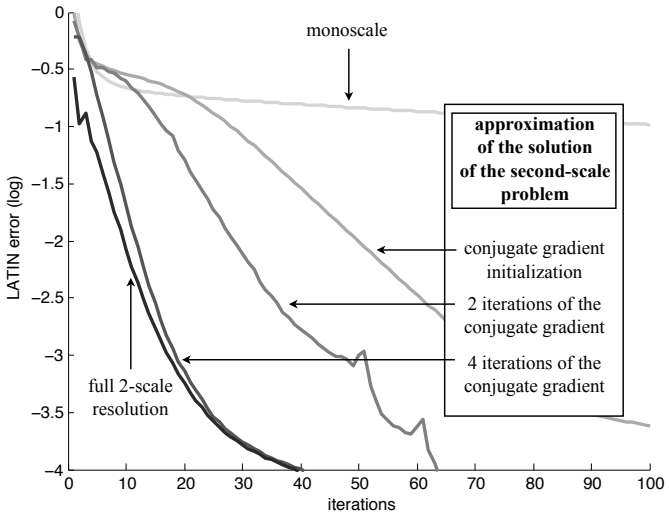


Figure 4: LaTIn convergence curves for different numbers of macro iterations

solve the macro-problem is divided by a factor 100.

Figure 6 presents a larger case where a direct computation of the macroproblem would be too expensive. This 12 10^6 dof problem is distributed on $10,960$ substructures and 32 10^3 interfaces, leading to a 300 10^3 macro d.o.f, the super-substructuring leads to a 40 10^3 condensed dof problems and 150 super-macro d.o.f. Figure 7 shows the extension of delamination in the final converged state.

The previous examples allow us to make an important remark related to our vision of the problem of large-scale simulation of delamination. The reader will notice that we use a homogeneously fine finite element discretization to approximate the displacement field in the structure. Specifically, the mesh is sufficiently fine to handle the propagation of a delamination front at an arbitrary location in the structure, which permits to handle multiple and arbitrary large delamination fronts without remeshing. The fact that we are not willing to remesh is a choice driven by the observation that error estimation, mesh adaptivity and field transfer are, for parallel problems and for problems with softening behaviors, a subject of research in themselves and no mature tool has emerged from pioneer investigations which can be found, for instance, in (Ladevèze, Moës, and Douchin, 1999; Huerta, Rodríguez-Ferran, and Diez, 2002; Larsson, Runesson, and Hansbo, 2003; Allix, Kerfriden, and Gosselet, 2010a; Parret-Fréaud, Rey, Gosselet, and Feyel, 2010). Of course, the price that we pay for the expected robustness associated to this choice is a large number of arguably unnecessary degrees of freedom.

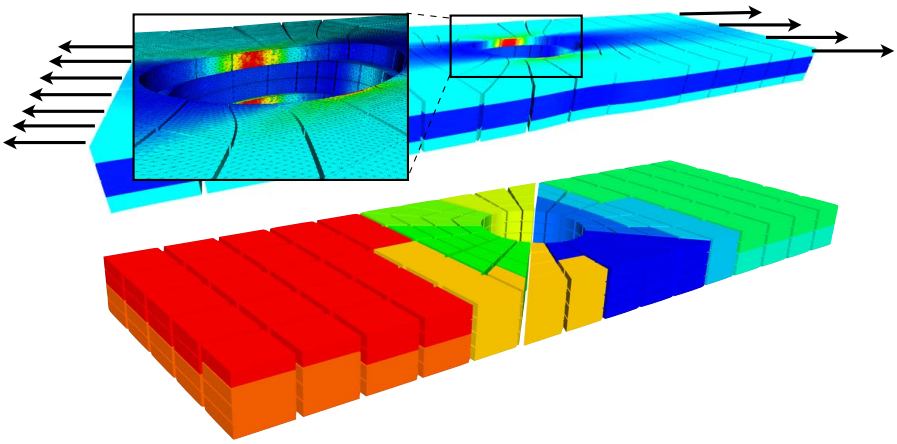


Figure 5: Different levels of substructuring

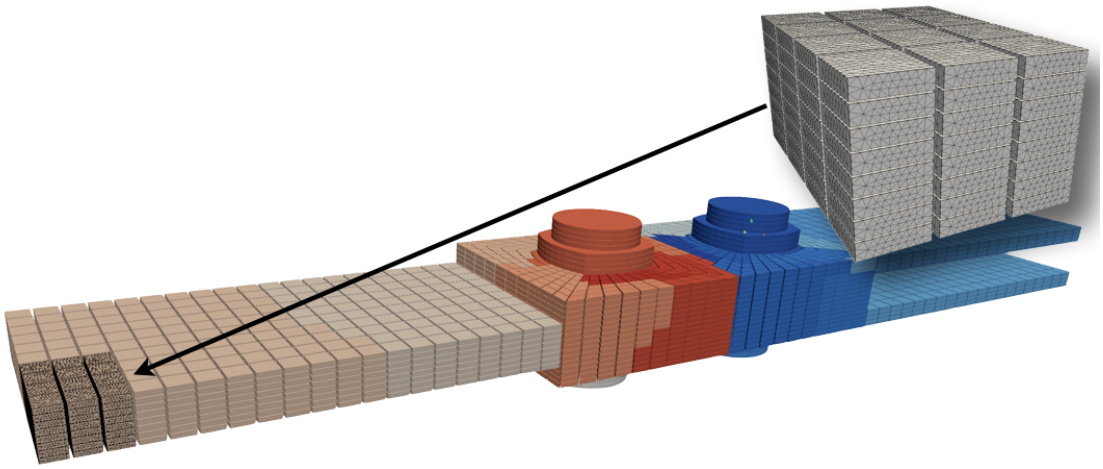


Figure 6: Bolted joint, substructuring and super-substructuring

4 Relocalization for the propagation of delamination

Let us consider a simple DCB case, as illustrated in Figure 8. Figure 9 shows that the convergence of the classical approach (blue bars) is strongly deteriorated when delamination starts to propagate (time step 3). Indeed, in average 10 times more iterations (and CPU time) are required. This drop in the convergence rate occurring when the cracks propagates can be explained by two driving factors:

- the singularity near the tip of the crack is very poorly represented by linear macro quantities. Therefore, the complementary part is not localized enough and many iterations are required to transmit them through the structure;
- the strong non-linearities require many iterations to converge, in particular

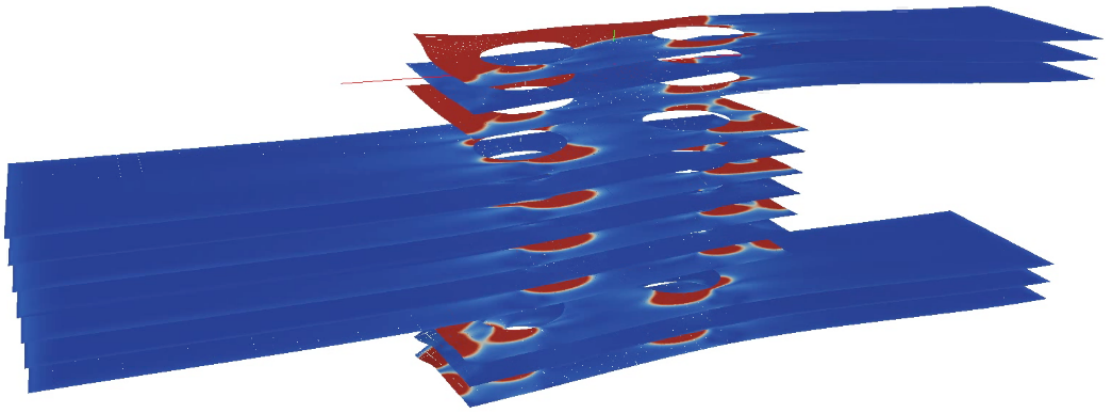


Figure 7: Final delamination state of the bolted joint

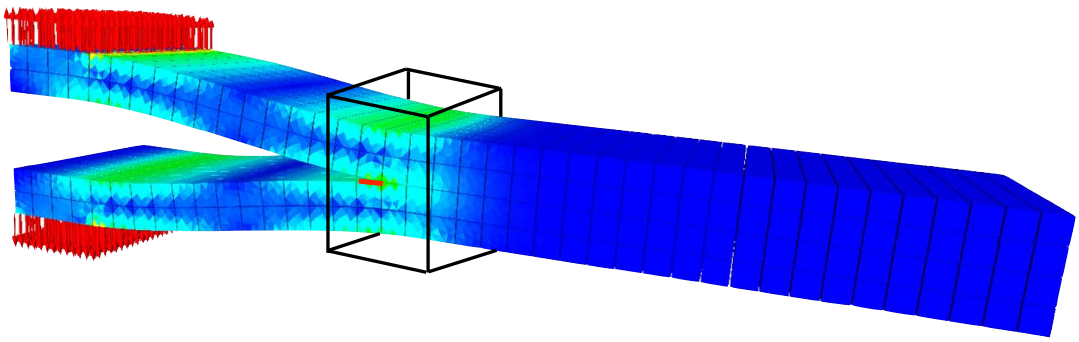


Figure 8: Relocalization box around the crack tip

the prediction of the location of the crack tip at a given time step requires the quasi-convergence of a large number of consecutive equilibrium states.

Using non-linear relocalization in a box containing the crack tip and the process zone is a way to deal with those two problems: after each linear stage, a subproblem is extracted around the crack tip (within the so-called relocalization box whose size is a parameter) as shown in Figure 8. This smaller problem is solved non-linearly with Robin boundary conditions satisfying the macro-equilibrium of the structure.

Being far enough from the singularity, the macro quantities at the boundary of the box represent the global fields correctly and are then relevant for the prediction of the non-linear evolution of the crack. The resulting modifications only have a small-scale effect which can be transmitted by the upcoming local stage.

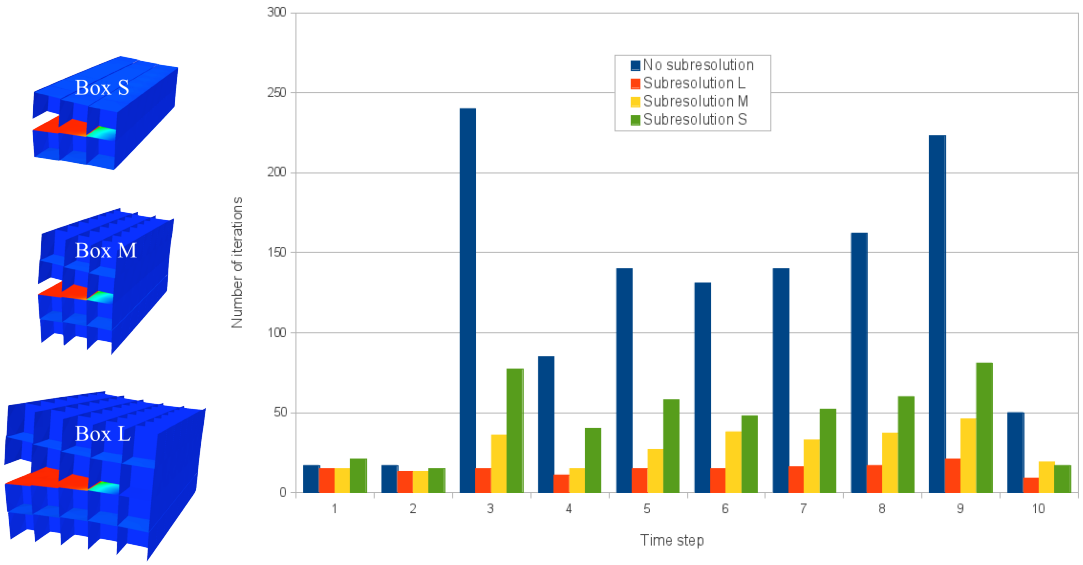


Figure 9: Performance of the relocalization technique w.r.t. the size of the relocalization box

The other bars in Figure 9 show the influence of the size of the relocalization box on the convergence. Of course the larger the box, the best the filtering of the singularities on the boundary of the box, and the fastest the convergence. The counterpart is that the larger the box, the more expensive the non-linear computations associated to the relocalized solutions, which are obtained by a LaTIn method restricted to the box, are.

5 Further improvements for delamination/buckling and contact

5.1 Taking into account the structure's slenderness

Optimal search direction of one subdomain (for the global step) is known to be the Schur complement of the rest of the structure. In general a very coarse approximation, in practice a scalar, is sufficient to ensure good convergence of the LaTIn method. In (Saavedra, Allix, and Gosselet, 2012) it was shown that for plates, the slenderness induces a structural anisotropy that has to be taken into account by the search directions. A dimensional analysis leads to the following relationship between the longitudinal (n) and transverse (t) search directions (L_0 is the characteristic length of the structure, h_0 is its thickness):

$$\frac{\left(k^m_{E_0}\right)_n}{\left(k^m_{E_0}\right)_t} = \left(\frac{L_0}{h_0}\right)^2$$

Here superscript *m* stands for the micro part of the search direction. The macro part of the search direction has to be set to a large value on perfect interfaces so that the macro continuity of displacements is almost satisfied throughout the iterative process.

Figure 10 shows that these adaptations of the search directions (i.e. anisotropic search directions) are necessary to ensure the scalability of the method.

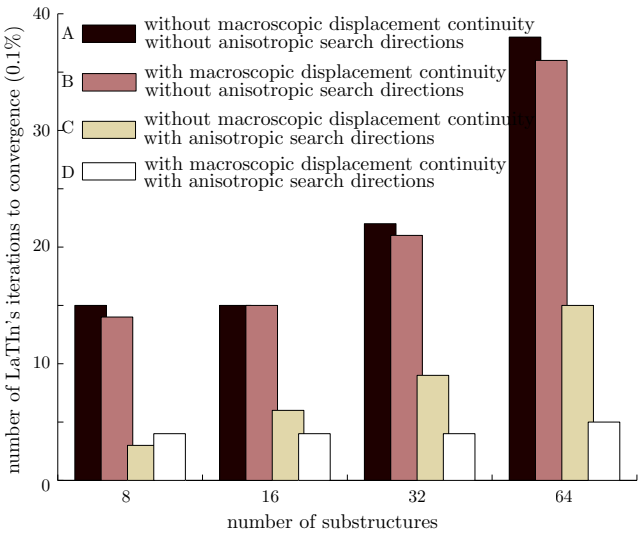


Figure 10: Influence of the search directions and of the substructuration in a bending test case

5.2 Update of the search directions

In the context of buckling and contact, the state of the structure can have large influence on its response. This implies that search directions which are optimal in one configuration can be unadapted to others, leading to a drop in the convergence rate. Strategies to update the search directions according to the status of the interfaces must then be settled. Typically for cohesive interfaces, the optimal value is related to the current stiffness of the interface (which depends on the damage state); for fully delaminated interfaces, search directions should be large when contact occurs whereas they should be null when the crack is open.

Figure 11 presents the effect of the updating of the search direction according to the damage state. In this simple DCB case, all strategies converge. Of course, the more frequent the updates the less iterations, but each update requires recomputation and factorization of the operators so that a compromise must be found to achieve minimal CPU time.

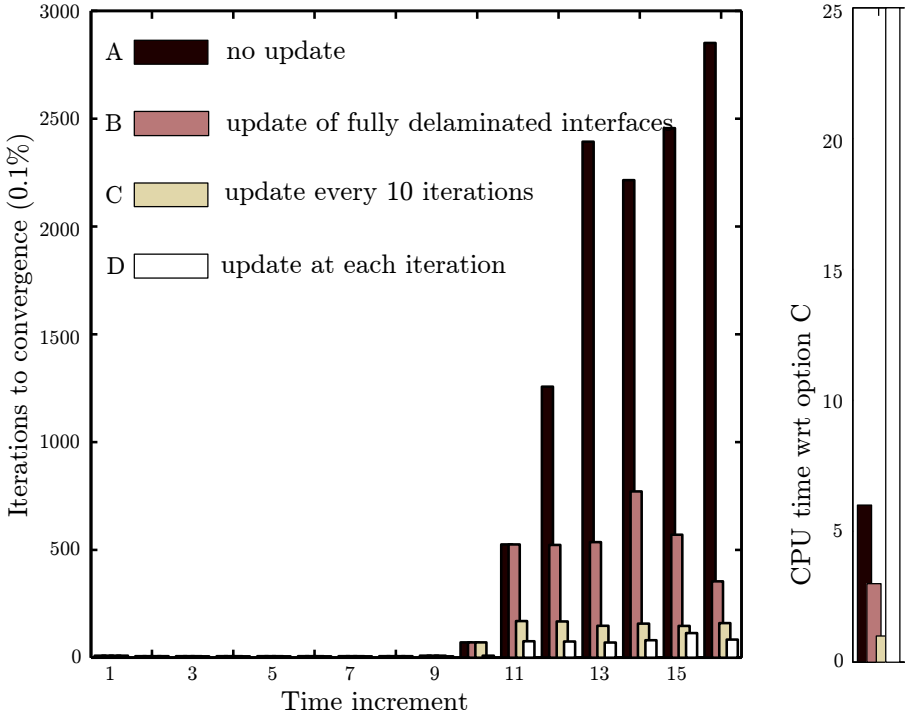


Figure 11: Iterations and CPU time to convergence when using different updating strategies

Figure 12 shows the evolution of the error for one time step of a opening contact problem: the plate is initially delaminated in its middle (large closed crack) and a compression loading is imposed. When the optimal search direction is chosen a priori, only 10 iterations are required. When the wrong one is chosen, the algorithm converges to a non-physical solution. Starting from an incorrect initial guess, a simple update procedure (every 10 iterations the parameters are adapted to current contact state) enables to converge to the solution. Note that the convergence is slow because of the unusual extension of the initial contact zone.

5.3 Pre-delaminated plate sensitive to imperfection

Let us consider the example of a pre-delaminated plate submitted to compression. The example is designed to be sensitive to imperfection. Indeed two buckling modes can be excited (Figure 13): a global buckling mode (closed interface - mode II delamination) and a local buckling (open interface - mode I delamination). The problem is split into 1280 substructures leading to 3248 interfaces with about 1.2 millions of d.o.f. The macro-problem involves 30,000 d.o.f. when the super-macro one only involves 132 d.o.f.

When the imperfection is small, the response of the buckling mode of the plate is

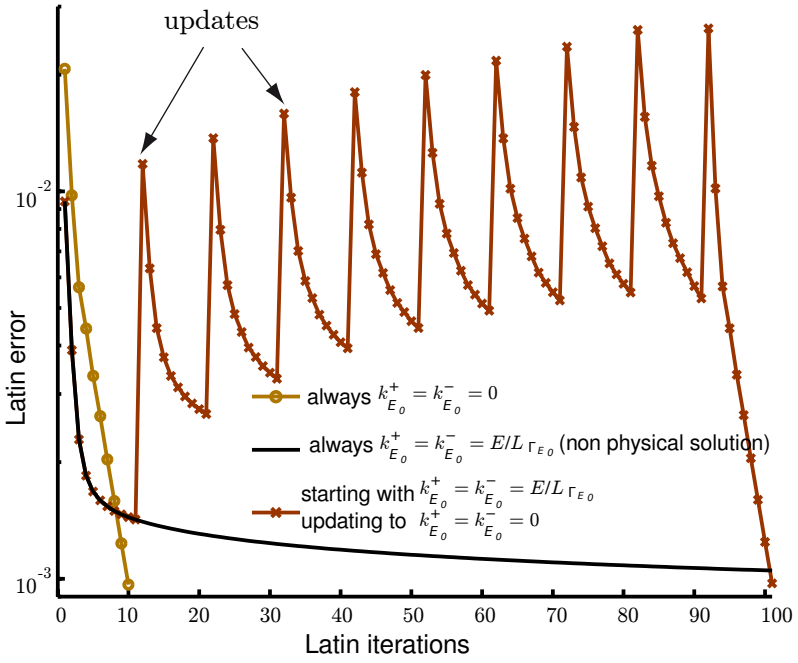


Figure 12: Evolution of the error when using different updating strategies

a global one (mode II delamination) and the delamination starts to propagate for a transversal displacement of about 0.75 mm (see Figure 14 - mode II curve).

When the level of the perturbation is at least ten times greater ($\underline{F}_d = 2$ N), a local buckling mode is excited, which leads to the propagation of the delamination for a transversal displacement of about 0.36 mm (see Figure 14 - mode I curve). Figure 15 shows the configuration and the crack propagation of the last time step.

6 Example of multiple delamination of a laminated plate in compression

The following example has been chosen because it involves all the difficulties discussed in this last section. Here, the delamination between plies and the possibility of contact and separation between delaminated surfaces in large displacements are studied (others possibles scenarios like transverse cracking are obviously ignored by the model). It is a $[0°/90°]_s$ composite plate, submitted to compression (prescribed displacement $\underline{u}_d$) and pre-delaminated (with the same initial length) on the central interface and on the upper one. A normal perturbation is applied $\underline{F}_d$ at the level of the top ply (see Figure 16). The layers are assumed to be elastic and orthotropic with the following characteristics in the referenced basis (1- fiber direction, 2- transverse direction in the ply, 3- normal direction): $E_1 = 185,500$ MPa, $E_2 = E_3 = 9,900$ MPa, $\nu_{12} = \nu_{13} = 0.34$, $\nu_{23} = 0.5$, $G_{12} = G_{13} = 6,160$ MPa, $G_{23} = 3,080$ MPa. The problem is split into 1,280 substructures leading to 3,248

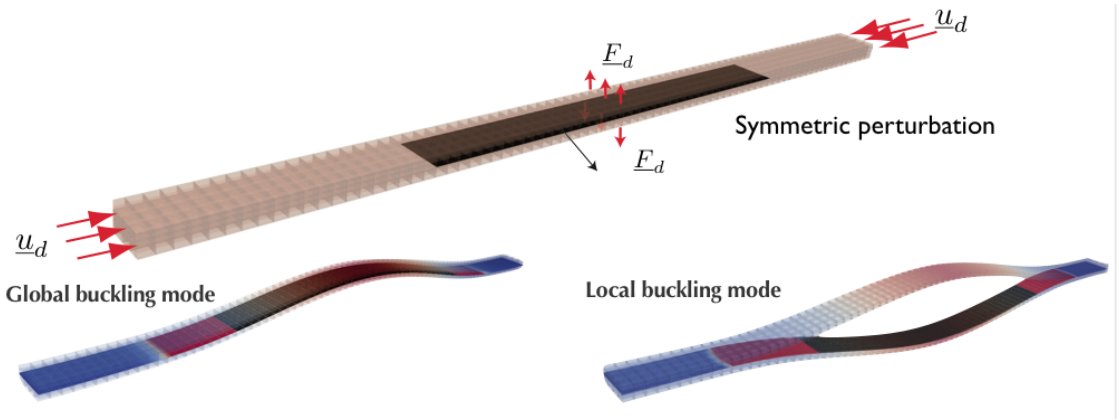


Figure 13: Pre-delaminated plate with associated potential buckling modes

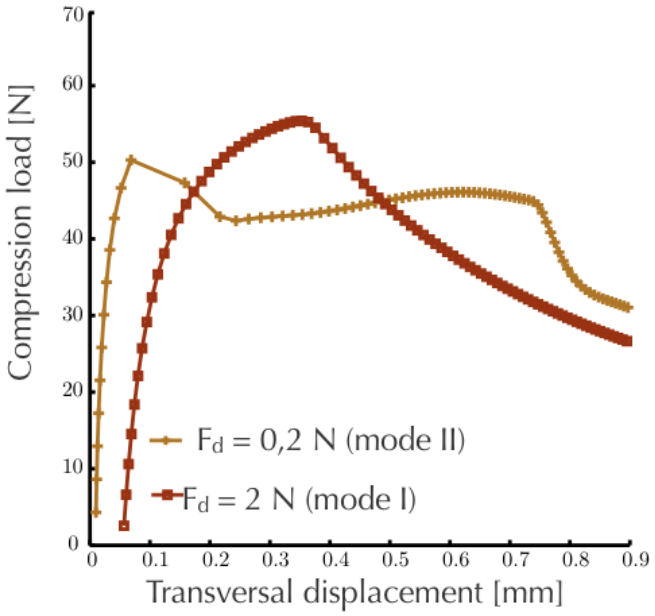


Figure 14: Load displacement curves depending on the imperfection

interfaces, 2 millions of d.o.f. are involved. The macro-problem involves 30,000 d.o.f. when the super-macro one only involves 168 d.o.f. The time interval is discretized into 120 times steps and 30 processors have been used. On Figure 17, the force-normal displacement responses for the three groups of layers in the middle of the plate are plotted and Figure 18 shows three typical configurations. The top ply buckles first for an applied load of 80 N leading to a non symmetric configuration (configuration A). In this configuration a loss of contact between the two upper plies and the lower one is observed. When the applied load reaches 100 N, the central and lower plies buckle which leads to a recontact between the upper

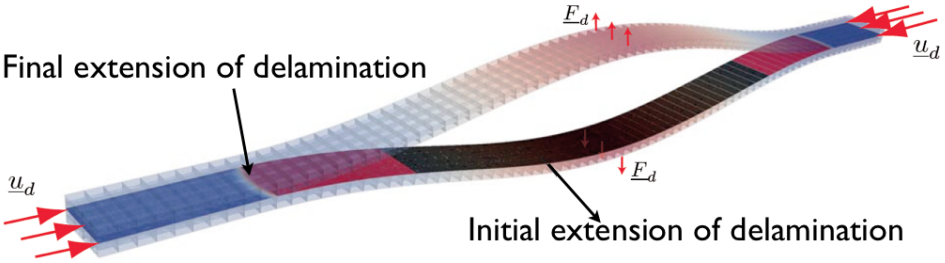


Figure 15: Final state of the plate in case of local buckling

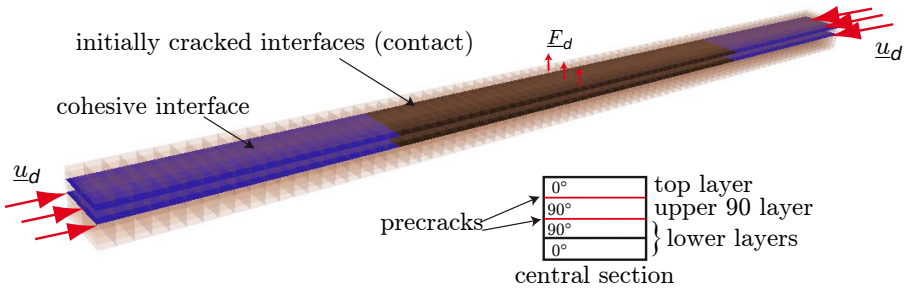


Figure 16: Configuration of the specimen

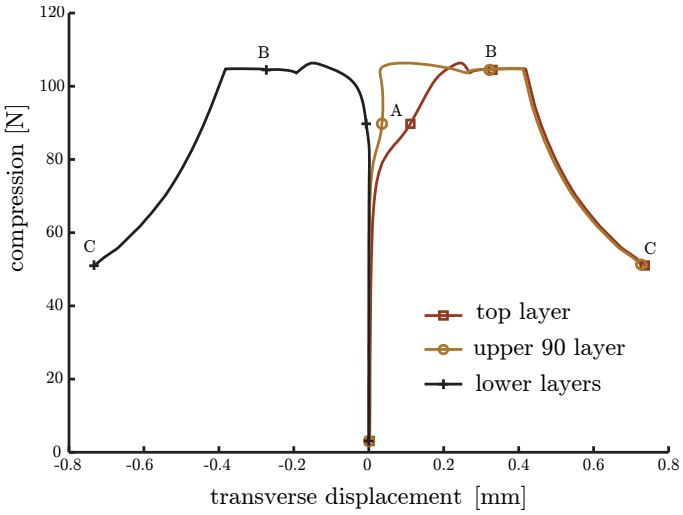


Figure 17: Displacement of three sets of plies during the experiment

plies (configuration B). Finally, the configuration C is symmetric. For a value of the transverse displacement of about 0.5 mm, the delamination starts to propagate in the central interface leading to a decrease of the reaction forces.

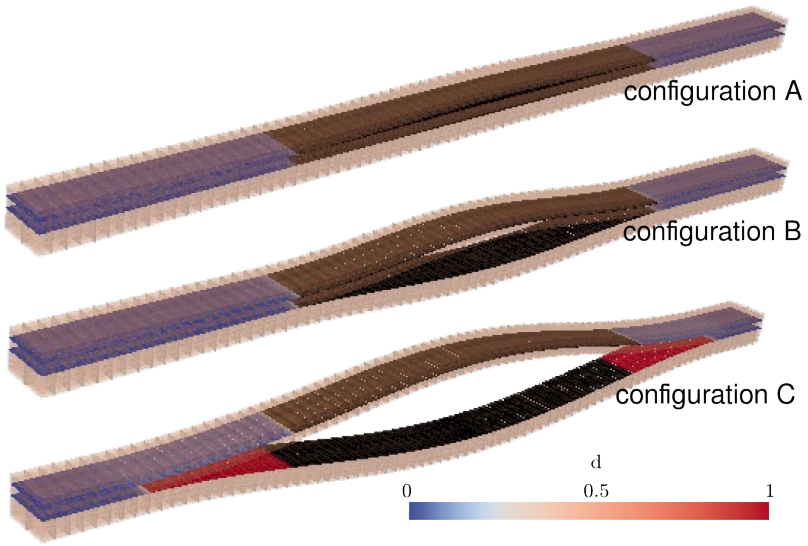


Figure 18: Three configurations. A: all plies separated, B: recontact, C: ruin

7 Conclusion

In order to circumvent the numerous problems associated with the large scale simulation of delamination problems, we have developed a multi-scale framework, wherein a Domain Decomposition approach is optimized to deal with the severe non-linearities. A lot of work is still required to make the approach more generic. The problem of code development is also an issue that needs to be taken into account when developing such dedicated methods. The challenge we would like to address is the one of the Virtual Delamination Testing, in a fully integrated sense. Nevertheless, those studies have been conducted by restricting the potential damage modes of the laminate to delamination. However, it is known that in many cases, like in the case of low-velocity impacts or in the case of the analysis of the tolerance to defects, lumping the damages onto the interfaces cannot reproduce the delamination patterns adequately. In fact, these patterns are often driven by the interaction between transverse cracking and delamination.

Two main approaches are followed in this more general case. In (Lubineau, Ladevèze, and Marsal, 2007) a micro analysis leading to a meso-scale model where a non-local coupling between the interface and the adjacent plies associated with transverse-cracking / delamination interaction has been developed. A validation of this approach by making use of open-holed tests, which exhibits very different delamination patterns due to scale effect (Green, Wisnom, and Hallett, 2007) can be found in (Abisset, Daghia, and Ladevèze, 2011). The non-locality of the ply model requires to have all the information throughout the thickness of a given ply within

a unique sub-domain. This is not a problem with the physical decomposition at the meso-scale proposed in the paper. However, it becomes an issue if one makes use of an automatic decomposition of the structure, as was addressed in (Bordeu, Boucard, and Lubineau, 2010). Another approach aims at introducing the transverse cracks explicitly, or at least the transverse cracks that have a strong influence on the delamination pattern (van der Meer, Sluys, Hallett, and Wisnom, 2012) by means of X-FEM like technique. In the framework proposed in this paper, following this direction requires to development an hybrid multi-scale X-FEM method, as was proposed in (Guidault, Allix, Champaney, and Cornuault, 2008). However, the complexity and highly-intrusive character of such an approach is an issue in terms of code development. A promising alternative is to perform a local enrichment by means of a non-intrusive coupling of the structural solver with specialized external solvers dedicated to specific failure modes, as was proposed in (Gendre, Allix, Gosselet, and Comte, 2009). For instance, one can couple the tools presented in this paper with the one developed to simulate a micro-model where all the cracks are introduced explicitly (Violeau, Ladevèze, and Lubineau, 2009). This idea was successfully applied in (Daghia and Ladevèze, 2012-to appear).

Acknowledgement: Part of the research leading to these results has received funding from the European Community's Seventh Framework Program FP7/2007-2013 under grant agreement n°213371.

References

Abisset, E.; Daghia, F.; Ladevèze, P. (2011): On the validation of a damage mesomodel for laminated composites by means of open-hole tensile tests on quasi-isotropic laminates. *Composites Part A: Applied Science and Manufacturing*, vol. 42, no. 10, pp. 1515 – 1524.

Allix, O.; Blanchard, L. (2006): Mesomodeling of delamination: towards industrial applications. *Composites Science and Technology*, vol. 66, no. 6, pp. 731–744.

Allix, O.; Corigliano, A. (1996): Modeling and simulation of crack propagation in mixed-modes interlaminar fracture specimens. *International Journal of Fracture*, vol. 77, no. 2, pp. 111–140.

Allix, O.; Kerfriden, P.; Gosselet, P. (2010): On the control of the load increments for a proper description of multiple delamination in a domain decomposition framework. *International Journal for Numerical Methods in Engineering*, vol. 83, pp. 1518–1540.

Allix, O.; Kerfriden, P.; Gosselet, P. (2010): A relocalization technique for the multiscale computation of delamination in composite structures. *Computer Modeling in Engineering and Sciences*, vol. 55, no. 3, pp. 271–292.

Allix, O.; Ladevèze, P. (1992): Interlaminar interface modelling for the prediction of delamination. *Composite Structures*, vol. 22, no. 4, pp. 235–242.

Belytschko, T.; Loehnert, S.; Song, J.-H. (2007): Multiscale aggregating discontinuities: A method for circumventing loss of material stability. *International Journal for numerical Methods in Engineering*, vol. 73, pp. 869–894.

Bordeu, F.; Boucard, P.-A.; Lubineau, G. (2010): A mesoscale model for damage, cracking and delamination prediction in composite materials. *Science and Engineering of Composite Materials*, vol. 17, no. 4, pp. 271–282.

Cai, X.-C.; Keyes, D. (2002): Nonlinearly preconditioned inexact newton algorithms. *SIAM Journal on Scientific Computing*, vol. 24, pp. 183–200.

Camanho, P. P.; Dávila, C. G. (2002): Mixed-mode decohesion finite elements for the simulation of delamination in composite materials. Technical Paper 211737, NASA, 2002.

Champaney, L.; Boucard, P. A.; Guinard, S. (2008): Adaptive multi-analysis strategy for contact problems with friction. *Computational Mechanics*, vol. 42, no. 2, pp. 305–315. Workshop on Challenges in Computational Mechanics, Cachan, France, May, 2006.

Coenen, E.; Kouznetsova, V.; Bosco, E.; Geers, M. (2012): A multi-scale approach to bridge microscale damage and macroscale failure: a nested computational homogenization-localization framework. *International Journal of Fracture*, pp. 1–22. 10.1007/s10704-012-9765-4.

Cresta, P.; Allix, O.; Rey, C.; Guinard, S. (2007): Nonlinear localization strategies for domain decomposition methods: Application to post-buckling analyses. *Computer Methods in Applied Mechanics and Engineering*, vol. 196, pp. 1436–1446.

Daghia, F.; Ladevèze, P. (2012-to appear): A micro-meso computational strategy for the prediction of the damage and failure of laminates. *Composite structures*.

de Moura, M.; Goncalves, J.; Marques, A.; DeCastro, P. (1997): Modeling compression failure after low velocity impact on laminated composites using interface elements. *Journal of Composite Materials*, vol. 31, no. 15, pp. 1462–1479.

Dostál, Z.; Horák, D.; Vlach, O. (2007): Feti-based algorithms for modelling of fibrous composite materials with debonding. *Mathematics and Computers in Simulation*, vol. 76, pp. 57–64.

Efendiev, Y.; Galvis, J. (2011): A domain decomposition preconditioner for multiscale high-contrast problems. In *Domain Decomposition Methods in Science and Engineering XIX*.

Farhat, C.; Mandel, J.; Roux, F. X. (1994): Optimal convergence properties of the feti domain decomposition method. *Computer Methods in Applied Mechanics and Engineering*, vol. 115, no. 3-4, pp. 365 – 385.

Feyel, F. (2003): A multilevel finite element (fe2) to describe the response of highly non-linear structures using generalized continua. *Computer Methods in Applied Mechanics and Engineering*, vol. 192, pp. 3233–3244.

Fish, J.; Shek, K.; Pandheeradi, M.; Shephard, M. S. (1997): Computational plasticity for composite structures based on mathematical homogenization: Theory and practice. *Computer Methods in Applied Mechanics and Engineering*, vol. 148, no. 1-2, pp. 53 – 73.

Gendre, L.; Allix, O.; Gosselet, P.; Comte, F. (2009): Non-intrusive and exact global/local techniques for structural problems with local plasticity. *Computational Mechanics*, vol. 44, no. 2, pp. 233–245.

Germain, N.; Besson, J.; Feyel, F.; Gosselet, P. (2007): High-performance parallel simulation of structure degradation using non-local damage models. *International Journal for Numerical Methods in Engineering*, vol. 71, pp. 253–276.

Green, B. G.; Wisnom, M. R.; Hallett, S. R. (2007): An experimental investigation into the tensile strength scaling of notched composites. *Composites Part A: Applied Science and Manufacturing*, vol. 38, no. 3, pp. 867–878.

Guidault, P.; Allix, O.; Champaney, L.; Navarro, J. (2007): A two-scale approach with homogeneisation for the computation of cracked structures,. *Computer and Structures*, vol. 85, no. 17-18, pp. 1360–1371.

Guidault, P. A.; Allix, O.; Champaney, L.; Cornuault, C. (2008): A multiscale extended finite element method for crack propagation. *Computer Methods in Applied Mechanics and Engineering*, vol. 197, no. 5, pp. 381–399.

Herakovich, C. T. (2012): Mechanics of composites: A historical review. *Mechanics Research Communications*, vol. 41, pp. 1–20.

Hill, R. (1963): Elastic properties of reinforced solids: some theoretical principles. *Journal of the Mechanics and Physics of Solids*, vol. 11, no. 5, pp. 357–372.

Huerta, A.; Rodríguez-Ferran, A.; Diez, P. (2002): Error estimation and adaptivity for nonlinear fe analysis. *International Journal of Applied Mathematics and Computer Science*, vol. 12, no. 1, pp. 59–70.

Kerfriden, P.; Allix, O.; Gosselet, P. (2009): A three-scale domain decomposition method for the 3d analysis of debonding in laminates. *Computational Mechanics*, vol. 44, no. 3, pp. 343–362.

Klawonn, A.; Rheinbach, O.; Wohlmuth, B. (2007): Dual-primal iterative substructuring for almost incompressible elasticity. *Springer Lecture Notes in Computer Science and Engineering*, vol. 55, pp. 399–406.

Kouznetsova, V.; Geers, M. G. D.; Brekelmans, W. A. M. (2002): Multi-scale constitutive modelling of heterogeneous materials with a gradient-enhanced computational homogenization scheme. *International Journal for Numerical Methods in Engineering*, vol. 54, pp. 1235–1260.

Kruis, J.; Bittnar, Z. (2007): Reinforcement-matrix interaction modelled by feti method. In *Domain Decomposition Methods in Science and Engineering XVII*.

Ladevèze, P.; Dureisseix, D. (1999): Une nouvelle stratégie de calcul micro/macro en mécanique des structures. *Comptes Rendus de l'Académie des Sciences-Series IIB-Mechanics-Physics-Astronomy*, vol. 327, no. 12, pp. 1237–1244.

Ladevèze, P.; Loiseau, O.; Dureisseix, D. (2001): A micro–macro and parallel computational strategy for highly heterogeneous structures. *International Journal for Numerical Methods in Engineering*, vol. 52, pp. 121–138.

Ladevèze, P.; Moës, N.; Douchin, B. (1999): Constitutive relation error estimators for (visco)plastic finite element analysis with softening. *Computer Methods in Applied Mechanics and Engineering*, vol. 176, pp. 247–264.

Ladevèze, P.; Néron, D.; Gosselet, P. (2007): On a mixed and multiscale domain decomposition method. *Computer Methods in Applied Mechanics and Engineering*, vol. 196, pp. 1526–1540.

Larsson, F.; Runesson, K.; Hansbo, P. (2003): Time finite elements and error computation for (visco)plasticity with hardening or softening. *International journal for numerical methods in engineering*, vol. 56, pp. 2213–2232.

Le Tallec, P. (1994): Domain decomposition methods in computational mechanics. *Computational Mechanics Advances*, vol. 1, pp. 121–220.

Lloberas-Valls, O.; Rixen, D.; Simone, A.; Sluys, L. (2012): Multiscale domain decomposition analysis of quasi-brittle heterogeneous materials. *International Journal for Numerical Methods in Engineering*, vol. 89, pp. 1337–1366.

Lubineau, G.; Ladevèze, P.; Marsal, D. (2007): Towards a bridge between the micro- and mesomechanics of delamination for laminated composites. *Composites Science and Technology*, vol. 66, no. 6, pp. 698–712.

Mandel, J. (1993): Balancing domain decomposition. *Communications on Numerical Methods in Engineering*, vol. 9, pp. 233–241.

Massart, T.; Peerlings, R.; Geers, M. (2007): Structural damage analysis of masonry walls using computational homogenization. *International Journal of Damage Mechanics*, , no. 16, pp. 199–226.

Parret-Fréaud, A.; Rey, C.; Gosselet, P.; Feyel, F. (2010): Fast estimation of discretization error for fe problems solved by domain decomposition. *Computer Methods in Applied Mechanics and Engineering*, vol. 199, no. 49-52, pp. 3315–3323.

Pebrel, J.; Rey, C.; Gosselet, P. (2008): A nonlinear dual domain decomposition method: application to structural problems with damage. *International Journal of Multiscale Computational Engineering*, vol. 6, no. 3, pp. 251–262.

Pellegrino, C.; Galvanetto, U.; Schrefler, B. A. (1999): Numerical homogenization of periodic composite materials with non-linear material components. *International Journal for Numerical Methods in Engineering*, vol. 46, pp. 1609–1637.

Qiu, Y.; Crisfield, M. A.; Alfano, G. (2001): An interface element formulation for the simulation of delamination with buckling. *Engineering Fracture Mechanics*, vol. 68, no. 16, pp. 1755–1776.

Saavedra, K.; Allix, O.; Gosselet, P. (2012): On a multiscale strategy and its optimization for the simulation of combined delamination and buckling. *International Journal for Numerical Methods in Engineering*, vol. 91, no. 7, pp. 772–798.

Sánchez-Palencia, E. (1980): *Non homogeneous media and vibration theory*, volume 127 of *Lecture Notes in Physics*. Springer Verlag, Berlin.

Schellekens, J. C. J.; de Borst, R. (1994): Free edge delamination in carbon-epoxy laminates: a novel numerical/experimental approach. *Composite Structures*, vol. 28, no. 4, pp. 357–374.

van der Meer, F. P.; Sluys, L. J.; Hallett, S. R.; Wisnom, M. R. (2012): Computational modeling of complex failure mechanisms in laminates. *Journal of Composite Materials*, vol. 46, no. 5, pp. 603–623.

Violeau, D.; Ladevèze, P.; Lubineau, G. (2009): Micromodel-based simulations for laminated composites. *Composites Science and Technology*, vol. 69, no. 9, pp. 1364–1371.

Xu, X.; Needleman, A. (1994): Numerical simulations of fast crack growth in brittle solids. *Journal of the Mechanics and Physics of Solids*, vol. 42, pp. 1397–1434.

Zohdi, T. I.; Oden, J. T.; Rodin, G. J. (1996): Hierarchical modeling of heterogeneous bodies. *Computer methods in applied mechanics and engineering*, vol. 138, no. 1-4, pp. 273–298.

Zohdi, T. I.; Wriggers, P. (2008): *An introduction to computational micromechanics*, volume 20 of *Lecture Notes in Applied and Computational Mechanics*. Springer Verlag.

B Brouillon de [Oumaziz *et al.*, 2016], revisite de la méthode Latin appliquée au calcul d'assemblage

Cet article donne une présentation de la méthode mixte de décomposition de domaine LaTIn. Plus précisément, nous utilisons les notations blocs maintenant classiques pour les méthodes FETI et BDD, pour donner une version la plus dépouillée possible de la méthode. Afin d'être mise en œuvre dans le code généraliste code_aster qui ne possède pas de conditions aux limites mixtes en mécanique, nous utilisons un ajout de matière pour réaliser les conditions de Robin (généralisées). Ce recouvrement virtuel n'empêche pas de traiter des interfaces de contact frottant comme cela est détaillé. L'article est complété par une annexe (qui sera incluse dans une autre publication) relative à la mise en place du problème grossier (*aka* LaTIn multiéchelle).

Computational Mechanics manuscript No.
(will be inserted by the editor)

Paul Oumaziz · Pierre Gosselet · Pierre-Alain Boucard · Stéphane Guinard

A non-invasive implementation of a mixed domain decomposition method for frictional contact problems

Received: date / Accepted: date

Abstract A non-invasive implementation of the Latin domain decomposition method for frictional contact problems is described. The formulation implies to deal with mixed (Robin) conditions on the faces of the sub-domains, which is not a classical feature of commercial software. Therefore we propose a new implementation of the linear stage of the Latin method with a non-local search direction built as the stiffness of a layer of elements on the interfaces. This choice enables us to implement the method within the opensource software Code_Aster, and to derive 2D and 3D examples with similar performance as standard Latin method.

Keywords Latin method · Domain decomposition · Contact problems · Assemblies

1 Introduction

Industrialists more and more consider conducting simulation of their products at the scale of the (sub)system instead of the scale of the individual part. To do so, it is crucial to use robust methods capable of handling large contact zones with friction. Such interface law are highly non-regular and complex to handle numerically. The contact formulations often involve Lagrange multipliers [35; 34; 37] and regularization [1; 38]. When the problem can be tracked back to a linear complementary problem, direct solvers, like the Lemke method, are available. However they do not scale efficiently with the number of degrees of freedom. A very general family of iterative methods to handle contact is formed by the Uzawa algorithms [24; 19].

Industrial mechanical simulations also involve large numbers of degrees of freedom. Domain decomposition techniques are suitable to tackle this issue: by distributing the computation over parallel hardware architecture, they lift the memory limits of direct solvers. Moreover when well configured they may accelerate the resolution. Main domain decomposition iterative solvers are: Schwarz methods [17], Balancing Domain Decomposition [29], Finite Element Tearing and Interconnecting [14; 15; 13] as well as their constrained counterparts [8; 16].

However tackling contact within domain decomposition method is not trivial, in particular if one wants the contact interfaces to match the interfaces between sub-domains. One notable adaptation of FETI was proposed in [9; 10; 11; 12]. Robustness of domain decomposition methods was recently

Paul Oumaziz · Pierre Gosselet · Pierre-Alain Boucard
LMT-Cachan, ENS-Cachan/CNRS/Univ. Paris-Saclay,
61 avenue du président Wilson, 94235 Cachan, FRANCE
Tel.: +33147402402, +33147405333, +33147405330, Fax: +33147402785
E-mail: oumaziz@lmt.ens-cachan.fr, gosselet@lmt.ens-cachan.fr, boucard@lmt.ens-cachan.fr
Stéphane Guinard
Computational Structural Mechanics,AGI for Airbus Group Innovations,
18 rue Marius Terce, 31300 Toulouse, FRANCE,
E-mail: stephane.guinard@airbus.com

significantly improved by the introduction of adapted augmentation spaces obtained through prior quasi-local analysis [36] or multi-preconditioned approaches [21]. However these methods are not yet implemented in commercial software due to their invasive nature and the significant implementation effort required to include them in industrial software.

The Latin method [25] is particularly adapted to treat contact with domain decomposition. In this method, sub-domains and interfaces are considered as different entities with their own mechanical behavior. Under this assumption interfaces behavior corresponds to the nonlinear contact laws. The principle of the method is to separate the linear equations over the sub-domains from the nonlinear local formulation of contact problems, and to formulate the problem as the search of a fixed point.Two successive problems are defined and solved at each iteration: independent linear systems set on sub-domains (linear stage) followed by independent nonlinear pointwise equation on the interfaces with explicit solutions (local stage). The steps of the methods are linked by alternating directions [18] which can be interpreted as interface stiffnesses (Robin conditions), this connects the Latin method with mixed domain decomposition methods, also known as non-overlapping optimized Schwarz techniques [17].

The current challenge is to implement this method inside a commercial software. The difficulty comes from the treatment of the mixed conditions on the boundary of the sub-domains. Indeed, sub-domains computation with Robin's boundary condition is not a classic feature of commercial mechanical simulation software. Thus we propose a non-invasive approach which enables its implementation. The Robin's condition is dealt with a "non-local" stiffness over the faces of the sub-domains. The search direction is built as a mechanical problem on a layer of elements located on the interfaces. Even though the non-invasive aspect is crucial to the linear stage, this modification implies reformulation of the one from the local stage.

The paper is organized as follow: the reference problem is presented in sub-structured form in section 2. A full description of the non-invasive Latin approach is given in section 3. The contact and friction are treated in section 4. The last section focuses on the numerical studies. A comparison between a "standard" Latin method and the new non-invasive formulation is shown. An 2D academic case is presented to validate the contact computation, then a 3D assembly with frictional contact is treated.

2 Reference problem in substructured form

2.1 Substructured problem

2.1.1 Notations

We consider a family of non-overlapping sub-domains $(\Omega_E)_{E\in[1,N]}$ of $\mathbb{R}^d$ ($d = 1$, 2 or 3). Each sub-domain corresponds to a part of an assembly. We assume each part is constituted by an isotropic linear elastic material whose Hooke's tensor $\mathbb{K}$ is characterized by the Young modulus E and the Poisson ratio ν. Small perturbations and quasi-static isotherm evolutions are assumed. The displacement field is designated by $\underline{u}$, the stress field by $\boldsymbol{\sigma}$ and the strain field by $\boldsymbol{\varepsilon}$. When needed the subscript E is used for the restriction to sub-domain Ω_E.

Each part Ω_E is subjected to Neumann boundary conditions $\underline{F}_d$ on $\partial_F\Omega_E$, Dirichlet boundary conditions $\underline{u}_d$ on $\partial_u\Omega_E$ and body force $\underline{f}_d$. It also interacts with any neighboring sub-domain $\Omega_{E'}$ through the interface $\Gamma_{EE'} \subset \partial\Omega_E$. Let $\mathcal{E}$ be the set of all sub-domains and $\mathcal{G}$ the set of all interfaces.

Remark 1 Of course we have $\partial\Omega_E \cap \partial\Omega_{E'} \subset \Gamma_{EE'}$, but $\Gamma_{EE'}$ also contains all portions of the boundary which may enter in contact. Thanks to the small displacement hypothesis, this information needs not be updated during the resolution (one piece of the boundary can only interact with one given neighbor).

For each interface, displacement and force fields are defined on the two sides. They are respectively written $(\underline{W}_{EE'}, \underline{W}_{E'E}) \in (H^{1/2}(\Gamma_{EE'}))^2$ and $(\underline{F}_{EE'}, \underline{F}_{E'E}) \in \left(H^{-1/2}(\Gamma_{EE'})\right)^2$. $\underline{F}_{EE'}$ represents the force exerted by the sub-domain E' on the sub-domain E. In the case of quasi-static evolution

velocity fields $\left(\dot{\underline{W}}_{EE'}, \dot{\underline{W}}_{E'E}\right)$ are also considered. The interface displacement field is linked to the displacement inside the sub-domain by the trace operator: $\underline{W}_{EE'} = \mathrm{tr}\left(\underline{u}_E\right)_{|\Gamma_{EE'}}$.

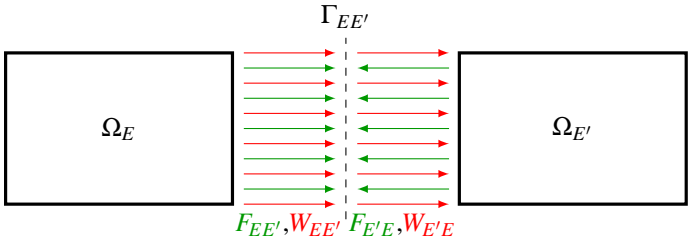


Fig. 1 Force and displacement fields of an interface

In our formulation, not only the sub-domains but also the interfaces are granted a mechanical behavior.

2.1.2 Weak formulation on the sub-domains

For any sub-domain Ω_E, let us define local admissibility space $\mathcal{U}_E = \left\{\underline{u} \in H^1(\Omega_E), \underline{u} = \underline{u}_d \text{ on } \partial_u\Omega_E\right\}$, and its associated vector space $\mathcal{U}_E^0 = \left\{\underline{u} \in H^1(\Omega_E), \underline{u} = 0 \text{ on } \partial_u\Omega_E\right\}$.

The weak formulation can be written as:

$$\forall \Omega_E \in \mathcal{E}, \begin{cases} \forall \underline{\dot{u}}_E \in \mathcal{U}_E^0, \displaystyle\int_{\Omega_E} \boldsymbol{\varpi}_E : \boldsymbol{\varepsilon}(\underline{\dot{u}}_E)\,\mathrm{d}\Omega = \int_{\Omega_E} \underline{f}_d \cdot \underline{\dot{u}}_E\,\mathrm{d}\Omega + \int_{\partial_F\Omega_E} \underline{F}_d \cdot \underline{\dot{u}}_E\,\mathrm{d}\Gamma + \sum_{E'} \int_{\Gamma_{EE'}} \underline{F}_{EE'} \cdot \underline{\dot{u}}_E \mathrm{d}\Gamma \\ \boldsymbol{\varpi}_E = \mathbb{K}_E : \boldsymbol{\varepsilon}(\underline{u}_E) \\ \underline{W}_{EE'} = \mathrm{tr}\left(\underline{u}_E\right)_{|\Gamma_{EE'}} \end{cases} \tag{1}$$

where one can recognize the conservation of the momentum of the sub-domain submitted to given loads and neighbors' interactions, the constitutive equation and the trace relation on the interfaces.

2.1.3 Interface behavior

The interface $\Gamma_{EE'}$ can be granted a mechanical behavior given by a relation of the form:

$$\underline{F}_{EE'}(\tau) = b_{EE'}\left(\dot{\underline{W}}_{E'E} - \dot{\underline{W}}_{EE'}, t < \tau\right) \tag{2}$$

Such a relation can be used to model spring interfaces or cohesive interfaces [23; 32]. Note that $b_{E'E}$ is not independent from $b_{EE'}$ because the balance of the interface must be preserved:

$$\underline{F}_{EE'} + \underline{F}_{E'E} = 0 \tag{3}$$

The interface $\Gamma_{EE'}$ can also be granted kinematic constraints, like in the case of perfect cohesion or contact. Typically a perfect interface is characterized by:

$$\begin{cases} \underline{F}_{EE'} + \underline{F}_{E'E} = 0 \\ \underline{W}_{EE'} = \underline{W}_{E'E} \end{cases} \tag{4}$$

Intuitively, perfect and contact interfaces can be considered as limit cases of mechanical behaviors (2) (think of a perfect interface as an infinitely stiff interface). In the following we use the term "interface behavior" and the notation $b_{EE'}$ for any kind of interface, except when being more specific is required.

2.2 Discrete formulation

The continuous formulation above is rather standard in the Latin literature. We now propose to completely describe the resulting discrete system, using operators and notations inspired from [20].

2.2.1 Discretization

A classical FE is used in the sub-domains for the displacement field. Let $\underline{\phi}^u_E$ be the matrix of shape function and $\mathbf{U}_E$ the vector of nodal displacement, such that $\underline{u}_E = \underline{\phi}^u_E \mathbf{U}_E$.

The interface displacement is also discretized with shape functions $\underline{\phi}^w_{EE'} : \underline{W}_{EE'} = \underline{\phi}^w_{EE'} \mathbf{W}_{EE'}$. It is assumed that $\underline{\phi}^w_{EE'} = \underline{\phi}^w_{E'E}$. In the classic description of the Latin method, interface traction ($\underline{F}_{EE'}$) is also discretized. This assumption is not useful in the case of matching grids where the key quantity is the work of the traction in the discretized displacements, that is to say the nodal reactions $\mathbf{F}_{EE'}$:

$$\forall \underline{W}^*_{EE'} = \underline{\phi}^w_{EE'} \mathbf{W}^*_{EE'}, \quad \int_{\Gamma_{EE'}} \underline{F}_{EE'} \cdot \underline{W}^*_{EE'} \mathrm{d}S = \int_{\Gamma_{EE'}} \underline{F}_{EE'} \cdot (\underline{\phi}^w_{EE'} \mathbf{W}^*_{EE'}) \mathrm{d}S = \mathbf{W}^{*T}_{EE'} \mathbf{F}_{EE'} \tag{5}$$

In the case where sub-domains have matching grids, we suggest to choose $\underline{\phi}^w_{EE'} = \mathrm{tr}\left(\underline{\phi}^u\right)_{|\Gamma_{EE'}}$. In that case we can directly build the discrete counterpart to the trace operator as a boolean matrix $\mathbf{N}_{EE'}$ such that

$$\mathbf{W}_{EE'} = \mathbf{N}_{EE'} \mathbf{U}_E \tag{6}$$

In the case where sub-domains have non-matching grids, sub-domains are connected to the interface by a mortar-like approach [3; 2] which sums up to choosing a discretization for the traction field and ensuring the following work equivalence:

$$\forall \underline{F}^*_{EE'} \in \mathcal{F}_{EE'}\left(\Gamma_{EE'}\right), \int_{\Gamma_{EE'}} \underline{F}^*_{EE'} \cdot (\underline{W}_{EE'} - (\mathrm{tr}\, \underline{u}_E)_{|\Gamma_{EE'}}) \mathrm{d}S = 0 \tag{7}$$

where the subspace $\mathcal{F}_{EE'}$ is spanned by the shape functions $\underline{\phi}^F_{EE'}$ such that $\int_{\Gamma_{EE'}} \underline{\phi}^F_{EE'}{}^T \underline{\phi}^w_{EE'}$ is invertible. This leads to a non-boolean $\mathbf{N}_{EE'}$ matrix.

To easily handle the many interfaces of one sub-domain, we define concatenated operators:

$$\mathbf{W}_E = \begin{pmatrix} \vdots \\ \mathbf{W}_{EE'} \\ \vdots \end{pmatrix}, \quad \mathbf{F}_E = \begin{pmatrix} \vdots \\ \mathbf{F}_{EE'} \\ \vdots \end{pmatrix}, \quad \mathbf{N}_E = \begin{pmatrix} \ddots & & \\ & \mathbf{N}_{EE'} & \\ & & \ddots \end{pmatrix}, \quad E' \text{ spans all neighbors} \tag{8}$$

In the end, the discrete counterpart to system (1) can be written as:

$$\begin{aligned} &\text{Subdomains:} \quad \forall \Omega_E \in \mathcal{E}, \begin{cases} \mathbf{K}_E \mathbf{U}_E = \mathbf{f}_{dE} + \mathbf{N}^T_E \mathbf{F}_E \\ \mathbf{W}_E = \mathbf{N}_E \mathbf{U}_E \end{cases} \\ &\text{Interfaces:} \quad \forall \Gamma_{EE'} \in \mathcal{G}, \begin{cases} \mathbf{F}_{EE'} + \mathbf{F}_{E'E} = 0 \\ \mathbf{F}_{EE'}(\tau) = \mathbf{b}_{EE'}\left(\dot{\mathbf{W}}_{E'E} - \dot{\mathbf{W}}_{EE'}, t < \tau\right) \end{cases} \end{aligned} \tag{9}$$

where $\mathbf{f}_{dE}$ stands for the generalized forces associated to given loads and displacements (it is assumed that non zeros Dirichlet conditions were substituted).

Remark 2 The relation between displacement and velocity has not been written to avoid the overloading of the expression. This relation will be implied in the following of the article.

2.2.2 Global notations

Global variables are used to simplify the writing of all the relations. Let's consider a given discrete interface variable $\mathbf{x}_{EE'}$. As said earlier $\mathbf{x}_E$ represents the gathering of all the $(\mathbf{x}_{EE'})_{E'}$. $\mathbf{x}$ will represent the gathering of all the $\mathbf{x}_E$ defined on each sub-domain Ω_E, same procedure applies to operators:

$$\mathbf{x} = \begin{pmatrix} \vdots \\ \mathbf{x}_E \\ \vdots \end{pmatrix}, \qquad \mathbf{K} = \begin{pmatrix} \ddots & & 0 \\ & \mathbf{K}_E & \\ 0 & & \ddots \end{pmatrix}, \text{ where } E \text{ spans all sub-domains} \tag{10}$$

2.2.3 Sum and difference assembly operator

We introduce operators which permit to make neighboring sub-domains communicate: the operator $\mathbf{A}$ makes sum of interface vectors whereas the operator $\mathbf{B}$ makes differences:

$$\begin{aligned}(\mathbf{AF})_{|\Gamma_{\{EE'\}}} &= \mathbf{F}_{EE'} + \mathbf{F}_{E'E} \\ (\mathbf{BW})_{|\Gamma_{\{EE'\}}} &= \mathbf{W}_{E'E} - \mathbf{W}_{EE'}\end{aligned} \tag{11}$$

Because operators $\mathbf{A}$ and $\mathbf{B}$ assemble contributions from neighboring sub-domains, their range lies in an "in-between interface" written $\Gamma_{\{EE'\}}$. There is an arbitrary sign convention in the difference operator which plays no role in the following. Note that because multiple points are handled in the matrix $\mathbf{N}$, operators $\mathbf{A}$ and $\mathbf{B}$ are full-ranked.

Let $\mathbb{R}^{\{\Gamma\}} = \text{Range}(\mathbf{A}) = \text{Range}(\mathbf{B})$ stands for the vector space supported by the "in-between interface" and $\mathbb{R}^{\Gamma} = \text{Range}(\mathbf{N})$ be the vector space constituted by the boundary degrees of freedom (on $(\Gamma_{EE'})_{E,E'}$).

The following property is fundamental to handle interface quantities:

Proposition 1 $\text{Range}(\mathbf{B}^T) \overset{\perp}{\oplus} \text{Range}(\mathbf{A}^T) = \mathbb{R}^{\Gamma}$.

When needed, for regular enough interfaces, we will split the normal n and tangential t components.

2.2.4 Discrete problem

Under these notations, the discrete problem equivalent to the sub-structured problem is:

$$\begin{cases} \mathbf{KU} = \mathbf{f}_d + \mathbf{N}^T\mathbf{F} & \text{Equilibrium of the sub-domains} \\ \mathbf{W} = \mathbf{NU} & \text{Trace of the sub-domain displacement} \\ \mathbf{F}(\tau) = \mathbf{B}^T\mathbf{b}(\mathbf{B}\dot{\mathbf{W}}, t < \tau) & \text{Interfaces' behavior} \end{cases} \tag{12}$$

This notation for the interface's behavior makes it clear that the equilibrium is ensured: $\mathbf{AF} = \mathbf{AB}^T\mathbf{b}(\mathbf{BW}) = 0$. In the case of perfect interfaces, the equations become:

$$\begin{cases} \mathbf{AF} = 0 \\ \mathbf{BW} = 0 \end{cases} \tag{13}$$

Contact is detailed in section 4.

3 The non-invasive quasi-static Latin method

3.1 Principle of the Latin method

3.1.1 Separation of the equations

In order to solve the problem (12), the Latin method [25] is applied. The first idea of the Latin method is to separate the equations in two groups. The first one concerns the sub-domains equations, which in our case are all linear. The second one gathers the equations which manage the behavior of the interfaces, these equations are all local (i.e. pointwise) in space and time.

Remark 3 Historically the separation of equations is made between linear and local equations. We will keep this denomination even if sub-domain/interface would be more appropriate in our case.

These two groups define respectively two sets of partial solutions to the problem: $\mathcal{A}$ and $\mathcal{L}$. The solution of the whole problem is the intersection of these two sets:

$$\mathcal{A} : (\mathbf{F}, \mathbf{W}) \text{ solutions to } \begin{cases} \mathbf{KU} = \mathbf{f}_d + \mathbf{N}^T\mathbf{F} \\ \mathbf{W} = \mathbf{NU} \end{cases} \tag{14}$$

$$\mathcal{L} : (\widehat{\mathbf{F}}, \widehat{\mathbf{W}}) \text{ solutions to } \mathbf{F}(\tau) = \mathbf{B}^T \mathbf{b}(\mathbf{B}\dot{\mathbf{W}}, t < \tau) \tag{15}$$

The set $\mathcal{A}$ is an affine space often called space of admissible fields. The set $\mathcal{L}$ is in general a manifold, in the case of perfect interfaces, it is a vector space:

$$\mathcal{L} : (\widehat{\mathbf{F}}, \widehat{\mathbf{W}}) \text{ solutions to } \begin{cases} \mathbf{A}\widehat{\mathbf{F}} = 0 \\ \mathbf{B}\widehat{\mathbf{W}} = 0 \end{cases} \tag{16}$$

3.1.2 Iterations

In order to reach the solution of the whole problem an iterative method is applied where partial solutions are found alternatively in each set $\mathcal{A}$ and $\mathcal{L}$.

The so-called *local stage* consists in, starting from a partial solution $s_n = (\mathbf{F}_n, \mathbf{W}_n, \dot{\mathbf{W}}_n) \in \mathcal{A}$, searching for a partial solution $\widehat{s}_n = (\widehat{\mathbf{F}}_n, \widehat{\mathbf{W}}_n, \widehat{\dot{\mathbf{W}}}_n) \in \mathcal{L}$. This is made possible by enforcing a search direction of the form:

$$\mathbf{F} - \widehat{\mathbf{F}} - \mathbf{k}_V^+ \left(\dot{\mathbf{W}} - \widehat{\dot{\mathbf{W}}} \right) = 0 \tag{17}$$

where $\mathbf{k}_V^+$ is an operator which can be chosen by the user. In order to benefit the local character of the equations which define $\mathcal{L}$, $\mathbf{k}_V^+$ is often chosen to be diagonal.

The so-called *linear stage* consists in, starting from a partial solution $\widehat{s}_n = (\widehat{\mathbf{F}}_n, \widehat{\mathbf{W}}_n, \widehat{\dot{\mathbf{W}}}_n) \in \mathcal{L}$, searching for a partial solution $s_{n+1} = (\mathbf{F}_{n+1}, \mathbf{W}_{n+1}, \widehat{\dot{\mathbf{W}}}_{n+1}) \in \mathcal{A}$. This in made possible by enforcing a search direction of the form:

$$\mathbf{F}_{n+1} - \widehat{\mathbf{F}}_n + \mathbf{k}_V^- \left(\dot{\mathbf{W}}_{n+1} - \widehat{\dot{\mathbf{W}}}_n \right) = 0 \tag{18}$$

where $\mathbf{k}_V^-$ is an operator which can be chosen by the user.

A relaxation is often applied at the end of linear stages:

$$s_{n+1} \leftarrow s_n + \alpha(s_{n+1} - s_n) \tag{19}$$

with $0 < \alpha \leqslant 1$.

The iterations are graphically represented on Figure 2.

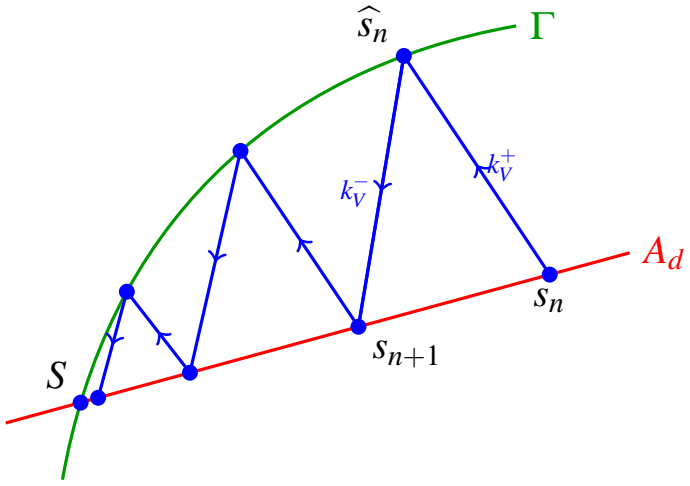


Fig. 2 Latin scheme

Proposition 2 *The convergence of the iterations is proved [25] when* $\mathbf{k}^- = \mathbf{k}^+$ *are symmetric definite positive operator and* $\alpha = 0.5$ *for maximal monotone behaviours* $\mathbf{b}$.

3.1.3 Interpretation of search directions

Considering an interface $\Gamma_{EE'}$ search directions are often written at the continuous level as a relation of the form:

$$\underline{F}_{EE'} - \widehat{\underline{F}}_{EE'} \pm {k_V}^{EE'}(\underline{\dot{W}}_{EE'} - \widehat{\underline{\dot{W}}}_{EE'}) = 0 \tag{20}$$

This leads to the following considerations:

- In its most general meaning, ${k_V}^{EE'}$ is an operator from $H^{1/2}(\Gamma_{EE'})$ to $H^{-1/2}(\Gamma_{EE'})$. Choosing it to be linear is a classical hypothesis to preserve the linearity of the sub-domains' computations. But it has no reason from being local.
- In the literature, a relation like (20), is often referred to as a generalized Robin condition. A possible way to implement it is to use Ventcell conditions [22].
- In [7] it was proposed to accept the non-locality, and make it clearly appear by choosing a search direction of the following form:

  $$J_{EE'}(\underline{F}_{EE'} - \widehat{\underline{F}}_{EE'}) \pm k_V'^{EE'}(\underline{\dot{W}}_{EE'} - \widehat{\underline{\dot{W}}}_{EE'}) = 0 \tag{21}$$

  where $J_{EE'}$ is Riesz isomorphism from $H^{-1/2}(\Gamma_{EE'})$ to $H^{1/2}(\Gamma_{EE'})$, $k_V'^{EE'}$ can easily be chosen to be scalar. This choice leads to dense matrices of search direction which are not computationally tractable.
- In practice, ${k_V}^{EE'}$ is often taken as a scalar or a local orthotropic operator [33] (without taking the care to use Riesz' isomorphism). In that case the search direction is a classical Robin condition.
- Even when ${k_V}^{EE'}$ is chosen to be scalar, there remains two possible interpretations of the search direction.
  First the "weak" interpretation where discretization leads to:

  $$\begin{aligned} \int_{\Gamma_{EE'}} (\underline{F}_{EE'} - \widehat{\underline{F}}_{EE'}) \cdot \dot{W}^* \, dS \pm \int_{\Gamma_{EE'}} k_V^{EE'}(\underline{\dot{W}}_{EE'} - \widehat{\underline{\dot{W}}}_{EE'}) \cdot \dot{W}^* \, dS = 0 \\ \mathbf{F}_{EE'} - \widehat{\mathbf{F}}_{EE'} \pm \mathbf{k}_V^{EE'}(\dot{\mathbf{W}}_{EE'} - \widehat{\dot{\mathbf{W}}}_{EE'}) = 0 \end{aligned} \tag{22}$$

  and the discrete search direction $\mathbf{k}_V^{EE'}$ is some sort of weighted L^2-mass matrix of the interface. Second the "strong" (collocation) interpretation where one directly writes:

  $$\mathbf{F}_{EE'} - \widehat{\mathbf{F}}_{EE'} \pm \mathbf{k}_V^{EE'}(\dot{\mathbf{W}}_{EE'} - \widehat{\dot{\mathbf{W}}}_{EE'}) = 0 \tag{23}$$

  which has the evident advantage of making the search direction local.
- Note that a local search direction can also be derived from a "weak" interpretation by lumping $\mathbf{k}_{EE'}$ or simply keeping only the diagonal terms.

3.2 The non-invasive Latin method

The mono-scale Latin method is the direct application of the two principles exposed previously. We detail the method in the case of perfect interfaces. The linear stage is presented first since it is mainly impacted by the non-invasive hypothesis. Several possibilities to tackle the local stage are presented afterwards.

3.2.1 Linear stage

The linear stage consists in solving problems on the sub-domains. Equations (14) and (18) lead to the following equations which are independent per sub-domains, and where $(\widehat{\mathbf{F}}, \widehat{\dot{\mathbf{W}}})$ are known from previous local stage:

$$\text{Solve for } \mathbf{U} \text{ solution to: } \mathbf{K}\mathbf{U} + \mathbf{N}^T\mathbf{k}_V^-\mathbf{N}\dot{\mathbf{U}} = \mathbf{f}_d + \mathbf{N}^T\left(\widehat{\mathbf{F}} + \mathbf{k}_V^-\widehat{\dot{\mathbf{W}}}\right) \tag{24}$$

In order to solve the differential equation on $\mathbf{U}$, an Euler implicit integration time scheme is applied : $\dot{\mathbf{U}}_{t+1} = \dfrac{\mathbf{U}^{t+\Delta t} - \mathbf{U}_t}{\Delta t}$. That leads to the problem written in displacement:

$$\left(\mathbf{K} + \mathbf{N}^T \frac{\mathbf{k}_V^-}{\Delta t}\mathbf{N}\right)\mathbf{U}^{t+\Delta t} = \mathbf{f}_d + \mathbf{N}^T\left(\widehat{\mathbf{F}}^{t+\Delta t} + \mathbf{k}_V^- \widehat{\dot{\mathbf{W}}}^{t+\Delta t} + \frac{\mathbf{k}_V^-}{\Delta t}\mathbf{W}^t\right) \tag{25}$$

$$\text{Then compute } \begin{cases} \dot{\mathbf{W}}^{t+\Delta t} = \mathbf{N}\dot{\mathbf{U}}_{t+1} \\ \mathbf{F}^{t+\Delta t} = \widehat{\mathbf{F}}^{t+\Delta t} + \mathbf{k}_V^-\left(\widehat{\dot{\mathbf{W}}}^{t+\Delta t} - \widehat{\mathbf{W}}^{t+\Delta t}\right) \end{cases} \tag{26}$$

Remark 4 Equation (24) can be interpreted as the discretization of the sub-domains equilibrium under a generalized Robin boundary condition. The term $(\mathbf{N}^T\mathbf{k}_V^-\mathbf{N})$ in (24) corresponds to the interface impedance, it is a non-standard term in commercial software for mechanical problems (there often exist implementations for thermal problems, since it corresponds to convection conditions).

3.2.2 Non-invasive implementation for the linear stage

Here the aim is to propose a new implementation of the generalized Robin condition prone to be implemented in a commercial software. First note that other options were possible:

- first, the interface could be connected to the sub-domain by node-to-node springs elements leading to a local operator.
- second, cohesive elements could be used in their elastic regime. Depending on the chosen quadrature, a local operator can be obtained.

The idea which we investigate in this study, is to think of $\mathbf{k}^- = \mathbf{k}_V^-/\Delta t$ as a generalized Robin condition which can be realized by adding matter on the boundary of the sub-domains. This strategy leads to a non-local search direction.

The matter added at interface $\Gamma_{EE'}$ is written $\theta_{EE'}$, the Hooke tensor associated to its behavior is $\mathbb{K}_{\theta_{EE'}}$. The addition is called a sole. A zero Dirichlet condition is imposed on the part $\partial_u \mathbb{K}_{\theta_{EE'}}$ of the boundary of the sole which is not in contact with the interface.

At the continuous level, the search direction operator written in displacement $\underline{\underline{k}}_{EE'}$ is defined by the following relation:

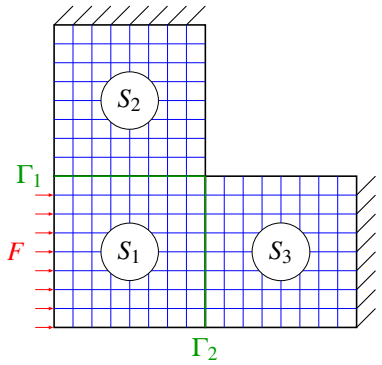
$$\underline{F}_{EE'} = k_{EE'}(\underline{W}_{EE'}) \qquad \Leftrightarrow \qquad \begin{cases} \underline{u} \in H^1(\theta_{EE'}),\ \underline{u} = \underline{W}_{EE'} \text{ on } \Gamma_{EE'},\ \underline{u} = 0 \text{ on } \partial_u\theta_{EE'} \\ \forall \underline{v} \in H^1(\theta_{EE'}),\ \underline{v} = 0 \text{ on } \Gamma_{EE'} \cup \partial_u\theta_{EE'} \\ \displaystyle\int_{\theta_{EE'}} \boldsymbol{\varepsilon}(\underline{v}) : \mathbb{K}_{\theta_{EE'}} : \boldsymbol{\varepsilon}(\underline{u})\ \mathrm{d}\Omega = \int_{\Gamma_{EE'}} \underline{F}_{EE'} \cdot \underline{v}\,\mathrm{d}S \end{cases} \tag{27}$$

At the discrete level, if $\mathbf{K}_{\theta_{EE'}}$ is the stiffness matrix of the sole (with zero Dirichlet boundary conditions taken into account), then $\mathbf{k}^-$ is the Schur complement of $\mathbf{K}_{\theta_{EE'}}$ which condenses the stiffness on the interface degrees of freedom.

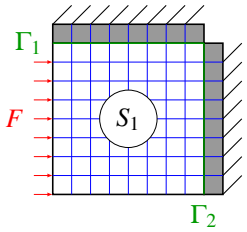
Remark 5 A parallel can be done with the Restricted Additive Schwarz method [5; 4]. Indeed the matter addition at the interface may be seen as an overlap between two sub-domains. However contrary to the Schwarz method, the boundary of the overlap which is not in contact with the interface is blocked, whereas a displacement come from the sub-domains is imposed in the Schwarz method. Moreover the material characteristics of the added matter is a parameter of the method.

From an implementation point of view, solving the problem of the linear stage $\left(\mathbf{K} + \mathbf{N}^T\mathbf{k}^-\mathbf{N}\right)$ corresponds to solving a mechanical problem on the sub-domain extended with the sole, loaded by an inner interface traction field, (Figure 3) which is a classic operation in most software. This permits to compute $\mathbf{W}_{EE'}$ and $\mathbf{U}_E$.

The nodal reaction $\mathbf{F}_{EE'}$ can be obtained by post-processing the sub-domain or by computing a Dirichlet problem on the sole alone.



(a) 3 sub-domains with 2 interfaces



(b) Subdomain with soles. Interface forces are applied on (Γ_1, Γ_2)

Fig. 3 Example of a problem to solve at linear stage

3.2.3 The local stage

The local stage corresponds to solving a system of equation at the interfaces. Equations (2) and (17) lead to searching for $\left(\widehat{\mathbf{W}}, \widehat{\dot{\mathbf{W}}}, \widehat{\mathbf{F}}\right)$, solution to:

$$\begin{cases} \widehat{\mathbf{F}}(\tau) = \mathbf{b}\left(\widehat{\dot{\mathbf{W}}}, t < \tau\right) & \text{Constitutive law} \\ \widehat{\mathbf{F}} - \mathbf{F} - \mathbf{k}_V^+\left(\widehat{\dot{\mathbf{W}}} - \dot{\mathbf{W}}\right) = 0 & \text{Search direction} \end{cases} \tag{28}$$

We detail the case of a perfect interface:

$$\begin{cases} \mathbf{A}\widehat{\mathbf{F}} = 0 & \text{Equilibrium} \\ \mathbf{B}\widehat{\mathbf{W}} = 0 & \text{Continuity of displacement} \\ \widehat{\mathbf{F}} - \mathbf{F} - \mathbf{k}_V^+\left(\widehat{\dot{\mathbf{W}}} - \dot{\mathbf{W}}\right) = 0 & \text{Search direction} \end{cases} \tag{29}$$

The use of the search direction in the relation of equilibrium and the relation of continuity leads to the following expressions of $\left(\widehat{\mathbf{W}}^{t+\Delta t}, \widehat{\mathbf{F}}^{t+\Delta t}\right)$:

$$\begin{cases} \mathbf{A}\mathbf{k}_V^+\widehat{\mathbf{W}}^{t+\Delta t} = \mathbf{A}\mathbf{k}_V^+\dot{\mathbf{W}}^{t+\Delta t} + \mathbf{A}\dfrac{\mathbf{k}_V^+}{\Delta t}\widehat{\mathbf{W}}^t - \mathbf{A}\mathbf{F}^{t+\Delta t} \\ \mathbf{B}\mathbf{k}_V^{+-1}\widehat{\mathbf{F}}^{t+\Delta t} = \mathbf{B}\mathbf{k}_V^{+-1}\mathbf{F}^{t+\Delta t} - \mathbf{B}\mathbf{W}^{t+\Delta t} \end{cases} \tag{30}$$

The equilibrium and continuity relations also impose that $\widehat{\mathbf{F}}$ and $\widehat{\mathbf{W}}$ respectively belong to Range$(\mathbf{B}^T)$ and Range$(\mathbf{A}^T)$. Thus one can deduce that:

$$\boxed{\begin{cases} \widehat{\mathbf{W}}^{t+\Delta t} = \mathbf{A}^T\left(\mathbf{A}\mathbf{k}_V^+\mathbf{A}^T\right)^{-1}\left[\mathbf{A}\mathbf{k}_V^+\dot{\mathbf{W}}^{t+\Delta t} - \mathbf{A}\mathbf{F}^{t+\Delta t} + \mathbf{A}\dfrac{\mathbf{k}_V^+}{\Delta t}\widehat{\mathbf{W}}^t\right] \\ \widehat{\mathbf{F}}^{t+\Delta t} = \mathbf{B}^T\left(\mathbf{B}\mathbf{k}_V^{+-1}\mathbf{B}^T\right)^{-1}\left[\mathbf{B}\mathbf{k}_V^{+-1}\mathbf{F}^{t+\Delta t} - \mathbf{B}\dot{\mathbf{W}}^{t+\Delta t}\right] \end{cases}} \tag{31}$$

Remark 6 If we consider a continuous formulation, the solution of the perfect interface problem for the local stage becomes:

$$\begin{cases} \underline{\widehat{\mathbf{W}}}_{\mathbf{EE'}}{}^{t+\Delta t} = \underline{\widehat{\mathbf{W}}}_{\mathbf{E'E}}{}^{t+\Delta t} = (\mathbf{k_{EE'}} + \mathbf{k_{E'E}})^{-1} \left[-\underline{\mathbf{F}}_{\mathbf{EE'}}{}^{t+\Delta t} - \underline{\mathbf{F}}_{\mathbf{E'E}}{}^{t+\Delta t} + \mathbf{k_{EE'}} \underline{\widehat{\mathbf{W}}}_{\mathbf{EE'}}{}^{t} \dots \right. \\ \qquad \left. + \dots \mathbf{k_{E'E}} \widehat{\mathbf{W}}^{t}_{\mathbf{E'E}} + \mathbf{k_{EE'}}{}^{V} \underline{\dot{\mathbf{W}}}^{t+\Delta t}_{\mathbf{EE'}} \dots \right. \\ \qquad \left. + \dots \mathbf{k_{E'E}}{}^{V} \underline{\dot{\mathbf{W}}}^{t+\Delta t}_{\mathbf{E'E}} \right] \\ \underline{\widehat{\mathbf{F}}}_{\mathbf{EE'}}{}^{t+\Delta t} = -\underline{\widehat{\mathbf{F}}}_{\mathbf{E'E}}{}^{t+\Delta t} = \left(\mathbf{k_{EE'}}{}^{-1} + \mathbf{k_{E'E}}{}^{-1} \right)^{-1} \left[\underline{\dot{\mathbf{W}}}^{t+\Delta t}_{\mathbf{E'E}} - \underline{\dot{\mathbf{W}}}^{t+\Delta t}_{\mathbf{EE'}} \dots \right. \\ \qquad \left. + \dots \mathbf{k_{EE'}}{}^{-1} \underline{\mathbf{F}}_{\mathbf{EE'}}{}^{t+\Delta t} - \mathbf{k_{E'E}}{}^{-1} \underline{\mathbf{F}}_{\mathbf{E'E}}{}^{t+\Delta t} \right] \end{cases} \tag{32}$$

Thus the case of a perfect interface is a resolution of a mechanical problem over two assembled soles with a Neumann condition on the interface between them.

Remark 7 The case of local stage with contact will be developed further. A major difficulty will be constituted by the fact that the proposed "non-invasive" search direction $\mathbf{k}^-$ is non-local. Thus the choice $\mathbf{k}^+ = \mathbf{k}^-$, which is recommended by the theory, is not efficient from a computation point of view. Propositions will be made to obtain a diagonal operator $\mathbf{k}^+$ "close to" $\mathbf{k}^-$.

3.2.4 Control of the convergence

An error indicator η can be defined to control the convergence of the method. This indicator represents the distance between a local solution in $\mathcal{L}$ and a linear solution in $\mathcal{A}$:

$$\eta = \frac{\left(\dot{\mathbf{W}} - \widehat{\dot{\mathbf{W}}}\right)^T \mathbf{k}_V^- \left(\dot{\mathbf{W}} - \widehat{\dot{\mathbf{W}}}\right) + \left(\mathbf{F} - \widehat{\mathbf{F}}\right)^T \mathbf{k}_V^{-}{}^{-1} \left(\mathbf{F} - \widehat{\mathbf{F}}\right)}{\dot{\mathbf{W}} \mathbf{k}_V^- \dot{\mathbf{W}} + \mathbf{F} \mathbf{k}_V^{-}{}^{-1} \mathbf{F} + \widehat{\dot{\mathbf{W}}} \mathbf{k}_V^- \widehat{\dot{\mathbf{W}}} + \widehat{\mathbf{F}} \mathbf{k}_V^{-}{}^{-1} \widehat{\mathbf{F}}} \tag{33}$$

Remark 8

- The indicator is post-processed from the linear stage.
- The indicator depends on the search direction.
- Other indicators based on true errors **AF** and **BW**, which are computed at the begin of the local step can also be used.

4 Treatment of the contact

4.1 Frictionless contact

In order to write the contact problem, we assume sufficient smoothness for the interface so that we can define a local basis at each interface node: n stands for the outer normal direction and t for the tangent direction (the tangent basis is (t_1, t_2) in 3D). Conventionally the same normal vector is used on both neighbors, the orientation of the vector is chosen in agreement with the sign convention of **B**. Thus we assume all vectors and matrix are turned in the local interface frame (n, t).

In order to preserve locality, the search direction must be a diagonal operator (in the normal-tangent frame). Note that obtaining such an operator is not trivial in the case of search directions obtained with matter addition. In our case the diagonal of the stiffness operator of the sole is chosen.

The local stage for contact interfaces regroups the different relations that govern the contact plus the search directions written in term of velocity. As for the linear stage, an implicit scheme is applied to link the speed distribution and the displacement distribution. Thus the problem to solve becomes:

$$\text{Find } \left(\widehat{\mathbf{W}}^{t+\Delta t}, \widehat{\dot{\mathbf{W}}}^{t+\Delta t}, \widehat{\mathbf{F}}^{t+\Delta t}\right) \text{ solution to: } \begin{cases} \mathbf{B}_n \widehat{\mathbf{W}}_n^{t+\Delta t} + \mathbf{j}_n \geqslant 0 \\ \mathbf{A}\widehat{\mathbf{F}}^{t+\Delta t} = 0 \\ \mathbf{B}_n \widehat{\mathbf{F}}_n^{t+\Delta t} \geqslant 0 \\ \mathbf{B}_t \widehat{\mathbf{F}}_t^{t+\Delta t} = 0 \\ \left(\mathbf{B}_n \widehat{\mathbf{F}}_n^{t+\Delta t}\right)^T \left(\mathbf{B}_n \widehat{\mathbf{W}}_n^{t+\Delta t} + \mathbf{j}_n\right) = 0 \\ \widehat{\mathbf{F}}^{t+\Delta t} - \mathbf{F}^{t+\Delta t} - \mathbf{k}_V^+ \left(\widehat{\dot{\mathbf{W}}}^{t+\Delta t} - \dot{\mathbf{W}}^{t+\Delta t}\right) = 0 \\ \widehat{\mathbf{W}}^{t+\Delta t} = \widehat{\mathbf{W}}^t + \Delta t \widehat{\dot{\mathbf{W}}}^{t+\Delta t} \end{cases} \tag{34}$$

where hat-less quantities and quantities at time step t are known from previous time step and linear stage. $\mathbf{j}_n$ is the initial normal gap.

To solve the problem, the time integration scheme is used in the non-interpenetration relation, and then the search direction is applied to make $\widehat{\mathbf{F}}^{t+\Delta t}$ appear, so that we can control the sign of the normal component.

$$\mathbf{B}_n \widehat{\mathbf{W}}_n^{t+\Delta t} + \mathbf{j}_n = \mathbf{B}_n \widehat{\mathbf{W}}_n^t + \mathbf{j}_n + \Delta t \mathbf{B}_n \left(\dot{\mathbf{W}}_n^{t+\Delta t} + {\mathbf{k}_{Vn}^+}^{-1} \left(\widehat{\mathbf{F}}_n^{t+\Delta t} - \mathbf{F}_n^{t+\Delta t}\right)\right) \geqslant 0 \tag{35}$$

which can be reorganized as:

$$\underbrace{\mathbf{B}_n \widehat{\mathbf{W}}_n^t + \mathbf{j}_n + \Delta t \mathbf{B}_n \dot{\mathbf{W}}_n^{t+\Delta t} - \Delta t \mathbf{B}_n {\mathbf{k}_{Vn}^+}^{-1} \mathbf{F}_n^{t+\Delta t}}_{\mathbf{C}_n} + \Delta t \mathbf{B}_n {\mathbf{k}_{Vn}^+}^{-1} \widehat{\mathbf{F}}_n^{t+\Delta t} \geqslant 0 \tag{36}$$

$\mathbf{C}_n$ is a computable quantity which enables us to evaluate nodewise the feasibility of a contactless solution. In the following non-bold characters stand for a given node, we have:

$$\boxed{\begin{cases} \text{Contact} & \Leftrightarrow C_n \leqslant 0 \\ \text{No-contact} & \Leftrightarrow C_n > 0 \end{cases}} \tag{37}$$

No contact case $\boxed{C_n > 0}$

In that case, the solution $\widehat{F}_n^{t+\Delta t} = 0$ and $B_n \widehat{W}^{t+\Delta t} + j_n = C_n > 0$ satisfies all equations.

Contact case $\boxed{C_n \leqslant 0}$

In that case, the solution is $\Delta t B_n {k_{Vn}^+}^{-1} \widehat{F}_n^{t+\Delta t} = -C_n \geqslant 0$ and $B_n \widehat{W}_n^{t+\Delta t} + j_n = 0$. To compute $\widehat{F}_n^{t+\Delta t}$ one simply has to remember that it can be expressed as $\widehat{F}_n^{t+\Delta t} = B_n^T F_n^*$ for some F_n^* (indeed $\widehat{F}_n^{t+\Delta t}$ verifies $A_n \widehat{F}_n^{t+\Delta t} = 0$), and thus:

$$\boxed{F_n^{t+\Delta t} = -B_n^T \left(B_n {k_{Vn}^+}^{-1} B_n^T\right)^{-1} \frac{C_n}{\Delta t}} \tag{38}$$

$\widehat{W}_n^{t+\Delta t}$ then can be deduced using the search direction and the time integration.

For the tangential component of the interface distributions, we use the fact that $\widehat{F}_t^{t+\Delta t} = 0$. Then with the search direction and the temporal integration scheme, we obtain :

$$\boxed{\begin{aligned} \widehat{\dot{W}}_t^{t+\Delta t} &= \dot{W}_t^{t+\Delta t} - {k_{Vt}^+}^{-1} F_t^{t+\Delta t} \\ \widehat{W}_t^{t+\Delta t} &= \widehat{W}_t^{t+\Delta t} + \Delta t \left(\dot{W}_t^{t+\Delta t} - {k_{Vt}^+}^{-1} F_t^{t+\Delta t}\right) \end{aligned}} \tag{39}$$

As a conclusion it is possible to define a nodewise indicator which gives the contact status. This indicator can be expressed only with the distribution $\left(\widehat{W}_t^t, \dot{W}^{t+\Delta t}, F^{t+\Delta t}\right)$ known from the previous linear stage and previous time step of local stage. Then the computation of the solutions $\left(\widehat{W}^{t+\Delta t}, \widehat{\dot{W}}^{t+\Delta t}, \widehat{F}^{t+\Delta t}\right)$ is explicit.

4.2 Coulomb friction

The frictional aspect is taken into account by adding new conditions in the case of contact. First the normal components can be found as explained in previous subsection. Then in the case of contact, another indicator is computed to evaluate the possibility of sliding.

Let μ be Coulomb's friction coefficient. The friction conditions can be written nodewise as:

$$\begin{cases} \text{If } \|\widehat{F}_t^{t+\Delta t}\| \leqslant \mu |\widehat{F}_n^{t+\Delta t}| \text{ then } B_t \widehat{\dot{W}}_t^{t+\Delta t} = 0 \\ \text{If } \|\widehat{F}_t^{t+\Delta t}\| = \mu |\widehat{F}_n^{t+\Delta t}| \text{ then } \exists \lambda > 0,\ B_t \widehat{\dot{W}}_t^{t+\Delta t} = \lambda \left(B_t B_t^T\right)^{-1} B_t \widehat{F}_t^{t+\Delta t} \end{cases} \tag{40}$$

where $\widehat{F}_t^{t+\Delta t}$ is still submitted to the balance condition $A_t \widehat{F}_t^{t+\Delta t} = 0$.

With the same idea as for frictionless contact, it is possible to define an indicator that makes precise the status of the node. The tangential component of the force distribution is computed and compared to the norm of the normal one. Beforehand, as the search direction is semi positive definite, a decomposition of the tangential velocity distribution is done:

$$\exists\, (\alpha, \beta),\ \widehat{\dot{W}}_t^{t+\Delta t} = A_t^T \alpha + {k_{Vt}^+}^{-1} B_t^T \beta \tag{41}$$

$A_t^T \alpha$ corresponds to the sticking part whereas ${k_{Vt}^+}^{-1} B_t^T \beta$ corresponds to the sliding part.

With such an expression for the velocity distribution, the search direction is used to computed the force distribution:

$$\widehat{F}_t^{t+\Delta t} = F_t^{t+\Delta t} + k_{Vt}^+ \left(\widehat{\dot{W}}_t^{t+\Delta t} - \dot{W}_t^{t+\Delta t}\right) \tag{42}$$

Applying the equilibrium of the force distribution, α can be determined as:

$$\alpha = \left(A_t k_{Vt}^+ A_t^T\right)^{-1} A_t \left[k_{Vt}^+ \dot{W}^{t+\Delta t} - F^{t+\Delta t}\right] \tag{43}$$

Thus the expression of the force distribution becomes:

$$\widehat{F}_t^{t+\Delta t} = \underbrace{\left[\mathbb{I} - k_{Vt}^+ A_t^T \left(A_t k_{Vt}^+ A_t^T\right)^{-1} A_t\right] F^{t+\Delta t} - k_{Vt}^+ \left[\mathbb{I} - A_t^T \left(A_t k_{Vt}^+ A_t^T\right)^{-1} A_t k_{Vt}^+\right] \dot{W}_t^{t+\Delta t}}_{G_t} + B_t^T \beta \tag{44}$$

G_t represents the sticking contribution to the force distribution. It only depends on contributions from the linear stage and so can be computed directly at the local stage. It is used as a nodewise indicator for the possibility of sliding.

$$\boxed{\begin{array}{ll} \text{Sticking} & \Leftrightarrow \|G_t\| \leqslant \mu |F_n^{t+\Delta t}| \\ \text{Sliding} & \Leftrightarrow \|G_t\| > \mu |F_n^{t+\Delta t}| \end{array}} \tag{45}$$

Sticking case If $\boxed{\|G_t\| \leqslant \mu |\widehat{F}_n^{t+\Delta t}|}$

We can directly set $\beta = 0$, $\widehat{F}_t^{t+\Delta t} = G_t$. The velocity distribution is computed with the expression of α.

$$\begin{cases} \widehat{\dot{W}}_t^{t+\Delta t} = A_t^T \left(A_t k_{Vt}^+ A_t^T\right)^{-1} A_t \left(k_t^+ \dot{W}_t^{t+\Delta t} - F_t^{t+\Delta t}\right) \\ \widehat{W}_t^{t+\Delta t} = \widehat{W}_t^t + \Delta t \widehat{\dot{W}}_t^{t+\Delta t} \end{cases} \tag{46}$$

Sliding case: If $\boxed{\|G_t\| > \mu|\widehat{F}_n^{t+\Delta t}|}$

β and $\lambda > 0$ must be found such that $\|\widehat{F}_t^{t+\Delta t}\| = \mu|\widehat{F}_n^{t+\Delta t}|$ and $B_t\widehat{\dot{W}}_t^{t+\Delta t} = \lambda(B_tB_t^T)^{-1}B_t\widehat{F}_t^{t+\Delta t}$.

With the expression of the tangential component of the velocity, the traction can be expressed as a function of β and λ:

$$(B_tB_t^T)^{-1}B_t\widehat{F}_t^{t+\Delta t} = \frac{1}{\lambda}B_t{k_{Vt}^+}^{-1}B_t^T\beta \tag{47}$$

Moreover $\widehat{F}_t^{t+\Delta t} = G_t + B_t^T\beta$. Thus the relation between β and λ is:

$$(B_tB_t^T)^{-1}B_tG_t + \beta = B_t{k_{Vt}^+}^{-1}B_t^T\frac{\beta}{\lambda} \tag{48}$$

and

$$\beta = \lambda\left(-\lambda\mathbb{I} + B_t{k_{Vt}^+}^{-1}B_t^T\right)^{-1}(B_tB_t^T)^{-1}B_tG_t \tag{49}$$

The tangential force distribution can be expressed as a function of λ:

$$\widehat{F}_t^{t+\Delta t} = \left(\mathbb{I} - B_t^T\left(\mathbb{I} - \frac{B_t{k_{Vt}^+}^{-1}B_t^T}{\lambda}\right)^{-1}(B_tB_t^T)^{-1}B_t\right)G_t \tag{50}$$

And $\lambda > 0$ is determined by the relation $\|\widehat{F}_t^{t+\Delta t}\| = \mu|\widehat{F}_n^{t+\Delta t}|$ given from the sliding assumption. It is more convenient to consider the intermediate quantity (non-redundant between the two sides of the interface): $\|(B_tB_t^T)^{-1}B_t\widehat{F}_t^{t+\Delta t}\| = \mu|(B_nB_n^T)^{-1}B_n\widehat{F}_n^{t+\Delta t}|$. Then we must ensure:

$$\left\|\left(\mathbb{I} - \left(\mathbb{I} - \frac{B_t{k_{Vt}^+}^{-1}B_t^T}{\lambda}\right)^{-1}\right)\underbrace{(B_tB_t^T)^{-1}B_tG_t}_{G_t'}\right\| = \underbrace{\mu|(B_nB_n^T)^{-1}B_n\widehat{F}_n^{t+\Delta t}|}_{D} \tag{51}$$

Note that the square matrices are all diagonal.

Remark 9

- $\widehat{F}_t^{t+\Delta t}$ can be expressed as $\widehat{F}_t^{t+\Delta t} = \underline{\underline{\gamma}}G_t'$ as $\underline{\underline{\gamma}} = \left(\mathbb{I} - \left(\mathbb{I} - \frac{B_t{k_{Vt}^+}^{-1}B_t^T}{\lambda}\right)^{-1}\right)$
- In a 2D case, $\underline{\underline{\gamma}} \in \mathbb{R}_+^*$ and $G_t' \in \mathbb{R}$ so the unknown γ is easily computed with $\gamma = \frac{\mu|\widehat{F}_n^{t+\Delta t}|}{\|G_t'\|}$ and therefore:

$$\boxed{\widehat{F}_t^{t+\Delta t} = \mu|\widehat{F}_n^{t+\Delta t}|\frac{G_t'}{\|G_t'\|}} \tag{52}$$

- In a general 3D case, the equation 51 is a polynomial equation of the forth degree. However, in the case of the two tangential search directions are equal, $\underline{\underline{\gamma}} = \gamma\mathbb{I}$ with $\gamma > 0$. Under this assumption, a similar result than previously is obtained: $\gamma = \frac{\mu|\widehat{F}_n^{t+\Delta t}|}{\|G_t'\|}$ and therefore:

$$\boxed{\widehat{F}_t^{t+\Delta t} = \mu|\widehat{F}_n^{t+\Delta t}|\frac{G_t'}{\|G_t'\|}} \tag{53}$$

- The velocity distribution $\widehat{\dot{W}}^{t+\Delta t}$ is computed with the relation of the search direction and the expression of the force distribution $\widehat{F}^{t+\Delta t}$. Then the displacement is computed with the temporal integration scheme.

The Latin method for frictional contact is summarized in algorithm 1.

Algorithm 1: Summary of the Latin method

Input: Initialisation
while *Error criterion* $<$ *objective* **do**
  Local stage:
  foreach *time step* **do**
    foreach *interface* **do**
      if *interface is perfect* **then**
        Non-local computation on a double sole to obtain displacement : $\mathbf{W}^{t+\Delta t}$
        Explicit time scheme to compute velocity :$\widehat{\dot{\mathbf{W}}}^{t+\Delta t}$
        Search direction to compute force : $\widehat{\mathbf{F}}^{t+\Delta t}$
      else
        Contact interface:
        for *each node on the interface* **do**
          Compute the contact indicator C_n
          if $C_n \geqslant 0$ **then**
            No contact computation
          else
            Contact computation
            Compute the sticking indicator G_t
            if $\|G_t\| \leqslant \mu|\widehat{F}_n^{t+\Delta t}|$ **then**
              Sticking computation
            else
              Sliding computation
            end
          end
        end
      end
    end
  end
  Linear stage:
  foreach *time step* **do**
    foreach *sub-domain* **do**
      Computation of displacement $\mathbf{W}^{t+\Delta t}$ by solving a problem on sub-domain plus soles
      Explicit time scheme to compute velocity :$\dot{\mathbf{W}}^{t+\Delta t}$
      Search direction to compute force : $\mathbf{F}^{t+\Delta t}$
      Relaxation step: $s_{n+1} \leftarrow s_n + \alpha(s_{n+1} - s_n)$;
    end
  end
end

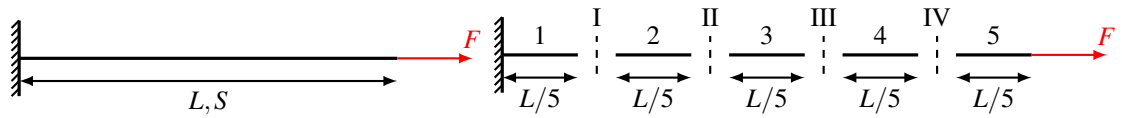
5 Numerical examples

First a validation of implementation of the non-invasive aspect is presented in 5.1 compared to a "standard" Latin with an invasive modification of the stiffness operator. This validation is performed on a simple traction beam study. Moreover the influence of the search direction on the convergence is shown. An academical test case for contact with friction is presented in 5.2. The objective is to validate the frictional contact by comparing results with the ones in [6]. A 3D study of a bolted assembly involving frictional contact with initial gap and pre-load interfaces is developed in 5.3. The computations are made with Code_Aster driven by a python instance.

5.1 Comparison "standard Latin" / non-invasive Latin

The comparison between a "standard" invasive Latin method and the non-invasive one is performed on a simple traction beam problem fixed on one side and under a load F on the other side (Figure 4). The Young modulus is E, the length L and a section S. A sub-structuring in 5 sub-domains is chosen. In this 1D case, the search direction is reduced to only one parameter per interface.

For the standard computation $\mathbf{k}^- = \mathbf{k}^+ = \frac{ES}{L}$ [25]. For the non-invasive version, the parameter is reduced to the Young modulus of the soles and therefore it is easy to obtain equivalent search directions for the two cases by choosing E_s the Young modulus of the soles as $E_s = E\frac{L_s}{L}$ with L_s



(a) Reference problem (b) Sub-structured problem

Fig. 4 Traction beam

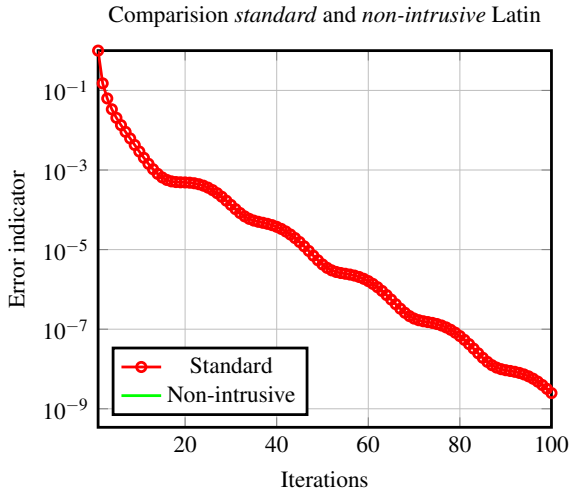


Fig. 5 Comparison between the standard and the non-invasive Latin

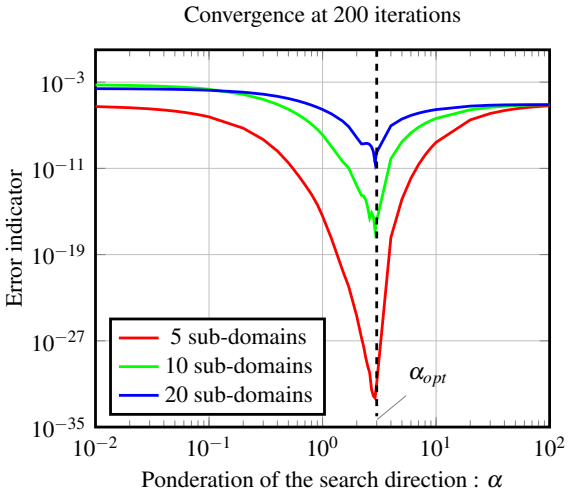


Fig. 6 Parametric study of the search direction

the length of the soles. As the soles are composed with a layer of one element, the size L_s is equal to the discretization h. A parametric computation is led to study the impact of the search direction on the convergence.

The numerical value of the parameters are given on the Table 1.

Table 1 Parameters

Parameters	Value
E	200 GPa
L	100 mm
S	10 mm
h=L_s	1 mm

As seen on the Figure 5 the standard and the non-invasive Latin method give the same results. This validates the non-invasive implementation.

On figure 6 the parametric study is presented to illustrate the dependency of convergence on the search direction. We use the multiplicative parameter α as $\mathbf{k}^- = \mathbf{k}^+ = \alpha\frac{ES}{L}$ for several sub-structuring. The dimensions of the global structure and the size of the discretization are the same for all the sub-structuring case. The results show that an optimal search direction exists and that it does not depend on the choice of the sub-structuring. The optimal search direction depends on the global structure. These two results are classical for the Latin computation.

Actually as the optimal search direction is not evident to compute for any structure, we keep a "non-optimal" standard one but sufficient to reach good level of convergence (indicator $< 10^{-5}$). The rule to choose $\mathbf{k}^+ = \mathbf{k}^- = \frac{ES}{L}$ is chosen. For 2D or 3D structure that is translated by adjusting the Young modulus of the soles to ensure that soles and sub-domains have approximatively the same rigidity.

5.2 An academic 2D friction case

the developed contact management is exemplified on an academic use-case [6]. The quasi-static problem is presented on the Figure 7. Two interfaces with frictional contact link respectively the sub-domains (1,2) and the sub-domains (2,3) with the parameters μ_1 and μ_2 as Coulomb coefficients. An additional interface enables to manage a boundary condition of unilateral contact with an initial gap j. The structure is under loads F_1 and F_2 defined by two time step t_0 and t_1. Firstly the structure is pre-load with the load F_1 before loading with F_2. The materials are still linear elastic defined by the Young modulus E and the Poisson ratio ν. Each sub-structure is discretized by 25×25 elements. The values of the different parameters are in Table 2.

Table 2 Parameters

Parameters	Value/Range
E	210 GPa
h	50 mm
j	0.04 mm
F_1^{max}	50 MPa
F_2^{max}	30 MPa
μ_1	$\in [0, 0.6]$
μ_2	$\in [0, 0.6]$

Depending of the chosen parameters for the Coulomb coefficient, the sub-domain 2 touches or not the boundary condition with the initial gap. So in order to illustrate this behavior, the force reaction on the boundary condition is computed at convergence of the algorithm and plotted depending on the two parameters μ_1 and μ_2 (Figure 8).

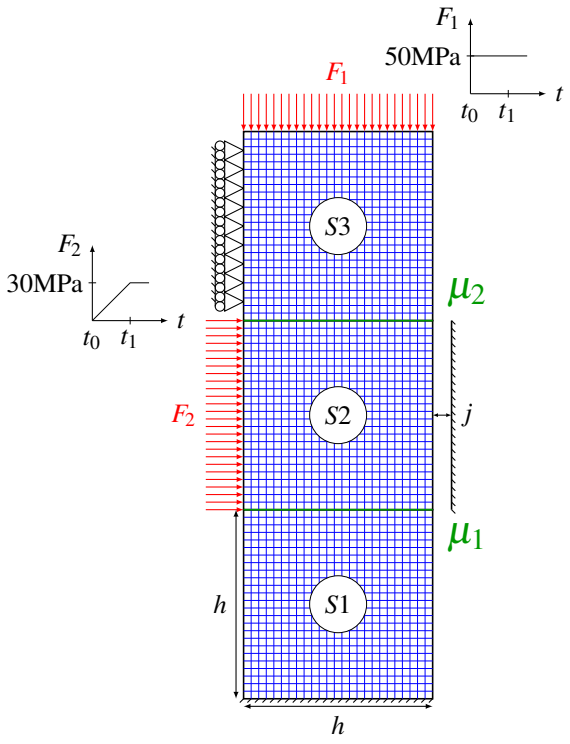


Fig. 7 Test case for frictional contact

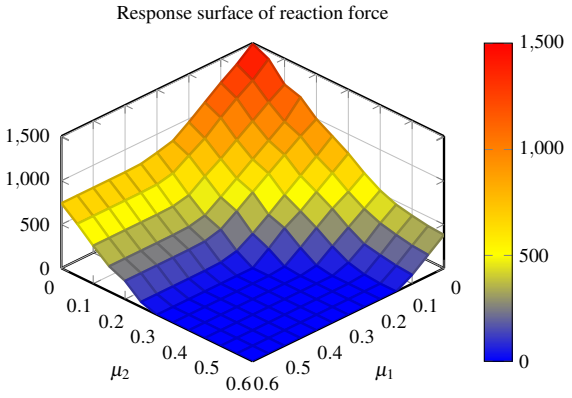
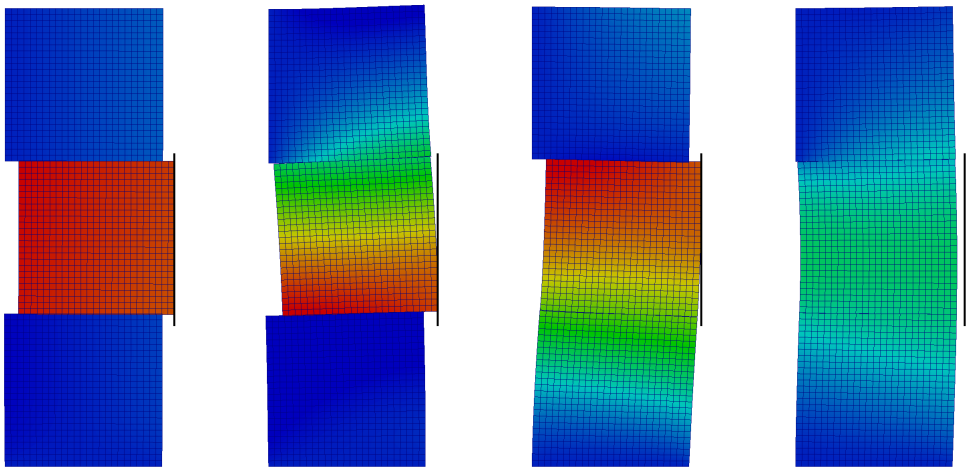


Fig. 8 Response surface of the reaction force wrt (μ_1, μ_2)

Our results correspond to those of Champaney and Boucard in [6]. We obtain the different configurations in which the contact on the interface with gap is reached or not. On Figures 9a,9b,9c contact is reached, Coulomb coefficients are too weak to retain the sub-domain 2. On the other hand, on Figure 9d, no contact is obtained on the interface with gap.



(a) $\mu_1 = 0$, $\mu_2 = 0$ (b) $\mu_1 = 0$, $\mu_2 = 0.6$ (c) $\mu_1 = 0.6$, $\mu_2 = 0$ (d) $\mu_1 = 0.6$, $\mu_2 = 0.6$

Fig. 9 Deformed shape after the second time step for different cases of friction coefficients - Color corresponds to horizontal displacement.

This 2D computation of frictional contact validated our non-invasive implementation. The contact formulation imposes to modify the non-local search direction into a local one. The convergence of the algorithm by using the diagonal of the stiffness of the soles as a local search direction proves that the condition of $\mathbf{k}_V^+ = \mathbf{k}_V^-$ is not really necessary.

5.3 A 3D contact problem

A 3D contact problem of a bolted assembly is presented here (Figure 10). The purpose of this example is to show that the non-invasive permits to tackle complex 3D case with frictional contact. The assembly is composed with two plates and three bolts, symmetry is exploited to limit the size of the problem. Each plate is assigned to three sub-domains. The bolts are also sub-structured in three sub-domains. One is the nut whereas the two others compose the screw. The interface between the two sub-domains of the screw permits to impose a pre-load.

The Young modulus is $E = 200$ GPa, the Poisson's ratio is $\nu = 0.3$. The frictional interface has a Coulomb coefficient of $\mu = 0.1$ and the gap between the screw and the plate is 0.001 mm. These parameters are recalled in Table 3. The problem is computed in four time steps: the first corresponds to the pre-load of the structure, the other ones represent a linear traction force F of the upper plate (Figure 11). The lower plate is fixed on the other side of the assembly.

Remark 10 The pre-load is managed with a contact interface. An initial negative gap imposed at the time step t_1 : $\mathbf{B}_n\mathbf{W}_{|t=t_1} = -\mathbf{j}_n$ with $\mathbf{j}_n > 0$ which results in a tension stress in the screw.

On Figure 13a we represent the elements in compression (σ_{zz} is negative) when the pre-load is imposed. On this example we find the classical Rötscher's pressure cone under the head of the screw [31] and the tension in the body of the screw on Figure 13b. The shearing stress σ_{xz} at the last time step is shown on Figure 14a. Stresses concentrate under heads of screws, due to relative movements of plates. The different contact zones are represented in black on Figure 15 and a detachment of the two plates is visible between the screws. The error indicator is presented on Figure 12.

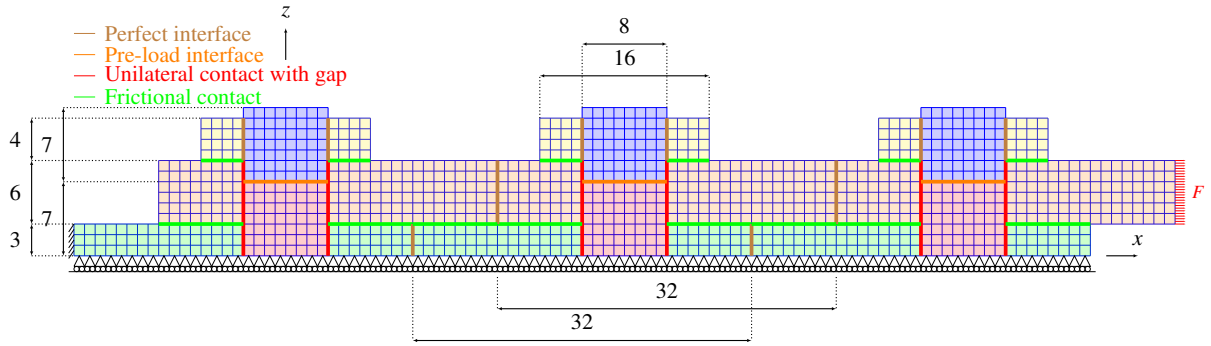


Fig. 10 Description of the interfaces and boundary conditions

Table 3 Parameters

Parameters	Value/Range
E	200 GPa
ν	0.3
pre-load j	0.001 mm
Time step	1 s
Number timestep	4
F_max	30 MPa
μ	0.1
gap j_n	0.001
Size mesh h	0.5 mm

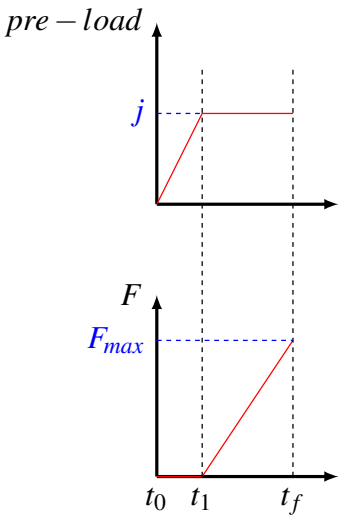


Fig. 11 History of the pre-loads in screws and load F

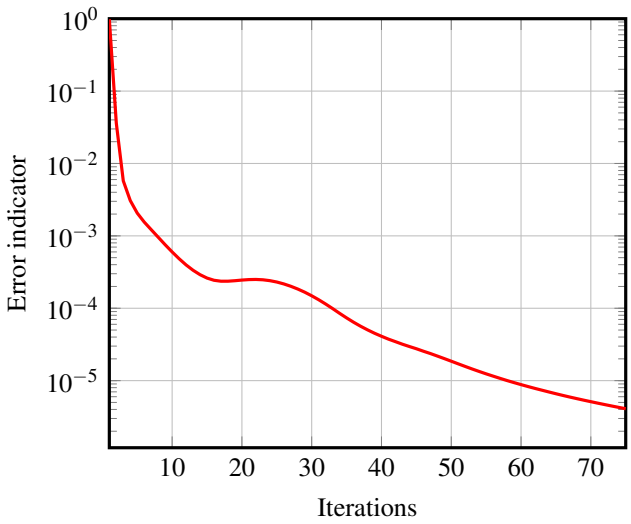
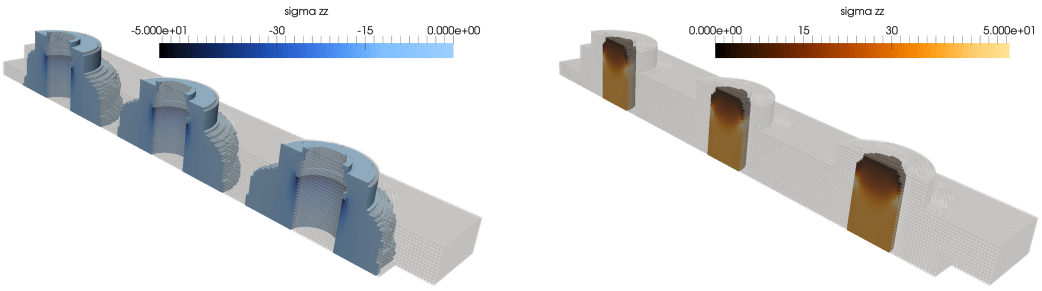


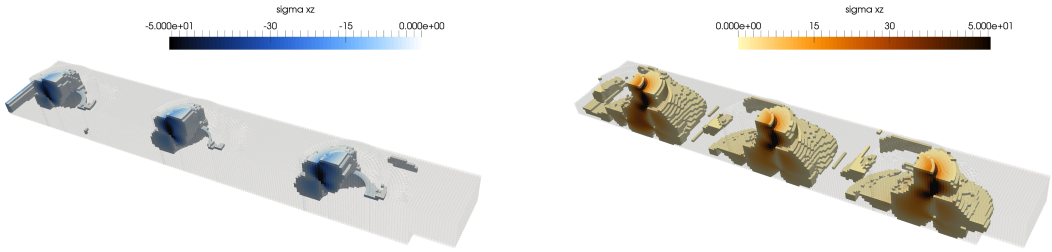
Fig. 12 Error indicator

This last use-case was derived from actual engineering activities experienced at Airbus. It highlights how the non-invasive methods permits to overcome critical issues raised by such configurations (multi bolts joints with significant structural effects of friction and pre-loads). These issues still prevent today from proper computations with commercial software, and demand expensive and time consuming experimental protocols when certifying. A main appeal of non-invasive methods for industry is to allow accurate computational investigations (rather than tests), while skipping any refurbishment of existing numerical frameworks: as demonstrated, non-invasive capabilities can be embedded in day-to-day commercial FE software.



(a) Rotscher's cone with the pre-load: $\sigma_{zz} \leqslant 0$ (b) Tension in the screw with the pre-load: $\sigma_{zz} \geqslant 0$

Fig. 13 Stress in MPa at the time step t_1



(a) Negative shearing stress σ_{xz} (b) Negative shearing stress σ_{xz}

Fig. 14 Stress in MPa at the last time step t_f

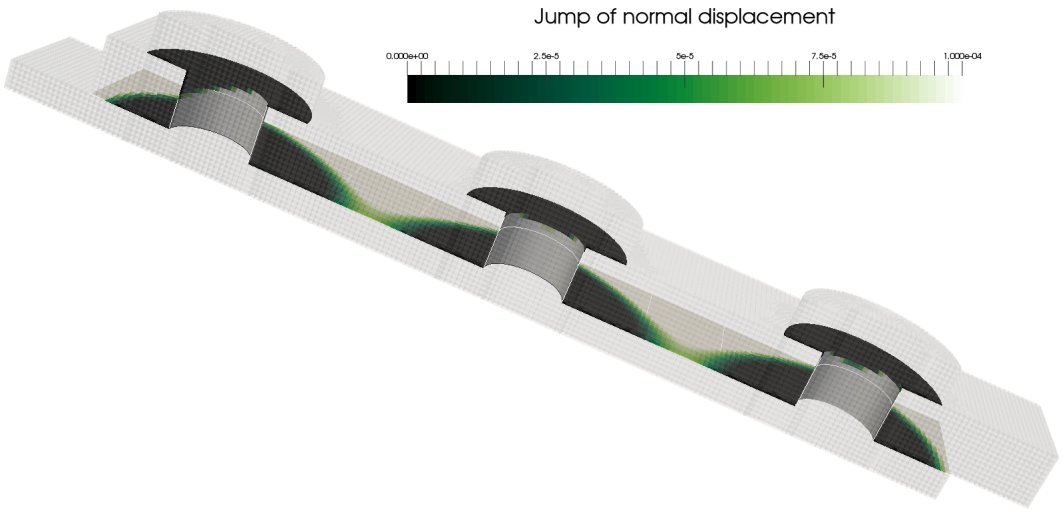


Fig. 15 Contact zones in black - jump of displacement in mm

6 Conclusion

In this paper we have presented a non-invasive derivation of a mixed domain decomposition capable of dealing with frictional contact interfaces. In particular the linear stage of the Latin method has been modified to handle Robin conditions over the boundaries of the sub-domains. A non-local stiffness

operator has been proposed to compute the search direction on the interfaces. It transforms the standard formulation into a simple assembly of stiffness over the sub-domains that any commercial software is able to compute.

The implementation of this derivation has been validated on a simple test case. Classical results of the Latin method about the influence of the search direction can be applied to our new non-invasive method: the optimal search direction does not depend on the sub-structuring but on the global structure.

The contact formulation imposes a scalar node-wise search direction at the local stage that implies different up and down search direction ($\mathbf{k}_V^- \neq \mathbf{k}_V^+$). In order to avoid convergence problems we need to have the closest local search direction from the non-local one. In that sense we choose the scalar search direction as the diagonal of the stiffness operator of the sole. According to that operation we transform the non-local behavior of the interfaces into a decoupled behavior suited to contact formulation. Numerical examples on a 2D academic case confirm the convergence of the algorithm and a first 3D contact problem of a bolted assembly has been presented

The future works will consist in implementing the extensibility of the method. This multi-scale approach, in the mean of [27], modifies the linear stage by introducing a macro resolution that ensures a global equilibrium of force distributions. This approach will be seen as a particular choice of the search direction that makes appear a specific macro projector. The multi-scale extensibility will be coupled with a parallel implementation of the algorithm in order to investigate its scalability.

Acknowledgements We thank Airbus Group Innovations and EDF for their financial and technical support.

References

1. Alart P, Curnier A (1991) A mixed formulation for frictional contact problems prone to Newton like solution methods. Computer Methods in Applied Mechanics and Engineering 92(3):353–375
2. Belgacem F, Hild P, Laborde P (1998) Recent Advances in Contact Mechanics The mortar finite element method for contact problems. Mathematical and Computer Modelling 28(4):263–271
3. Ben Belgacem F, Maday Y (1997) The mortar element method for three dimensional finite elements. ESAIM: Mathematical Modelling and Numerical Analysis 31(2):289–302
4. Cai XC, Sarkis M (1999) A restricted additive schwarz preconditioner for general sparse linear systems. Siam journal on scientific computing 21(2):792–797
5. Cai XC, Farhat C, Sarkis M (1998) A minimum overlap restricted additive schwarz preconditioner and applications to 3d flow simulations. Contemporary Mathematics 218:479–485
6. Champaney L, Boucard P (2003) Multiresolution strategy for the parametric study of assemblies including contact with friction. In: 7th International Conference on Computational Plasticity (COMPLAS), Barcelone, Espagne (cf. p. 41)
7. Desmeure G, Rey C, Gosselet P, Cresta P (2012) On the representation of interface traction field in a mixed domain decomposition method for structural assemblies. In: WCCM - 10th World congress on computational mechanics, Sao Paolo, Brazil
8. Dohrmann C (2003) A preconditioner for substructuring based on constrained energy minimization. SIAM Journal on Scientific Computing 25(1):246–258
9. Dostál Z, Horák D (2004) Scalable FETI with optimal dual penalty for a variational inequality. Numerical linear algebra with applications 11(56):455–472
10. Dostál Z, Horák D, Kučera R, Vondrák V, Haslinger J, Dobiáš J, Pták S (2005) FETI based algorithms for contact problems: scalability, large displacements and 3D Coulomb friction. Computer Methods in Applied Mechanics and Engineering 194(2-5):395–409
11. Dostál Z, Kozubek T, Vondr V, Brzobohaty T, Markopoulos A (2010) Scalable TFETI algorithm for the solution of multibody contact problems of elasticity. International Journal for Numerical Methods in Engineering 82(11):1384–1405, DOI 10.1002/nme.2807
12. Dostál Z, Kozubek T, Brzobohaty T, Markopoulos A, Vlach O (2012) Scalable TFETI with optional preconditioning by conjugate projector for transient frictionless contact problems of elasticity. Computer Methods in Applied Mechanics and Engineering 247-248:37–50
13. Farhat C, Mandel J (1998) The two-level FETI method for static and dynamic plate problems Part I: An optimal iterative solver for biharmonic systems. Computer Methods in Applied Mechanics and Engineering 155(1-2):129–151
14. Farhat C, Roux FX (1991) A method of finite element tearing and interconnecting and its parallel solution algorithm. International Journal for Numerical Methods in Engineering 32(6):1205–1227
15. Farhat C, Mandel J, Roux F (1994) Optimal convergence properties of the FETI domain decomposition method. Computer Methods in Applied Mechanics and Engineering 115(3-4):365–385

16. Farhat C, Lesoinne M, Patrick L, Pierson K, Rixen D (2001) FETI-DP: a dual–primal unified FETI method—part I: A faster alternative to the two-level FETI method. International Journal for Numerical Methods in Engineering 50(7):1523–1544, DOI 10.1002/nme.76
17. Gander MJ (2006) Optimized Schwarz Methods. SIAM Review 44(2):699–731
18. Glowinski R (2015) Variational Methods for the Numerical Solution of Nonlinear Elliptic Problems. Society for Industrial and Applied Mathematics, Philadelphia, PA, DOI 10.1137/1.9781611973785
19. Glowinski R, Le Tallec P (1989) Augmented Lagrangian and operator-splitting methods in nonlinear mechanics. SIAM studies in applied mathematics, Society for Industrial and Applied Mathematics, Philadelphia, DOI 10.1137/1.9781611970838
20. Gosselet P, Rey C (2007) Non-overlapping domain decomposition methods in structural mechanics. Archives of computational methods in engineering 13(4):515–572
21. Gosselet P, Rixen D, Roux FX, Spillane N (2015) Simultaneous-FETI and Block-FETI: robust domain decomposition with multiple search directions. International Journal for Numerical Methods in Engineering 104(10):905–927
22. Hoang TP, Japhet C, Kern M, Roberts J (2014) Ventcell conditions with mixed formulations for flow in porous media. In: Dickopf T, Gander M, Halpern L, Krause R, Pavarino L (eds) Domain Decomposition Methods in Science and Engineering XXII, Lugano (Switzerland), pp 531–540
23. Kerfriden P, Allix O, Gosselet P (2009) A three-scale domain decomposition method for the 3D analysis of debonding in laminates. Computational Mechanics 44(3):343–362
24. Koko J (2011) Uzawa block relaxation method for the unilateral contact problem. Journal of Computational and Applied Mathematics 235(8):2343–2356
25. Ladevèze P (1999) Nonlinear computational structural mechanics: new approaches and non-incremental methods of calculation. Mechanical Engineering Series, Springer-Verlag, New-York, DOI 10.1007/978-1-4612-1432-8
26. Ladevèze P, Dureisseix D (1999) Une nouvelle stratégie de calcul micro/macro en mécanique des structures. Comptes Rendus de l'Académie des Sciences - Series IIB - Mechanics-Physics-Astronomy 327(12):1237–1244
27. Ladevèze P, Nouy A (2003) On a multiscale computational strategy with time and space homogenization for structural mechanics. Computer Methods in Applied Mechanics and Engineering 192(28-30):3061–3087
28. Ladevèze P, Loiseau O, Dureisseix D (2001) A micro–macro and parallel computational strategy for highly heterogeneous structures. International Journal for Numerical Methods in Engineering 52(12):121–138
29. Mandel J, Brezina M (1993) Balancing domain decomposition: Theory and performance in two and three dimensions. Tech. rep., University of Colorado at Denver, Denver, CO, USA
30. Oumaziz P, Gosselet P, Boucard PA (2016) A non-intrusive implementation of a mixed domain decomposition method for frictional contact problems. submitted to Computational Mechanics
31. Rotscher F (1927) Die Maschinenelemente. Springer-Verlag, Berlin, Germany
32. Saavedra K, Allix O, Gosselet P (2012) On a multiscale strategy and its optimization for the simulation of combined delamination and buckling. International Journal for Numerical Methods in Engineering 91(7):772–798
33. Saavedra Redlich K (2012) Stratégie multiéchelle pour l'analyse du couplage flambage-délaminage de composites stratifiés. PhD thesis, École normale supérieure de Cachan
34. Simo JC, Laursen T (1992) An augmented Lagrangian treatment of contact problems involving friction. Computers & Structures 42(1):97–116
35. Simo JC, Wriggers P, Taylor RL (1985) A perturbed Lagrangian formulation for the finite element solution of contact problems. Computer Methods in Applied Mechanics and Engineering 50(2):163–180
36. Spillane N, Rixen DJ (2013) Automatic spectral coarse spaces for robust FETI and BDD algorithms. Internat J Num Meth Engin 95(11):953–990
37. Wriggers P (1996) Finite element methods for contact problems with friction. Tribology International 29(8):651–658
38. Yastrebov VA (2013) Numerical Methods in Contact Mechanics. Numerical methods in engineering series, ISTE/Wiley, London (UK) / Hoboken NJ (US)

A The multi-scale strategy in the Latin method

During the linear stage, displacement and forces distributions do not verify the constitutive law of the interface. In the case of perfect interfaces we have : $\mathbf{BW} \neq 0$ and $\mathbf{AF} \neq 0$. Thus the idea is to modify the linear stage in order to enforce a "global" verification of the constitutive law.

First we explain, with a new point of view, the classical approach which is applicable for any interface behavior and which results in a weak balance over the whole structure. Second we show how enforcing a global weak continuity is also possible in the case of perfect interfaces.

For simplicity reasons, the method is presented in the case of a static problem.

A.1 Multi-scale strategy with global equilibrium

In order to weaken the balance of the interfaces $\mathbf{AF} = 0$, we introduce a displacement macro space, characterized by the basis $\mathbb{W}$ so that the weaken balance can be written as:

$$\mathbb{W}^T \mathbf{AF} = 0 \tag{54}$$

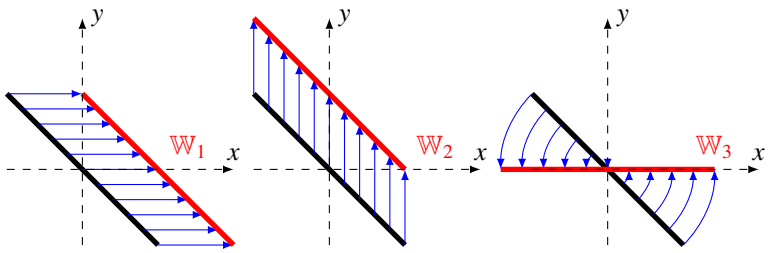
We first briefly discuss the classical choices for $\mathbb{W}$, then we present how the weak balance constraint is handled in the method.

A.1.1 Definition of the displacement macro space

Saint-Venant's principle asserts that $\mathbb{W}$ should contain the trace of the subdomains rigid-body motions in order that the long range interactions be transmitted over the whole structure.

In order to deal with complex situations (heterogeneities, jagged interfaces), a more complete basis should be accessible via GENEO-inspired computations [36].

The classical macro-space for the Latin method is driven by simplicity of implementation and homogenization considerations: it is defined interface by interface and it contains not only the trace of rigid body motions but also first degree interface extension and shear modes.



(a) Translation along x (b) Translation along y (c) Rotation around z

Fig. 16 Examples of macro displacement modes

A.1.2 Definition of the linear stage

The macro-equilibrium imposes to reconsider the search direction which can not be defined independently per interface. The idea is to try to satisfy at best the search direction under the macroscopic constraint (54). This leads to the following modified search direction:

$$\begin{cases} \left(\mathbf{F} - \widehat{\mathbf{F}}\right) + \mathbf{k}^- \left(\mathbf{W} - \widehat{\mathbf{W}}\right) + \mathbf{k}^- \mathbf{A}^T \mathbb{W} \alpha = 0 \\ \mathbb{W}^T \mathbf{AF} = 0 \end{cases} \tag{55}$$

where α is a Lagrange multiplier to be determined

Remarks: Classically [26; 28] the multiplier is written $\tilde{W}$, it corresponds in our case to $\mathbf{A}^T \mathbb{W} \alpha$.

With the quasi-static assumption the problem of the linear stage becomes:

Find $(\mathbf{W}, \mathbf{F})$ solution to :

$$\begin{cases} \mathbf{KU} = \mathbf{f}_d + \mathbf{N}^T \mathbf{F} \\ \mathbf{W} = \mathbf{NU} \\ \mathbb{W}^T \mathbf{AF} = 0 \\ \left(\mathbf{F} - \widehat{\mathbf{F}}\right) + \mathbf{k}^- \left(\mathbf{W} - \widehat{\mathbf{W}}\right) + \mathbf{k}^- \mathbf{A}^T \mathbb{W} \alpha = 0 \end{cases} \tag{56}$$

A.1.3 New interpretation of the multi-scale strategy

In order to solve the problem, the Lagrange's multiplier α needs to be computed. Therefore, with the expression of the search direction and with the relations of macro equilibrium, we obtain an expression that links the multiplier with the unknown displacement. Then the search direction can be expressed without multiplier.

First we apply $\mathbb{W}^T\mathbf{A}$ to the search direction.

$$\underbrace{\mathbb{W}^T\mathbf{A}\mathbf{F}}_{=0 \text{ (macro equilibrium)}} - \mathbb{W}^T \underbrace{\mathbf{A}\widehat{\mathbf{F}}}_{=0 \text{ (local equilibrium)}} + \mathbb{W}^T\mathbf{A}\mathbf{k}^-\left(\mathbf{W}-\widehat{\mathbf{W}}\right)+\mathbb{W}^T\mathbf{A}\mathbf{k}^-\mathbf{A}^T\mathbb{W}\alpha = 0 \tag{57}$$

Thus the expression of the multiplier according to the displacement distribution is:

$$\alpha = \left(\mathbb{W}^T\mathbf{A}\mathbf{k}^-\mathbf{A}^T\mathbb{W}\right)^{-1}\left[\mathbb{W}^T\mathbf{A}\mathbf{k}^-\left(\widehat{\mathbf{W}}-\mathbf{W}\right)\right] \tag{58}$$

Once reinjected in the search direction, we obtain:

$$\mathbf{F} = \mathbf{k}^-\left[\mathbb{I}-\mathbf{A}^T\mathbb{W}\left(\mathbb{W}^T\mathbf{A}\mathbf{k}^-\mathbf{A}^T\mathbb{W}\right)^{-1}\mathbb{W}^T\mathbf{A}\mathbf{k}^-\right]\left(\widehat{\mathbf{W}}-\mathbf{W}\right)+\widehat{\mathbf{F}} \tag{59}$$

Finally the linear system to solves can be written as:

$$\boxed{\begin{array}{l}\left[\mathbf{K}+\mathbf{N}^T\mathbf{k}^-\left(\mathbb{I}-\mathbf{P}_M\right)\mathbf{N}\right]\mathbf{U} = \mathbf{f}_d+\mathbf{N}^T\widehat{\mathbf{F}}+\mathbf{N}^T\mathbf{k}^-\left(\mathbb{I}-\mathbf{P}_M\right)\widehat{\mathbf{W}} \\ \text{With } \mathbf{P}_M = \mathbf{A}^T\mathbb{W}\left(\mathbb{W}^T\mathbf{A}\mathbf{k}^-\mathbf{A}^T\mathbb{W}\right)^{-1}\mathbb{W}^T\mathbf{A}\mathbf{k}^-\end{array}} \tag{60}$$

Remarks:

- $\mathbf{P}_M$ is a projector of interface displacements on the macro-subspace. It does not contains information related to the real structure $\mathbf{K}$, only the initial (aka mono) search direction $\mathbf{k}^-$ (and the macro space $\mathbb{W}$) impacts $\mathbf{P}_M$,
- The multi-scale approach can be interpreted as a particular search direction. Indeed one can recognize the mono-scale search direction softened by the operator $\mathbf{P}_M$. It is crucial to notice that this modification is low ranked.
- $\mathbf{P}_m = \mathbb{I}-\mathbf{P}_M$ is a micro projector that filter the macro part of the displacement. Moreover the definition of this micro projector implies that the micro part of the displacement does not work beside the macro part of the force.

$$\begin{array}{ll}\ker\mathbf{P}_m = \text{ Im }\left(\mathbf{A}^T\mathbb{W}\right) & \text{Filter the macro part of displacement} \\ \text{Im }\mathbf{P}_m = \ker\left(\mathbb{W}^T\mathbf{A}\mathbf{k}^-\right) & \text{The micro part does not work beside the macro part}\end{array} \tag{61}$$

The search direction thus can be written as:

$$\begin{cases}\left(\mathbf{F}-\widehat{\mathbf{F}}\right)+\mathbf{k}^-\left(\mathbf{W}-\widehat{\mathbf{W}}\right)+\mathbf{k}^-\mathbf{A}^T\mathbb{W}\alpha = 0 \\ \alpha = \left(\mathbb{W}^T\mathbf{A}\mathbf{k}^-\mathbf{A}^T\mathbb{W}\right)^{-1}\left[\mathbb{W}^T\mathbf{A}\mathbf{k}^-\left(\widehat{\mathbf{W}}-\mathbf{W}\right)\right]\end{cases} \Longrightarrow \boxed{\mathbf{F} = \mathbf{P}_m^T\left[\widehat{\mathbf{F}}-\mathbf{k}^-\left(\mathbf{W}-\widehat{\mathbf{W}}\right)\right]} \tag{62}$$

Remarks: Contrarily to what may appear from other publications [28; 27], there is no need to introduce a force macrospace.

A.1.4 Practical computation

The matrix associated to the system to solve at the linear stage thus can be written as:

$$\mathbf{K}^M := \left[\mathbf{K}+\mathbf{N}^T\mathbf{k}^-\left(\mathbb{I}-\mathbf{P}_M\right)\mathbf{N}\right] = \underbrace{\left[\mathbf{K}+\mathbf{N}^T\mathbf{k}^-\mathbf{N}\right]}_{\mathbf{K}^m} - \underbrace{\mathbf{N}^T\mathbf{k}^-\mathbf{P}_M\mathbf{N}}_{\mathbf{k}^-\mathbf{A}^T\mathbb{W}(\mathbb{W}^T\mathbf{A}\mathbf{k}^-\mathbf{A}^T\mathbb{W})^{-1}\mathbb{W}^T\mathbf{A}\mathbf{k}^-} \tag{63}$$

The classical operator of the monoscale approach $\mathbf{K}^m$ is thus corrected by a symmetric positive matrix of rank $\dim\mathbb{W}$. It can be solved by the Shermann-Morrisson-Woodbury formula.

$$\mathbf{K}^{M^{-1}} = \mathbf{K}^{m^{-1}}+\mathbf{K}^{m^{-1}}\mathbf{N}^T\mathbf{k}^-\mathbf{A}^T\mathbb{W}\left(\mathbb{W}^T\mathbf{A}\left[\mathbf{k}^-\mathbf{N}\mathbf{K}^{m^{-1}}\mathbf{N}^T-\mathbb{I}\right]\mathbf{k}^-\mathbf{A}^T\mathbb{W}\right)^{-1}\mathbb{W}\mathbf{A}\mathbf{k}^-\mathbf{N}\mathbf{K}^{m^{-1}} \tag{64}$$

Of course the result exactly corresponds to the classical computations presented in other Latin related papers. Nevertheless, it clearly appears that what is referred to in the Latin literature as the "second micro problem" corresponds to a (macro)correction of the initial guess which only involves a recombination of already computed data ($\mathbf{K}^{m^{-1}}\mathbf{N}^T\mathbf{k}^-\mathbf{A}^T\mathbb{W}$) instead of a second local resolution as generally stated.

A.2 A macro strategy for a global continuity of displacement

For now, this strategy only applies in the case of perfect interfaces. Moreover, there is no available theory guiding the choice of the constraints except inspiration from FETI-DP approaches. For now, it is thus just a proof of feasibility. Future work will try to alleviate these issues.

A.2.1 Definition of the linear stage

For a perfect interface satisfying the behavior law $\mathbf{BW} = 0$, the macro-continuity problem can be written as:

$$\mathbb{F}^T\mathbf{BW} = 0 \tag{65}$$

where $\mathbb{F}$ is some given basis of "macroscopic forces". As said earlier, there is no properties equivalent to the Saint-Venant's principle to guide the choice of $\mathbb{F}$.

As for the global balance, the search direction is satisfying at best under the macro constraint, leading to the following modified search direction:

$$\begin{aligned}&\mathbb{F}^T\mathbf{BW} = 0\\ &\mathbf{F} - \widehat{\mathbf{F}} + \mathbf{k}^- \left(\mathbf{W} - \widehat{\mathbf{W}}\right) + \mathbf{B}^T\mathbb{F}\beta = 0\end{aligned} \tag{66}$$

where β is the Lagrange multiplier associated with the constraint.

The problem to be solved at the linear stage becomes:

$$\boxed{\begin{array}{l}\text{Find } (\mathbf{W}, \mathbf{W}, \mathbf{F}) \text{ solution to :}\\ \begin{cases}\mathbf{KU} = \mathbf{f}_d + \mathbf{N}^T\mathbf{F}\\ \mathbf{W} = \mathbf{NU}\\ \mathbb{F}^T\mathbf{BW} = 0\\ \mathbf{F} - \widehat{\mathbf{F}} + \mathbf{k}^- \left(\mathbf{W} - \widehat{\mathbf{W}}\right) + \mathbf{B}^T\mathbb{F}\beta = 0\end{cases}\end{array}} \tag{67}$$

A.2.2 Computation of the macro strategy in displacement

The expression of the Lagrange's multiplier according to the unknown $\mathbf{F}$ is obtained by using the relation of global continuity $\mathbb{F}^T\mathbf{BW} = 0$:

$$\beta = -\left(\mathbb{F}^T\mathbf{B}\mathbf{k}^{-^{-1}}\mathbf{B}^T\mathbb{F}\right)^{-1}\mathbb{F}^T\mathbf{B}\left[\mathbf{k}^{-^{-1}}\left(\mathbf{F} - \widehat{\mathbf{F}}\right) - \widehat{\mathbf{W}}\right] \tag{68}$$

By injecting the expression of the multiplier in the search direction we get the relation between the unknown $\mathbf{W}$ and $\mathbf{F}$:

$$\left(\mathbb{I} - \mathbf{Q}_M\right)\left(\mathbf{F} - \widehat{\mathbf{F}}\right) + \mathbf{k}^-(\mathbf{W} - \widehat{\mathbf{W}}) = 0 \tag{69}$$

with $\mathbf{Q}_M = \mathbf{B}^T\mathbb{F}\left(\mathbb{F}^T\mathbf{B}\mathbf{k}^{-^{-1}}\mathbf{B}^T\mathbb{F}\right)^{-1}\mathbb{F}^T\mathbf{B}\mathbf{k}^{-^{-1}}$

Remarks:

- $\mathbf{Q}_M$ is a macro projector. As $\mathbf{P}_M$ it does not depends of the stiffness of the sub-domains but only on the search direction
- We cannot have directly the expression of $\mathbf{F}$ as function of $\mathbf{W}$ since only the micro part $\left(\mathbb{I} - \mathbf{Q}_M\right)\mathbf{F}$ of force is related to the displacement. The macro part of the force $\mathbf{Q}_M\mathbf{F}$ is on the form : $\mathbf{B}^T\mathbb{F}\gamma$

  We can introduce in the equilibrium relation the search direction:

$$\mathbf{KU} = \mathbf{f}_d + \mathbf{N}^T \underbrace{\left(\mathbb{I} - \mathbf{Q}_M\right)\mathbf{F}}_{\text{micro part via search direction}} + \mathbf{N}^T \underbrace{\mathbf{Q}_M\mathbf{F}}_{\text{macro part: } \mathbf{B}^T\mathbb{F}\gamma^{t+\Delta t}} \tag{70}$$

We have now a problem with two unknown: γ and $\mathbf{U}$. γ is determined by using the relation of global continuity after expressing $\mathbf{U}$ as a function of γ:

$$\gamma = -\left(\mathbb{F}^T\mathbf{BN}\left(\mathbf{K} + \mathbf{N}^T\mathbf{k}^-\mathbf{N}\right)^{-1}\mathbf{N}^T\mathbf{B}^T\mathbb{F}\right)^{-1}\mathbb{F}^T\mathbf{BN}\left(\mathbf{K} + \mathbf{N}^T\mathbf{k}^-\mathbf{N}\right)^{-1}\left[\mathbf{f}_d + \mathbf{N}^T\left(\mathbb{I} - \mathbf{Q}_M\right)\widehat{\mathbf{F}}\right] \tag{71}$$

By re-injecting the expression of γ in the equilibrium relation we obtain:

$$\left(\mathbf{K} + \mathbf{N}^T\mathbf{k}^-\mathbf{N}\right)\mathbf{U} = \left(\mathbb{I} - \mathbf{R}\right)\left[\mathbf{f}_d + \mathbf{N}^T\widehat{\mathbf{F}}\right] \tag{72}$$

with $\mathbf{R} = \mathbf{N}^T\mathbf{B}^T\mathbb{F}\left(\mathbb{F}^T\mathbf{BN}\left(\mathbf{K} + \mathbf{N}^T\mathbf{k}^-\mathbf{N}\right)^{-1}\mathbf{N}^T\mathbf{B}^T\mathbb{F}\right)^{-1}\mathbb{F}^T\mathbf{BN}\left(\mathbf{K} + \mathbf{N}^T\mathbf{k}^-\mathbf{N}\right)^{-1}$. We have used the fact that $\left(\mathbb{I} - \mathbf{R}\right)\mathbf{N}^T\left(\mathbb{I} - \mathbf{Q}_M\right) = 0$ to simplify the expression.

Contrarily to the global balance approach, it seems that the macro continuity constraint can not pas interpreted as a modification of the monoscale search direction.

C Article [Negrello *et al.*, 2016], méthodes de décomposition de domaine non-linéaires

Cet article présente les méthodes de décomposition de domaine non-linéaire. Sous réserve de pouvoir résoudre des problèmes locaux à conditions d'interface de Dirichlet, Neumann ou Robin, nous proposons :

- de généraliser les formulations primale/duale/mixte de décomposition de domaine au cas non-linéaire ;
- d'obtenir un algorithme de résolution qui alterne résolutions non-linéaires indépendantes par sous-domaines (en parallèle) et résolution d'un problème d'interface linéaire assemblé ayant la structure usuelle (additive) des décompositions de domaine ;
- d'adapter les critères de convergence des solveurs imbriqués pour minimiser les calculs.

Les méthodes sont testées sur quelques cas académiques, on voit que le choix de la condition d'interface est crucial. Le potentiel de la méthode mixte est mis en évidence. Un article proposant une heuristique de calcul de la condition de Robin est en cours de rédaction.

Substructured formulations of nonlinear structure problems – Influence of the interface condition

Camille Negrello[1], Pierre Gosselet[1], Christian Rey[1,2] and Julien Pebrel[1]
(1) LMT Cachan, ENS Cachan/CNRS/Univ. Paris Saclay,
61 Avenue du Président Wilson, 94235 Cachan France.
(2) Safran Tech, rue des jeunes bois, Chateaufort,
CS 80112, 78772 Magny-les-Hameaux France.

June 15, 2016

Abstract

We investigate the use of non-overlapping domain decomposition (DD) methods for nonlinear structure problems. The classic techniques would combine a global Newton solver with a linear DD solver for the tangent systems. We propose a framework where we can swap Newton and DD, so that we solve independent nonlinear problems for each substructure and linear condensed interface problems. The objective is to decrease the number of communications between subdomains and to improve parallelism. Depending on the interface condition, we derive several formulations which are not equivalent, contrarily to the linear case. Primal, dual and mixed variants are described and assessed on a simple plasticity problem.

Keywords: Domain decomposition; nonlinear mechanics; Newton solver; Krylov solver; parallel processing

1 Introduction

In order to solve large nonlinear structure problems, an efficient strategy is to combine a Newton-based solver which leads to a sequence of linear systems, and a domain decomposition approach to solve the tangent systems. Indeed such a strategy combines well-known and robust methods for which many refinement are available: Newton can be tangent/constant/secant/modified/arc-length [24], and domain decomposition solvers [33, 13, 32, 10, 17] can be equipped with preconditioners and coarse problems which make them reliable and scalable [26, 27, 44, 19]. Moreover, the computation of stiffness matrices being done in parallel independently for each subdomain, and information being reusable from one system to another [37, 39, 18], the overall performance is very satisfying in general [11, 3].

But there are cases where such an approach is not as pertinent as expected. For instance, when dealing with strong localized nonlinearity, many (global) Newton iterations are required, whereas most of the structure undergoes a linear evolution. In this situation, it would be interesting to differentiate between the linear or nonlinear nature of each subdomain's behavior, in order to decrease the number of global iterations and communications. The possibility to conduct local nonlinear computations has been investigated for a long time in the Schwarz framework (with or without overlap between subdomains) [1, 8, 30, 4, 31, 22]. More recently the possibility to define nonlinear versions of the Schur complement methods (often called nonlinear relocalization techniques) was studied: BDD in [6, 2], FETI in [35], FETI-DP in [25], BDDC in [21, 25].

The aim of this paper is to give a formal framework to develop the nonlinear versions of the well-known linear solver FETI [13], BDD [33, 32] and FETI2LM[41]. They rely on the concept of nonlinear condensation which was exposed in conferences [36] and seminars [16].

In section 2 and 3, we present the nonlinear system in its monolithic and substructured form. In section 4, we introduce the concept of nonlinear condensation in its primal, dual and mixed versions. In section 5, we show that applying a Newton method to these condensed formulations leads, at each outer iteration, to the parallel solution of local nonlinear systems with, depending on the formulation, Dirichlet, Neumann or Robin boundary conditions, and to the linear interface systems which can be solved by classical BDD, FETI or FETI2LM. Iterative solvers being involved inside the outer loop, we interpret the method as an inexact Newton solver [7] which leads us, in section 6, to tune the convergence thresholds depending on the current residual in order to avoid oversolving. First assessments are given in section 7 on a simple problem, yet representative of the nonlinearity encountered in industrial problems; a discussion ends the paper.

2 Reference problem

We consider the classic problem of the evolution, under the small perturbation hypothesis, of a structure occupying the domain Ω, submitted to body forces f, to traction forces F on the part $\partial_f\Omega$ of its boundary and to given displacements

u_g on the complementary part $\partial_u\Omega \neq \varnothing$. Note that the small perturbation hypothesis is crucial for the handling of rigid body motions in the dual approach, but it can be relaxed in other cases.

Let $\varepsilon(u)$ be the symmetric part of the gradient of displacement and σ the Cauchy stress tensor. The problem to be solved can be written as:

$$\begin{aligned}
&\text{at time } t \in [0,T],\ \text{find } u \in \mathcal{U} \ /\ \forall v \in \mathcal{U}_0 \\
&\int_\Omega \sigma : \varepsilon(v) d\Omega = \int_\Omega v^T f d\Omega + \int_{\partial_f\Omega} v^T F dS \\
&\sigma = \sigma(\varepsilon(u(\tau)), \tau \in [0,t])
\end{aligned}$$

where $\mathcal{U}$ is the space of kinematically admissible fields and $\mathcal{U}_0$ is the associated vector space:

$$\mathcal{U} = \left\{ u \in H^1(\Omega),\ u = u_g \text{ on } \partial_u\Omega \right\}$$

The notation $\sigma(\varepsilon(u(\tau)), \tau \in [0,t])$ means that the stress at one point depends on the whole history of the strain at this point (local nonlinear behavior). This history is most often materialized by internal variables like anelastic strain, hardening or damage.

The problem is discretized in space using the finite element method. The domain Ω is meshed; let $\boldsymbol{N}$ be the matrix of shape functions, and $\boldsymbol{u}$ be the nodal displacement unknowns such that $u = \boldsymbol{N}\boldsymbol{u}$. The problem is also supposed to be discretized in time, the discrete reference problem to be solved at Step t_n can be written as:

$$\text{Find } \boldsymbol{u}(t_n) \text{ so that } \boldsymbol{f}_{int}(\boldsymbol{u}) + \boldsymbol{f}_{ext} = 0 \tag{1}$$

with

$$\begin{aligned}
&\boldsymbol{f}_{ext} = \int_\Omega \boldsymbol{N}^T f d\Omega + \int_{\partial_f\Omega} \boldsymbol{N}^T F dS \\
&\boldsymbol{v}^T \boldsymbol{f}_{int} = -\int_\Omega \sigma_h : \varepsilon(\boldsymbol{N}\boldsymbol{v}) d\Omega \\
&\sigma_h = \sigma_h(\varepsilon(\boldsymbol{N}\boldsymbol{u}(t_j)),\ j \leqslant n)
\end{aligned}$$

σ_h is the discrete counterpart of σ, it depends of the whole discrete history. Integrals are classically computed numerically using Gauss quadrature, so that stress and internal variables are defined at Gauss points. Throughout the rest of the document, the time step will not be mentioned. In order to shorten expressions, Dirichlet boundary conditions u_g are implicitly taken into account within $\boldsymbol{f}_{int}$ and $\boldsymbol{f}_{ext}$.

Note that in the linear case we have: $\boldsymbol{f}_{int}(\boldsymbol{u}) = -\boldsymbol{K}\boldsymbol{u}$ where $\boldsymbol{K}$ is the stiffness matrix.

The classical solution strategy to Problem (1) is to use a Newton-Raphson algorithm to linearize the problem, and then solve a sequence of tangent systems.

3 Substructured formulation

We consider the conforming partition of Ω into N non-overlapping subdomains $\Omega^{(s)}$, so that each element exactly belongs to one subdomain. Superscript (s) will refer to data attached to domain $\Omega^{(s)}$. In order to ease the treatment of groups of subdomains, we define the following block notations:

$$\boldsymbol{x}^{\diamondsuit} = \begin{pmatrix} \vdots \\ \boldsymbol{x}^{(s)} \\ \vdots \end{pmatrix}, \quad \boldsymbol{x}^{\diamondsuit} = \begin{pmatrix} \dots & \boldsymbol{x}^{(s)} & \dots \end{pmatrix}, \quad \boldsymbol{x}^{\diamondsuit} = \begin{pmatrix} \ddots & & \boldsymbol{0} \\ & \boldsymbol{x}^{(s)} & \\ \boldsymbol{0} & & \ddots \end{pmatrix}$$

Note that in the case of nonlinearly-dependent data, we use the same notation but the dependence is implicitly local:

$$\boldsymbol{f}_{int}^{\diamondsuit}(\boldsymbol{u}^{\diamondsuit}) = \begin{pmatrix} \vdots \\ \boldsymbol{f}_{int}^{(s)}(\boldsymbol{u}^{(s)}) \\ \vdots \end{pmatrix}$$

The nodes on the interface between two subdomains play a specific role which we need to highlight. We note $\Gamma^{(i,j)}$ the set of nodes shared by subdomains $\Omega^{(i)}$ and $\Omega^{(j)}$, and $\Gamma^{(s)} = \bigcup_j \Gamma^{(s,j)}$ the interface nodes of subdomain $\Omega^{(s)}$. We use the subscript b for nodes belonging to the interface $\Gamma^{(s)}$ and the subscript i for internal nodes. The trace operator $\boldsymbol{t}^{(s)}$ extracts interface nodal values (on $\Gamma^{(s)}$) from subdomain data (in $\Omega^{(s)}$):

$$\begin{aligned}
&\forall s \in [\![1,N]\!],\ \boldsymbol{t}^{(s)}\boldsymbol{u}^{(s)} = \boldsymbol{u}_b^{(s)} \text{ which also can be written as } \boldsymbol{t}^{\diamondsuit}\boldsymbol{u}^{\diamondsuit} = \boldsymbol{u}_b^{\diamondsuit} \\
&\text{assuming adapted ordering, } \boldsymbol{t}^{(s)} = \begin{pmatrix} \boldsymbol{0}_{bi}^{(s)} & \boldsymbol{I}_{bb}^{(s)} \end{pmatrix}
\end{aligned}$$

See Figure 1(a,b) for an example. We note $\mathbb{R}_b^{\diamondsuit} = \text{Range}(\boldsymbol{t}^{\diamondsuit})$ the space of local interface vectors to which any vector $\boldsymbol{x}_b^{\diamondsuit}$ belongs.

Let $\Gamma_A = \bigcup \Gamma^{(s)}$ denote the totality of interface nodes, and let $\boldsymbol{A}^{(s)}$ be the canonical operator which injects nodes from $\Gamma^{(s)}$ to Γ_A. $\boldsymbol{A}^{(s)}$ is a boolean full column-rank matrix. With one local interface node in $\Gamma^{(s)}$ associated with exactly one node in Γ_A, the global assembling operator $\boldsymbol{A}^{\diamondsuit}$ has no left-kernel. It is then a full row-rank matrix (sometimes called a primal assembling operator). See Figure 1(b,c) for an example. We note $\mathbb{R}^{\Gamma_A} = \text{Range}(\boldsymbol{A}^{\diamondsuit})$ the space of vectors defined on that "primal" interface.

Any matrix $\boldsymbol{B}^{\diamondsuit}$ satisfying $\text{Range}(\boldsymbol{B}^{\diamondsuit^T}) = \text{Ker}(\boldsymbol{A}^{\diamondsuit})$ can be used as a dual assembling operator. In Figure 1(b,d) we give the most classical choice where $\boldsymbol{B}^{\diamondsuit}$ is a signed boolean matrix which describes each connection between interface nodes; we note Γ_B the set of connections. Note that operator $\boldsymbol{B}^{\diamondsuit}$ needs not to be full row-rank (in the classical case it is not whenever one interface degree of freedom is shared by more than two subdomains). We note $\mathbb{R}^{\Gamma_B} = \text{Range}(\boldsymbol{B}^{\diamondsuit})$ the space of vectors defined on that "dual" interface.

In order to decouple equations betweens subdomains, we introduce the nodal reaction $\boldsymbol{\lambda}_b^{(s)}$ imposed on subdomain $\Omega^{(s)}$ by its neighbors. The substructured reference problem can be written as:

$$\begin{aligned} &\text{Find } (\boldsymbol{u}^{\diamondsuit}, \boldsymbol{\lambda}_b^{\diamondsuit}) \text{ such that} \\ &\boldsymbol{f}_{int}^{\diamondsuit}(\boldsymbol{u}^{\diamondsuit}) + \boldsymbol{f}_{ext}^{\diamondsuit} + \boldsymbol{t}^{\diamondsuit^T} \boldsymbol{\lambda}_b^{\diamondsuit} = 0 \\ &\boldsymbol{A}^{\diamondsuit} \boldsymbol{\lambda}_b^{\diamondsuit} = 0 \\ &\boldsymbol{B}^{\diamondsuit} \boldsymbol{u}_b^{\diamondsuit} = 0 \end{aligned} \tag{2}$$

where the first equation expresses the equilibrium of each subdomain submitted to given forces and to unknown interface reactions $\boldsymbol{\lambda}_b^{\diamondsuit}$. The primal assembling operator $\boldsymbol{A}^{\diamondsuit}$ enables us to express the action-reaction principle while the dual assembling operator $\boldsymbol{B}^{\diamondsuit}$ enables us to express the continuity of the displacement field. In addition to Figure 1, note that in the case of two subdomains these operators can be written as $\boldsymbol{A}^{\diamondsuit} \boldsymbol{\lambda}_b^{\diamondsuit} = \boldsymbol{\lambda}_b^{(1)} + \boldsymbol{\lambda}_b^{(2)}$ and $\boldsymbol{B}^{\diamondsuit} \boldsymbol{u}_b^{\diamondsuit} = \boldsymbol{u}_b^{(1)} - \boldsymbol{u}_b^{(2)}$.

Assembling operators satisfy the following relationships:

- Assembling operators are orthogonal in the following sense:

$$\boldsymbol{A}^{\diamondsuit} \boldsymbol{B}^{\diamondsuit^T} = 0 \tag{3}$$

- Assembling operators generate local interface nodal vectors:

$$\text{Range}(\boldsymbol{A}^{\diamondsuit^T}) \overset{\perp}{\oplus} \text{Range}(\boldsymbol{B}^{\diamondsuit^T}) = \mathbb{R}_b^{\diamondsuit} \tag{4}$$

- Any local interface vector is uniquely defined as a combination of a balanced vector $\boldsymbol{B}^{\diamondsuit^T} \boldsymbol{x}_B$ and a continuous vector $\boldsymbol{A}^{\diamondsuit^T} \boldsymbol{x}_A$:

$$\begin{aligned} &\forall \boldsymbol{x}_b^{\diamondsuit},\ \exists (\boldsymbol{x}_B, \boldsymbol{x}_A) \in \mathbb{R}^{\Gamma_B} \times \mathbb{R}^{\Gamma_A} / \ \boldsymbol{x}_b^{\diamondsuit} = \boldsymbol{B}^{\diamondsuit^T} \boldsymbol{x}_B + \boldsymbol{A}^{\diamondsuit^T} \boldsymbol{x}_A \\ &\text{indeed } \begin{cases} \boldsymbol{x}_A = \left(\boldsymbol{A}^{\diamondsuit} \boldsymbol{A}^{\diamondsuit^T} \right)^{-1} \boldsymbol{A}^{\diamondsuit} \boldsymbol{x}_b^{\diamondsuit} \\ \boldsymbol{x}_B = \left(\boldsymbol{B}^{\diamondsuit} \boldsymbol{B}^{\diamondsuit^T} \right)^{+} \boldsymbol{B}^{\diamondsuit} \boldsymbol{x}_b^{\diamondsuit} \end{cases} \end{aligned} \tag{5}$$

  The use of the pseudo-inverse $\left(\boldsymbol{B}^{\diamondsuit} \boldsymbol{B}^{\diamondsuit^T} \right)^{+}$ is due to the potential presence of redundancies in the description of the connectivity between subdomains, though it is applied to a vector which belongs to $\text{Range}\left(\boldsymbol{B}^{\diamondsuit} \right) = \text{Range}\left(\boldsymbol{B}^{\diamondsuit} \boldsymbol{B}^{\diamondsuit^T} \right)$ so that $\boldsymbol{x}_B$ is well defined, and $\boldsymbol{B}^{\diamondsuit^T} \boldsymbol{x}_B$ does not depend on the choice of the pseudo-inverse.

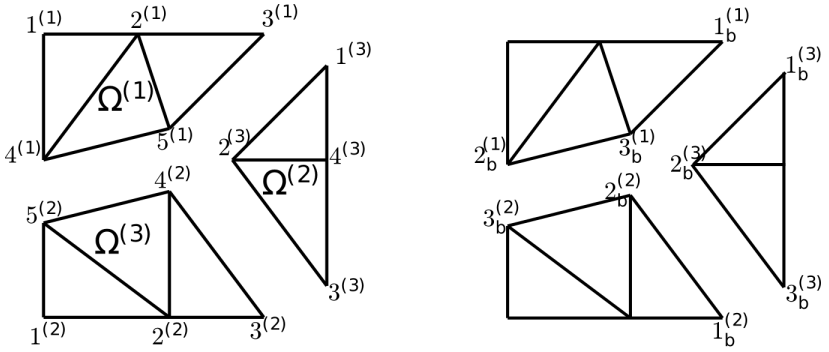
4 Nonlinear condensations

The substructured formulation (2) is strictly equivalent to the global formulation (1). We now propose various solution strategies which (under certain assumptions) all converge to the reference solution, though these methods, which also employ Newton-Raphson algorithm, are meant to generate a different sequence of linear systems.

4.1 Primal formulation

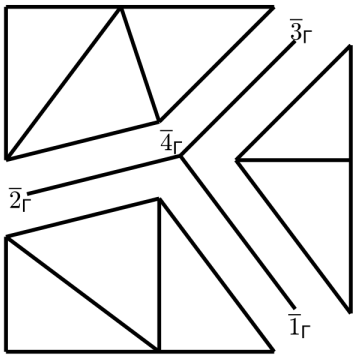
The primal formulation consists in rewriting system (2) in terms of one unknown interface displacement field $\boldsymbol{u}_A$:

$$\begin{aligned} &\text{Find } \boldsymbol{u}_A \in \mathbb{R}^{\Gamma_A} \text{ such that } \boldsymbol{A}^{\diamondsuit} \boldsymbol{\lambda}_b^{\diamondsuit} = 0 \\ &\text{where } \boldsymbol{\lambda}_b^{\diamondsuit} := -\left[\boldsymbol{f}_{int}^{\diamondsuit}(\boldsymbol{u}^{\diamondsuit}) + \boldsymbol{f}_{ext}^{\diamondsuit} \right]_b \\ &\text{and } \boldsymbol{u}^{\diamondsuit} \text{ solves } \begin{cases} \left[\boldsymbol{f}_{int}^{\diamondsuit}(\boldsymbol{u}^{\diamondsuit}) + \boldsymbol{f}_{ext}^{\diamondsuit} \right]_i = 0 \\ \boldsymbol{t}^{\diamondsuit} \boldsymbol{u}^{\diamondsuit} = \boldsymbol{A}^{\diamondsuit^T} \boldsymbol{u}_A \end{cases} \end{aligned} \tag{6}$$

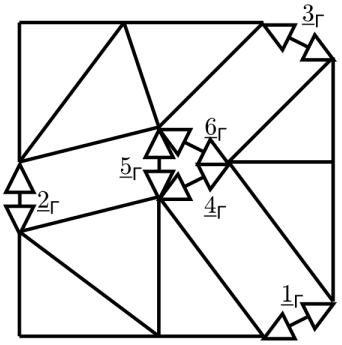


(a) Subdomains

(b) Local interface



(c) Primal interface



(d) Dual interface

$$\boldsymbol{t}^{\diamondsuit} = \begin{pmatrix} \begin{pmatrix} 0&0&1&0&0\\0&0&0&1&0\\0&0&0&0&1 \end{pmatrix} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \begin{pmatrix} 0&0&1&0&0\\0&0&0&1&0\\0&0&0&0&1 \end{pmatrix} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \begin{pmatrix} 1&0&0&0\\0&1&0&0\\0&0&1&0 \end{pmatrix} \end{pmatrix}$$

$$\boldsymbol{A}^{\diamondsuit} = \begin{pmatrix} \begin{pmatrix} 0&0&0\\0&1&0\\1&0&0\\0&0&1 \end{pmatrix} & \begin{pmatrix} 1&0&0\\0&0&1\\0&0&0\\0&1&0 \end{pmatrix} & \begin{pmatrix} 0&0&1\\0&0&0\\1&0&0\\0&1&0 \end{pmatrix} \end{pmatrix}$$

$$\boldsymbol{B}^{\diamondsuit} = \begin{pmatrix} \begin{pmatrix} 0&0&0\\0&1&0\\1&0&0\\0&0&0\\0&0&1\\0&0&1 \end{pmatrix} & \begin{pmatrix} 1&0&0\\0&0&-1\\0&0&0\\0&1&0\\0&-1&0\\0&0&0 \end{pmatrix} & \begin{pmatrix} 0&0&-1\\0&0&0\\-1&0&0\\0&-1&0\\0&0&0\\0&-1&0 \end{pmatrix} \end{pmatrix}$$

Figure 1: Local numberings, interface numberings, trace and assembly operators

The last set of equations corresponds to the solution to independent mechanical problems for each subdomain with imposed displacement at the interface (a Dirichlet condition which implies that the continuity is automatically insured: $\boldsymbol{u}_b^{\diamondsuit} = \boldsymbol{A}^{\diamondsuit^T}\boldsymbol{u}_A \Rightarrow \boldsymbol{B}^{\diamondsuit}\boldsymbol{u}_b^{\diamondsuit} = 0$). This displacement has to be found so that the associated reactions $\boldsymbol{\lambda}_b^{\diamondsuit}$ are balanced on the interface.

If we assume that the last set of equations has a unique solution for any imposed interface displacement, then we can define an operator $\boldsymbol{S}_{nl}^{(s)}$ so that:

$$\boldsymbol{\lambda}_b^{(s)} = \boldsymbol{S}_{nl}^{(s)}(\boldsymbol{A}^{(s)^T}\boldsymbol{u}_A; \boldsymbol{f}_{ext}^{(s)}) \tag{7}$$

This operator can be viewed as a nonlinear version of the Schur complement; it computes the reaction associated with a given displacement. In the linear case, an explicit expression can be given:

$$\begin{aligned} \boldsymbol{S}_l^{(s)}(\boldsymbol{u}_b^{(s)}; \boldsymbol{f}_{ext}^{(s)}) &= \boldsymbol{S}_t^{(s)}\boldsymbol{u}_b^{(s)} - \boldsymbol{b}_p^{(s)} \\ \text{with } \boldsymbol{S}_t^{(s)} &= \boldsymbol{K}_{bb}^{(s)} - \boldsymbol{K}_{bi}^{(s)}\boldsymbol{K}_{ii}^{(s)^{-1}}\boldsymbol{K}_{ib}^{(s)} \\ \boldsymbol{b}_p^{(s)} &= \boldsymbol{f}_{ext\,b}^{(s)} - \boldsymbol{K}_{bi}^{(s)}\boldsymbol{K}_{ii}^{(s)^{-1}}\boldsymbol{f}_{ext\,i}^{(s)} \end{aligned} \tag{8}$$

The primal nonlinear condensed system reads:

$$\boxed{\begin{array}{l}\text{Find } \boldsymbol{u}_A \in \mathbb{R}^{\Gamma_A} \text{ such that} \\ \boldsymbol{A}^{\diamondsuit} \boldsymbol{S}_{nl}^{\diamondsuit}(\boldsymbol{A}^{\diamondsuit^T} \boldsymbol{u}_A; \boldsymbol{f}_{ext}^{\diamondsuit}) = 0\end{array}} \tag{9}$$

4.2 Dual formulation

The dual formulation consists in rewriting system (2) in terms of one unknown interface reaction field:

$$\begin{array}{l}\text{Find } \boldsymbol{\lambda}_B \in \mathbb{R}^{\Gamma_B} \text{ such that } \boldsymbol{B}^{\diamondsuit} \boldsymbol{t}^{\diamondsuit} \boldsymbol{u}^{\diamondsuit} = 0 \\ \text{where } \boldsymbol{u}^{\diamondsuit} \text{ solves } \boldsymbol{f}_{int}^{\diamondsuit}(\boldsymbol{u}^{\diamondsuit}) + \boldsymbol{f}_{ext}^{\diamondsuit} + \boldsymbol{t}^{\diamondsuit^T} \boldsymbol{B}^{\diamondsuit^T} \boldsymbol{\lambda}_B = 0\end{array} \tag{10}$$

The last equation corresponds to the solution to independent mechanical problems for each subdomain with imposed traction at the interface (local Neumann problems). This traction has to be found so that displacements are continuous at the interface. Implicitly local reactions have been defined by $\boldsymbol{\lambda}_b^{\diamondsuit} = \boldsymbol{B}^{\diamondsuit^T} \boldsymbol{\lambda}_B$, which guarantees the interface equilibrium $\boldsymbol{A}^{\diamondsuit} \boldsymbol{\lambda}_b^{\diamondsuit} = 0$.

In order for problem (10) to be well-posed, one necessary condition is the self-equilibrium of the substructure. Let $\boldsymbol{R}^{(s)}$ be the basis of kinematically admissible rigid body motions of subdomain $\Omega^{(s)}$ (if they exist, these displacements are exactly represented in the finite element function space), then we have $\boldsymbol{R}^{(s)^T} \boldsymbol{f}_{int}^{(s)} = 0$ (internal forces do not develop work within rigid body motions). This implies the admissibility condition for $\boldsymbol{\lambda}_B$:

$$\boldsymbol{R}^{\diamondsuit^T} \left(\boldsymbol{f}_{ext}^{\diamondsuit} + \boldsymbol{t}^{\diamondsuit^T} \boldsymbol{B}^{\diamondsuit^T} \boldsymbol{\lambda}_B \right) = 0 \tag{11}$$

If we assume that (10) has a unique solution (up to a rigid body motion $\boldsymbol{R}^{\diamondsuit} \boldsymbol{\alpha}^{\diamondsuit}$, where $\boldsymbol{\alpha}^{\diamondsuit}$ is the unknown amplitude of rigid body motions) for any imposed interface traction, then we can define an operator $\boldsymbol{F}_{nl}^{(s)}$ such that:

$$\boldsymbol{u}_b^{(s)} = \boldsymbol{F}_{nl}^{(s)}(\boldsymbol{B}^{(s)^T} \boldsymbol{\lambda}_B; \boldsymbol{f}_{ext}^{(s)}) + \boldsymbol{t}^{(s)} \boldsymbol{R}^{(s)} \boldsymbol{\alpha}^{(s)} \tag{12}$$

This operator can be viewed as a nonlinear version of the dual Schur complement (as employed in classical FETI methods [12] of which we borrow the notation $\boldsymbol{F}$); it computes the displacement associated with given reaction. In the linear case, an explicit expression can be given:

$$\begin{array}{rl}\boldsymbol{F}_l^{(s)}(\boldsymbol{\lambda}_b^{(s)}; \boldsymbol{f}_{ext}^{(s)}) &= \boldsymbol{F}_t^{(s)} \boldsymbol{\lambda}_b^{(s)} + \boldsymbol{b}_d^{(s)} \\ \text{with } \boldsymbol{F}_t^{(s)} &= \boldsymbol{t}^{(s)} \boldsymbol{K}^{(s)^+} \boldsymbol{t}^{(s)^T} \\ \boldsymbol{b}_d^{(s)} &= \boldsymbol{t}^{(s)} \boldsymbol{K}^{(s)^+} \boldsymbol{f}_{ext}^{(s)}\end{array} \tag{13}$$

We recall the following classical relationships [17]:

$$\begin{array}{rl}\boldsymbol{R}^{(s)} &= \mathrm{Ker}(\boldsymbol{K}^{(s)}) \\ \boldsymbol{R}_b^{(s)} &= \boldsymbol{t}^{(s)} \boldsymbol{R}^{(s)} = \mathrm{Ker}(\boldsymbol{S}_t^{(s)}) \\ \boldsymbol{R}^{(s)^T} \boldsymbol{f}_{ext}^{(s)} &= \boldsymbol{R}_b^{(s)^T} \boldsymbol{b}_p^{(s)}\end{array} \tag{14}$$

Finally, the dual nonlinear condensed system reads:

$$\boxed{\begin{array}{l}\text{Find } \boldsymbol{\lambda}_B \in \mathbb{R}^{\Gamma_B}, \boldsymbol{\alpha}^{\diamondsuit} \text{ such that} \\ \begin{cases} \boldsymbol{B}^{\diamondsuit} \left(\boldsymbol{F}_{nl}^{\diamondsuit}(\boldsymbol{B}^{\diamondsuit^T} \boldsymbol{\lambda}_B; \boldsymbol{f}_{ext}^{\diamondsuit}) + \boldsymbol{t}^{\diamondsuit} \boldsymbol{R}^{\diamondsuit} \boldsymbol{\alpha}^{\diamondsuit} \right) = 0 \\ \boldsymbol{R}^{\diamondsuit^T} \left(\boldsymbol{f}_{ext}^{\diamondsuit} + \boldsymbol{t}^{\diamondsuit^T} \boldsymbol{B}^{\diamondsuit^T} \boldsymbol{\lambda}_B \right) = 0 \end{cases}\end{array}} \tag{15}$$

4.3 Mixed formulation

The mixed formulation consists in introducing a new interface variable for each subdomain $\boldsymbol{\mu}_b^{(s)}$:

$$\boldsymbol{\mu}_b^{\diamondsuit} = \boldsymbol{\lambda}_b^{\diamondsuit} + \boldsymbol{Q}_b^{\diamondsuit} \boldsymbol{u}_b^{\diamondsuit} \tag{16}$$

The symmetric positive definite matrix $\boldsymbol{Q}_b^{\diamondsuit}$ is a parameter of the method. It can be interpreted as a stiffness (or an impedance) added to the interface (like in Robin-type boundary conditions).

The properties of the matrix $\boldsymbol{Q}_b^{\diamondsuit}$ imply a partitioning of the space of subdomain interface vectors:

$$\mathrm{Ker}(\boldsymbol{B}^{\diamondsuit}) \oplus \mathrm{Ker}(\boldsymbol{A}^{\diamondsuit}\boldsymbol{Q}_b^{\diamondsuit}) = \mathbb{R}_b^{\diamondsuit} \tag{17}$$

This property enables us to reformulate both interface conditions (2) in one single boundary equation:

$$\boldsymbol{A}^{\diamondsuit^T}\left(\boldsymbol{A}^{\diamondsuit}\boldsymbol{Q}_b^{\diamondsuit}\boldsymbol{A}^{\diamondsuit^T}\right)^{-1}\boldsymbol{A}^{\diamondsuit}\boldsymbol{\mu}_b^{\diamondsuit} - \boldsymbol{u}_b^{\diamondsuit} = 0 \tag{18}$$

indeed when multiplying this equation on the left by $\boldsymbol{B}^{\diamondsuit}$ or $\boldsymbol{A}^{\diamondsuit}\boldsymbol{Q}_b^{\diamondsuit}$ we recover classical interface conditions.

The mixed formulation of Problem (2) can be written as:

$$\begin{aligned}&\text{Find } \boldsymbol{\mu}_b^{\diamondsuit} \in \mathbb{R}_b^{\diamondsuit} \text{ such that } \boldsymbol{A}^{\diamondsuit^T}\left(\boldsymbol{A}^{\diamondsuit}\boldsymbol{Q}_b^{\diamondsuit}\boldsymbol{A}^{\diamondsuit^T}\right)^{-1}\boldsymbol{A}^{\diamondsuit}\boldsymbol{\mu}_b^{\diamondsuit} - \boldsymbol{u}_b^{\diamondsuit} = 0\\ &\text{where } \boldsymbol{u}^{\diamondsuit} \text{ solves } \boldsymbol{f}_{int}^{\diamondsuit}(\boldsymbol{u}^{\diamondsuit}) - \boldsymbol{t}^{\diamondsuit^T}\boldsymbol{Q}_b^{\diamondsuit}\boldsymbol{t}^{\diamondsuit}\boldsymbol{u}^{\diamondsuit} + \boldsymbol{t}^{\diamondsuit^T}\boldsymbol{\mu}_b^{\diamondsuit} + \boldsymbol{f}_{ext}^{\diamondsuit} = 0\end{aligned} \tag{19}$$

The last equation corresponds to the solution of independent nonlinear problems for each subdomain under Robin boundary conditions. If we assume that it has a unique solution for any given $\boldsymbol{\mu}_b^{\diamondsuit}$ (the addition of matrix $\boldsymbol{Q}_b^{\diamondsuit}$ suppresses rigid body motions), then we can define a nonlinear interface operator:

$$\boldsymbol{u}_b^{(s)} = \boldsymbol{M}_{nl}^{(s)}(\boldsymbol{\mu}_b^{(s)}; \boldsymbol{f}_{ext}^{(s)}, \boldsymbol{Q}_b^{(s)}) \tag{20}$$

In the linear case, the operator can be written as:

$$\begin{aligned}&\boldsymbol{M}_l^{(s)}(\boldsymbol{\mu}_b^{(s)}; \boldsymbol{f}_{ext}^{(s)}, \boldsymbol{Q}_b^{(s)}) = \boldsymbol{M}_t^{(s)}\boldsymbol{\mu}_b^{(s)} + \boldsymbol{b}_m^{(s)}\\ &\text{with } \boldsymbol{M}_t^{(s)} = \boldsymbol{t}^{(s)}(\boldsymbol{K}^{(s)} + \boldsymbol{t}^{(s)^T}\boldsymbol{Q}_b^{(s)}\boldsymbol{t}^{(s)})^{-1}\boldsymbol{t}^{(s)^T}\\ &\boldsymbol{b}_m^{(s)} = \boldsymbol{t}^{(s)}(\boldsymbol{K}^{(s)} + \boldsymbol{t}^{(s)^T}\boldsymbol{Q}_b^{(s)}\boldsymbol{t}^{(s)})^{-1}\boldsymbol{f}_{ext}^{(s)}\end{aligned} \tag{21}$$

One can recognize the operators of the FETI2LM approach [41].

The mixed nonlinear condensed problem can be written as:

$$\boxed{\begin{aligned}&\text{Find } \boldsymbol{\mu}_b^{\diamondsuit} \in \mathbb{R}_b^{\diamondsuit} \text{ such that}\\ &\boldsymbol{A}^{\diamondsuit^T}\left(\boldsymbol{A}^{\diamondsuit}\boldsymbol{Q}_b^{\diamondsuit}\boldsymbol{A}^{\diamondsuit^T}\right)^{-1}\boldsymbol{A}^{\diamondsuit}\boldsymbol{\mu}_b^{\diamondsuit} - \boldsymbol{M}_{nl}^{\diamondsuit}(\boldsymbol{\mu}_b^{\diamondsuit}; \boldsymbol{f}_{ext}^{\diamondsuit}, \boldsymbol{Q}_b^{\diamondsuit}) = 0\end{aligned}} \tag{22}$$

Note that the method makes use of the operator $\left(\boldsymbol{A}^{\diamondsuit}\boldsymbol{Q}_b^{\diamondsuit}\boldsymbol{A}^{\diamondsuit^T}\right)$ which has exactly the structure of an assembled condensed stiffness matrix. The factorization of such a matrix is expensive in the general case, but we can choose a specific fill-in of matrix $\boldsymbol{Q}_b^{\diamondsuit}$ for the assembled matrix $\left(\boldsymbol{A}^{\diamondsuit}\boldsymbol{Q}_b^{\diamondsuit}\boldsymbol{A}^{\diamondsuit^T}\right)$ to have a block diagonal structure, which makes its handling much cheaper.

Note that an equivalent formulation is possible where the boundary unknowns $\boldsymbol{\mu}_b^{\diamondsuit}$ are replaced by two interface unknowns: one balanced force $\boldsymbol{\gamma}_B$ and one continuous displacement fields $\boldsymbol{v}_A$:

$$\boldsymbol{\mu}_b^{\diamondsuit} = \boldsymbol{B}^{\diamondsuit^T}\boldsymbol{\gamma}_B + \boldsymbol{Q}_b^{\diamondsuit}\boldsymbol{A}^{\diamondsuit^T}\boldsymbol{v}_A \tag{23}$$

This is due to (17) which gives the following decomposition, similar to (5):

$$\begin{aligned}&\forall \boldsymbol{x}_b^{\diamondsuit} \in \mathbb{R}_b^{\diamondsuit}, \exists (\boldsymbol{x}_B, \boldsymbol{x}_A) \in \mathbb{R}^{\Gamma_B} \times \mathbb{R}^{\Gamma_A} \;/\; \boldsymbol{x}_b^{\diamondsuit} = \boldsymbol{Q}_b^{\diamondsuit}\boldsymbol{A}^{\diamondsuit^T}\boldsymbol{x}_A + \boldsymbol{B}^{\diamondsuit^T}\boldsymbol{x}_B\\ &\text{indeed } \begin{cases}\boldsymbol{x}_A = \left(\boldsymbol{A}^{\diamondsuit}\boldsymbol{Q}_b^{\diamondsuit}\boldsymbol{A}^{\diamondsuit^T}\right)^{-1}\boldsymbol{A}^{\diamondsuit}\boldsymbol{x}_b^{\diamondsuit}\\ \boldsymbol{x}_B = \left(\boldsymbol{B}^{\diamondsuit}\boldsymbol{Q}_b^{\diamondsuit^{-1}}\boldsymbol{B}^{\diamondsuit^T}\right)^{+}\boldsymbol{B}^{\diamondsuit}\boldsymbol{Q}_b^{\diamondsuit^{-1}}\boldsymbol{x}_b^{\diamondsuit}\end{cases}\end{aligned} \tag{24}$$

4.4 Note on the existence of nonlinear Schur complements

In previous sections we assumed the existence of nonlinear primal/dual/mixed Schur complement. For a well-posed global mechanical problem, local existence of such subdomain operator is most probable (see for instance [5] for a list of nonlinear frameworks where solutions exist). Yet the formulation, the material and the shape of the subdomain may strongly limit the domain of existence and uniqueness of the Dirichlet/Neumann/Robin problems: typically the large displacements hypothesis may prevent us from using too large compressive loads because of the possibility of buckling, as well as damage may prevent us from using too large tensile loads (see Section 7 for an illustration of these difficulties).

The global existence of the operators can be proved in the case of coercive continuous monotone operators; see for instance [42, 43] for an analysis at the level of the variational formulation and [14] for the analysis of the finite element approximation. Mechanically this framework is associated with positive hardening behaviors and certain contact laws, in small strains [29, 30]. In that context, Dirichlet, Neumann (assuming balanced load) and Robin problems are well-posed: the solution exists, is unique and varies continuously with respect to the load. Thanks to the continuity of the trace, the Schur operators can be defined. Moreover, the Schur complements inherit properties from the global problem (typically monotonicity, coercivity and continuity; see for instance [20] and associated bibliography), which makes the global condensed problems (9,15,22) well-posed.

5 Solution strategy

At this point, assuming the existence of local "Schur-type" nonlinear operators on subdomains, we have obtained one global nonlinear interface problem with an additive structure since it can be written as the assembly of subdomain contributions:

$$\boldsymbol{L}(\boldsymbol{x}) := \sum_s \boldsymbol{L}^{(s)}(\boldsymbol{x}) = 0 \tag{25}$$

We propose to apply a Newton-Raphson procedure to that system. Iteration k consists in solving:

$$\begin{aligned} &\frac{\partial \boldsymbol{L}}{\partial \boldsymbol{x}}(\boldsymbol{x}_k)\overset{\circ}{\boldsymbol{x}}_k = -\boldsymbol{L}(\boldsymbol{x}_k) \quad \text{and} \quad \boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \overset{\circ}{\boldsymbol{x}}_k \\ \text{i.e.} \quad &\left(\sum_s \frac{\partial \boldsymbol{L}^{(s)}}{\partial \boldsymbol{x}}(\boldsymbol{x}_k)\right)\overset{\circ}{\boldsymbol{x}}_k = -\sum_s \boldsymbol{L}^{(s)}(\boldsymbol{x}_k) \quad \text{and} \quad \boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \overset{\circ}{\boldsymbol{x}}_k \end{aligned} \tag{26}$$

We observe that the right-hand side, which corresponds to the evaluation of the residual, is obtained as the assembly of the result of independent solutions to nonlinear problems for each subdomain; whereas the left-hand side corresponds to the construction of the tangent operator as the assembly of subdomains' tangent operators.

The key point of this solution procedure is that the tangent condensed operators can actually be computed: indeed the nonlinear condensed operators are defined as an assembly of the trace (which are linear operations) of the solution to subdomains' problems, the linearization of which leads to the definition of their tangent stiffness matrices. In other words, the tangent of the nonlinear condensed operator (primal, dual or mixed) is the condensation of tangent subdomain operators.

We now explain in more detail the computation of the tangent operator for the primal approach. Let $\boldsymbol{S}_t^{(s)}$ be the tangent operator of the nonlinear Schur complement $\boldsymbol{S}_{nl}^{(s)}$. It can be defined as the unique linear operator satisfying the following relation for any $\overset{\circ}{\boldsymbol{u}}_b^{(s)}$:

$$\overset{\circ}{\boldsymbol{\lambda}}_b^{(s)} := \boldsymbol{S}_{nl}^{(s)}(\boldsymbol{u}_{b_k}^{(s)} + \overset{\circ}{\boldsymbol{u}}_b^{(s)}; \boldsymbol{f}_{ext}^{(s)}) - \boldsymbol{S}_{nl}^{(s)}(\boldsymbol{u}_{b_k}^{(s)}; \boldsymbol{f}_{ext}^{(s)}) = \boldsymbol{S}_{t_k}^{(s)}\overset{\circ}{\boldsymbol{u}}_b^{(s)} + o(\overset{\circ}{\boldsymbol{u}}_b^{(s)})$$

The associated subdomain problems can be written as:

$$\boldsymbol{f}_{int}^{(s)}(\boldsymbol{u}_k^{(s)}) + \boldsymbol{f}_{ext}^{(s)} + \boldsymbol{t}^{(s)^T}\boldsymbol{\lambda}_{b_k}^{(s)} = 0 \text{ with } \boldsymbol{t}^{(s)}\boldsymbol{u}_k^{(s)} = \boldsymbol{u}_{b_k}^{(s)}$$

The differentiation around $\boldsymbol{u}_k^{(s)}$ involves the tangent stiffness matrix $\boldsymbol{K}_{t_k}^{(s)}$:

$$-\boldsymbol{K}_{t_k}^{(s)}\overset{\circ}{\boldsymbol{u}}^{(s)} + \boldsymbol{t}^{(s)^T}\overset{\circ}{\boldsymbol{\lambda}}_b^{(s)} + o\left(\overset{\circ}{\boldsymbol{u}}^{(s)}\right) = 0 \text{ with } \boldsymbol{t}^{(s)}\overset{\circ}{\boldsymbol{u}}^{(s)} = \overset{\circ}{\boldsymbol{u}}_b^{(s)}$$

If we assume the well-posedness of the tangent Dirichlet problem ($\|(\boldsymbol{K}_{t_kii}^{(s)})^{-1}\| < \infty$), then we can condense the previous equation without amplifying the negligible terms and then we obtain that

$$\begin{aligned} \overset{\circ}{\boldsymbol{\lambda}}_b^{(s)} &= \boldsymbol{S}_{t_k}^{(s)}\overset{\circ}{\boldsymbol{u}}_b^{(s)} + o\left(\overset{\circ}{\boldsymbol{u}}_b^{(s)}\right) \\ &= \left(\boldsymbol{K}_{t_kbb}^{(s)} - \boldsymbol{K}_{t_kbi}^{(s)}(\boldsymbol{K}_{t_kii}^{(s)})^{-1}\boldsymbol{K}_{t_kib}^{(s)}\right)\overset{\circ}{\boldsymbol{u}}_b^{(s)} + o\left(\overset{\circ}{\boldsymbol{u}}_b^{(s)} - \boldsymbol{K}_{t_kbi}^{(s)}(\boldsymbol{K}_{t_kii}^{(s)})^{-1}\overset{\circ}{\boldsymbol{u}}_i^{(s)}\right) \end{aligned}$$

Hence the identification between the tangent Schur complement and the Schur complement of the tangent stiffness matrix.

In the following, we derive the important steps of the different strategies.

5.1 Primal formulation

In the primal case, applying the Newton algorithm to equation (9) leads to:

$$\left(\boldsymbol{A}^{\diamondsuit}\boldsymbol{S}_{t_k}^{\diamondsuit}\boldsymbol{A}^{\diamondsuit^T}\right)\mathring{\boldsymbol{u}}_{A_k} = -\boldsymbol{A}^{\diamondsuit}\boldsymbol{S}_{nl}^{\diamondsuit}(\boldsymbol{A}^{\diamondsuit^T}\boldsymbol{u}_{A_k};\boldsymbol{f}_{ext}^{\diamondsuit}) \tag{27}$$

where $\boldsymbol{S}_{t_k}^{(s)} = \left(\frac{\partial \boldsymbol{S}_{nl}^{(s)}}{\partial \boldsymbol{u}_b^{(s)}}(\boldsymbol{u}_{b_k}^{(s)};\boldsymbol{f}_{ext}^{(s)})\right)$ is the subdomain tangent primal Schur complement as given in (8).

First, let us focus on the right-hand side. It corresponds to the evaluation of a nonlinear Dirichlet problem for each subdomain with imposed interface displacements $\boldsymbol{u}_{A_k}$:

$$\boldsymbol{u}_k^{\diamondsuit} \text{ such that } \begin{cases} \left(\boldsymbol{f}_{int}^{\diamondsuit}(\boldsymbol{u}_k^{\diamondsuit}) + \boldsymbol{f}_{ext}^{\diamondsuit}\right)_i = 0 \\ \boldsymbol{t}^{\diamondsuit}\boldsymbol{u}_k^{\diamondsuit} = \boldsymbol{A}^{\diamondsuit^T}\boldsymbol{u}_{A_k} \end{cases}, \quad \boldsymbol{\lambda}_{b_k}^{\diamondsuit} := -\left(\boldsymbol{f}_{int}^{\diamondsuit}(\boldsymbol{u}_k^{\diamondsuit}) + \boldsymbol{f}_{ext}^{\diamondsuit}\right)_b \tag{28}$$

From the parallel solution of these systems, we obtain the internal displacements $\boldsymbol{u}_k^{\diamondsuit}$ and the reactions $\boldsymbol{\lambda}_{b_k}^{\diamondsuit}$ whose lack of balance $\boldsymbol{A}^{\diamondsuit}\boldsymbol{\lambda}_{b_k}^{\diamondsuit} = \boldsymbol{A}^{\diamondsuit}\boldsymbol{S}_{nl}^{\diamondsuit}(\boldsymbol{A}^{\diamondsuit^T}\boldsymbol{u}_{A_k};\boldsymbol{f}_{ext}^{\diamondsuit})$ is the right-hand side of the tangent system.

The left-hand side of (27) is the assembly of the subdomain's tangent primal Schur complements, that is to say a classical primal domain decomposition formulation of the tangent problem.

Thus, after an arbitrary initialization, the straightforward solution consists in repeating the following steps: (i) solving independent nonlinear Dirichlet systems for each subdomain, (ii) computing the lack of balance at the interface, and (iii) solving the global tangent interface problem by the application of the BDD algorithm [33, 32].

5.2 Dual formulation

In the dual case, applying Newton algorithm to equation (15) leads to:

$$\begin{cases} \left(\boldsymbol{B}^{\diamondsuit}\boldsymbol{F}_{t_k}^{\diamondsuit}\boldsymbol{B}^{\diamondsuit^T}\right)\mathring{\boldsymbol{\lambda}}_{B_k} + \boldsymbol{B}^{\diamondsuit}\boldsymbol{R}_b^{\diamondsuit}\mathring{\boldsymbol{\alpha}}_k^{\diamondsuit} = -\boldsymbol{B}^{\diamondsuit}\left(\boldsymbol{F}_{nl}^{\diamondsuit}(\boldsymbol{B}^{\diamondsuit^T}\boldsymbol{\lambda}_{B_k};\boldsymbol{f}_{ext}^{\diamondsuit}) + \boldsymbol{R}_b^{\diamondsuit}\boldsymbol{\alpha}_k^{\diamondsuit}\right) \\ \boldsymbol{R}_b^{\diamondsuit^T}\boldsymbol{B}^{\diamondsuit^T}\mathring{\boldsymbol{\lambda}}_{B_k} = 0 \end{cases} \tag{29}$$

where $\boldsymbol{F}_{t_k}^{(s)} = \left(\frac{\partial \boldsymbol{F}_{nl}^{(s)}}{\partial \boldsymbol{\lambda}_b^{(s)}}(\boldsymbol{\lambda}_{b_k}^{(s)};\boldsymbol{f}_{ext}^{(s)})\right)$ is the subdomains' tangent dual Schur complement in configuration k as given in (13).

As usually done in the FETI method, we introduce a projector $\boldsymbol{P}_B$ on $\mathrm{Ker}(\boldsymbol{R}_b^{\diamondsuit^T}\boldsymbol{B}^{\diamondsuit^T})$, so that we can seek $\boldsymbol{P}_B\mathring{\boldsymbol{\lambda}}_{B_k}$ instead of $\mathring{\boldsymbol{\lambda}}_{B_k}$, solution to:

$$\boldsymbol{P}_B^T\left(\boldsymbol{B}^{\diamondsuit}\boldsymbol{F}_{t_k}^{\diamondsuit}\boldsymbol{B}^{\diamondsuit^T}\right)\boldsymbol{P}_B\mathring{\boldsymbol{\lambda}}_{B_k} = -\boldsymbol{P}_B^T\boldsymbol{B}^{\diamondsuit}\boldsymbol{F}_{nl}^{\diamondsuit}(\boldsymbol{B}^{\diamondsuit^T}\boldsymbol{\lambda}_{B_k};\boldsymbol{f}_{ext}^{\diamondsuit}) \tag{30}$$

and the contribution of the rigid body motions is sought after $\boldsymbol{\lambda}_B$ is determined.

The right-hand side of (29) corresponds to the evaluation of nonlinear Neumann problems for each subdomain with imposed interface reaction $\boldsymbol{\lambda}_{B_k}$:

$$\boldsymbol{f}_{int}^{\diamondsuit}(\boldsymbol{u}_k^{\diamondsuit}) + \boldsymbol{f}_{ext}^{\diamondsuit} + \boldsymbol{t}^{\diamondsuit^T}\boldsymbol{B}^{\diamondsuit^T}\boldsymbol{\lambda}_{B_k} = 0 \tag{31}$$

From the parallel solution of these systems, we obtain the displacements $\boldsymbol{u}_k^{\diamondsuit}$ whose interface gap $\boldsymbol{P}_B^T\boldsymbol{B}^{\diamondsuit}\boldsymbol{t}^{\diamondsuit}\boldsymbol{u}_k^{\diamondsuit} = \boldsymbol{P}_B^T\boldsymbol{B}^{\diamondsuit}\boldsymbol{F}_{nl}^{\diamondsuit}(\boldsymbol{B}^{\diamondsuit^T}\boldsymbol{\lambda}_{B_k};\boldsymbol{f}_{ext}^{\diamondsuit})$ is the right-hand side of the tangent system.

The tangent matrix is the assembly of local tangent dual Schur complements computed at the value $\boldsymbol{u}_k^{\diamondsuit}$ obtained when computing the right-hand side. Then the first term of (29) is the assembly of the subdomain's tangent dual Schur complements, that is to say a classical dual domain decomposition formulation of the tangent problem.

In order for system (31) to be well posed, the rigid-body admissibility condition (11) has to be satisfied. This is realized by initializing $\boldsymbol{\lambda}_B$ by an admissible reaction $\boldsymbol{\lambda}_{B_0}$ which verifies:

$$\boldsymbol{R}^{\diamondsuit^T}\left(\boldsymbol{f}_{ext}^{\diamondsuit} + \boldsymbol{t}^{\diamondsuit^T}\boldsymbol{B}^{\diamondsuit^T}\boldsymbol{\lambda}_{B_0}\right) = 0 \tag{32}$$

Indeed, since increments $\mathring{\boldsymbol{\lambda}}_{B_k}$ satisfy a zero-admissibility condition (29), and because the admissibility is a linear condition thanks to the small perturbation hypothesis, any coming $\boldsymbol{\lambda}_{B_k}$ will satisfy the admissibility condition. Finding an initial value $\boldsymbol{\lambda}_{B_0}$ corresponds to the initialization of the FETI method.

Thus, the straightforward solution consists in solving a FETI coarse grid problem to initialize $\boldsymbol{\lambda}_{B_0}$, then repeating the following steps: (i) solving independent nonlinear Neumann systems for each subdomain, (ii) computing the displacement gap at the interface, and (iii) solving the global tangent interface problem by the application of FETI algorithm [12]. Note that the initialization can be improved by a full solution of the initial FETI system.

5.3 Mixed formulation

In the mixed case applying Newton algorithm to equation (22) leads to:

$$\left(\boldsymbol{A}^{\diamondsuit^T}\left(\boldsymbol{A}^{\diamondsuit}\boldsymbol{Q}_b^{\diamondsuit}\boldsymbol{A}^{\diamondsuit^T}\right)^{-1}\boldsymbol{A}^{\diamondsuit}-\boldsymbol{M}_{t_k}^{\diamondsuit}\right)\mathring{\boldsymbol{\mu}}_{b_k}^{\diamondsuit}$$
$$=\boldsymbol{M}_{nl}^{\diamondsuit}(\boldsymbol{\mu}_{b_k}^{\diamondsuit};\boldsymbol{f}_{ext}^{\diamondsuit},\boldsymbol{Q}_b^{\diamondsuit})-\boldsymbol{A}^{\diamondsuit^T}\left(\boldsymbol{A}^{\diamondsuit}\boldsymbol{Q}_b^{\diamondsuit}\boldsymbol{A}^{\diamondsuit^T}\right)^{-1}\boldsymbol{A}^{\diamondsuit}\boldsymbol{\mu}_{b_k}^{\diamondsuit} \quad (33)$$

where $\boldsymbol{M}_{t_k}^{(s)}=\left(\frac{\partial \boldsymbol{M}_{nl}^{(s)}}{\partial \boldsymbol{\mu}_b^{(s)}}(\boldsymbol{\mu}_{b_k}^{(s)};\boldsymbol{f}_{ext}^{(s)},\boldsymbol{Q}_b^{(s)})\right)$ is the subdomains' tangent mixed Schur complement in configuration k as given in (21).

The first term of the right-hand side of (33) corresponds to the evaluation of nonlinear Robin problems for each subdomain with interface impedance $\boldsymbol{Q}_b^{\diamondsuit}$ and imposed interface reaction $\boldsymbol{\mu}_{b_k}^{\diamondsuit}$:

$$\boldsymbol{f}_{int}^{\diamondsuit}(\boldsymbol{u}_k^{\diamondsuit})-\boldsymbol{t}^{\diamondsuit^T}\boldsymbol{Q}_b^{\diamondsuit}\boldsymbol{t}^{\diamondsuit}\boldsymbol{u}_k^{\diamondsuit}+\boldsymbol{t}^{\diamondsuit^T}\boldsymbol{\mu}_{b_k}^{\diamondsuit}+\boldsymbol{f}_{ext}^{\diamondsuit}=0 \quad (34)$$

from the parallel solution of these systems, we obtain the displacement $\boldsymbol{u}_k^{\diamondsuit}$ from which we deduce the mixed residual $\boldsymbol{b}_{m_k}^{\diamondsuit}=\boldsymbol{u}_{b_k}^{\diamondsuit}-\boldsymbol{A}^{\diamondsuit^T}\left(\boldsymbol{A}^{\diamondsuit}\boldsymbol{Q}_b^{\diamondsuit}\boldsymbol{A}^{\diamondsuit^T}\right)^{-1}\boldsymbol{A}^{\diamondsuit}\boldsymbol{\mu}_{b_k}^{\diamondsuit}$ which is the right-hand side of the tangent system.

The tangent operator is obtained by mixed condensation of the tangent stiffness matrix computed at the value $\boldsymbol{u}_k^{\diamondsuit}$ obtained when evaluating the right-hand side. Then the first term of (33) is a classical mixed domain decomposition formulation of the tangent problem.

Thus after an arbitrary initialization, the straightforward solution consists in repeating the following steps: (i) solving independent nonlinear Robin systems for each subdomain, (ii) computing the mixed residual, and (iii) solving the global tangent interface problem by the application of FETI-2LM algorithm [41].

5.4 Swapping the linear solvers

In any case, the solution procedure alternates independent nonlinear solutions for each subdomain with different boundary conditions, and global tangent solutions similar to a domain decomposition method applied to linear problems. As we will show later, the choice of the boundary condition strongly impacts the nonlinear subdomains' solving. As the tangent operator is always associated with the tangent stiffness matrix of subdomains, it is possible to choose any formulation for the linear problem.

Indeed, starting from (8,13,21) we have the following classical relationships [17]:

$$\begin{aligned}\boldsymbol{F}_t^{\diamondsuit}&=\boldsymbol{S}_t^{\diamondsuit^+} & \boldsymbol{M}_t^{\diamondsuit}&=\left(\boldsymbol{S}_t^{\diamondsuit}+\boldsymbol{Q}_b^{\diamondsuit}\right)^{-1}\\ \boldsymbol{b}_d^{\diamondsuit}&=\boldsymbol{F}_t^{\diamondsuit}\boldsymbol{b}_p^{\diamondsuit} & \boldsymbol{b}_m^{\diamondsuit}&=\boldsymbol{M}_t^{\diamondsuit}\boldsymbol{b}_p^{\diamondsuit}\end{aligned} \quad (35)$$

This implies that even if a formulation results in a specific residual after the subdomains' nonlinear solutions, any linear solver can be employed. Typically, the mixed formulation has interesting capabilities for nonlinear problems, but it is not very classical for linear system for which primal (BDD) or dual (FETI) formulations are more standard and embed powerful preconditioners. After computing $\boldsymbol{M}_{nl}^{\diamondsuit}(\boldsymbol{\mu}_{b_k}^{\diamondsuit};\boldsymbol{f}_{ext}^{\diamondsuit},\boldsymbol{Q}_b^{\diamondsuit})$, one can compute $\boldsymbol{b}_{m_k}^{\diamondsuit}=\boldsymbol{M}_{nl}^{\diamondsuit}(\boldsymbol{\mu}_{b_k}^{\diamondsuit};\boldsymbol{f}_{ext}^{\diamondsuit},\boldsymbol{Q}_b^{\diamondsuit})-\boldsymbol{A}^{\diamondsuit^T}\left(\boldsymbol{A}^{\diamondsuit}\boldsymbol{Q}_b^{\diamondsuit}\boldsymbol{A}^{\diamondsuit^T}\right)^{-1}\boldsymbol{A}^{\diamondsuit}\boldsymbol{\mu}_{b_k}^{\diamondsuit}$, deduce the equivalent primal right-hand side $\boldsymbol{b}_{p_k}^{\diamondsuit}=\left(\boldsymbol{S}_{t_k}^{\diamondsuit}+\boldsymbol{Q}_b^{\diamondsuit}\right)\boldsymbol{b}_{m_k}^{\diamondsuit}$ and use the associated interface lack of balance $\boldsymbol{A}^{\diamondsuit}\boldsymbol{b}_{p_k}^{\diamondsuit}$ as input to the tangent BDD solver. This property can directly be observed by introducing (23) in the mixed tangent system (33), since after premultiplying by $(\boldsymbol{Q}_b^{\diamondsuit}+\boldsymbol{S}_{t_k}^{\diamondsuit})$ we obtain:

$$\begin{aligned}\boldsymbol{S}_{t_k}^{\diamondsuit}\boldsymbol{A}^{\diamondsuit^T}\mathring{\boldsymbol{v}}_{A_k}-\boldsymbol{B}^{\diamondsuit^T}\mathring{\boldsymbol{\gamma}}_{B_k}&=(\boldsymbol{Q}_b^{\diamondsuit}+\boldsymbol{S}_{t_k}^{\diamondsuit})\left(\boldsymbol{M}_{nl}^{\diamondsuit}(\boldsymbol{\mu}_{b_k}^{\diamondsuit};\boldsymbol{f}_{ext}^{\diamondsuit},\boldsymbol{Q}_b^{\diamondsuit})-\boldsymbol{A}^{\diamondsuit^T}\boldsymbol{v}_{A_k}\right)\\&=(\boldsymbol{Q}_b^{\diamondsuit}+\boldsymbol{S}_{t_k}^{\diamondsuit})\boldsymbol{b}_{m_k}^{\diamondsuit}=\boldsymbol{b}_{p_k}^{\diamondsuit}\end{aligned} \quad (36)$$

which leads to the system set in terms of $\mathring{\boldsymbol{v}}_{A_k}$:

$$\left(\boldsymbol{A}^{\diamondsuit}\boldsymbol{S}_{t_k}^{\diamondsuit}\boldsymbol{A}^{\diamondsuit^T}\right)\mathring{\boldsymbol{v}}_{A_k}=\boldsymbol{A}^{\diamondsuit}\boldsymbol{b}_{p_k}^{\diamondsuit} \quad (37)$$

The following quantities can be deduced (even from an inexact solution):

$$\begin{aligned}\mathring{\boldsymbol{\mu}}_{b_k}^{\diamondsuit}&=\boldsymbol{S}_{t_k}^{\diamondsuit}\boldsymbol{A}^{\diamondsuit^T}\mathring{\boldsymbol{v}}_{A_k}-\boldsymbol{b}_{p_k}^{\diamondsuit}\\ \mathring{\boldsymbol{u}}_k^{\diamondsuit}&=(\boldsymbol{K}_{t_k}^{\diamondsuit}+\boldsymbol{t}^{\diamondsuit^T}\boldsymbol{Q}_b^{\diamondsuit}\boldsymbol{t}^{\diamondsuit})^{-1}\boldsymbol{t}^{\diamondsuit^T}\left(\boldsymbol{b}_{p_k}^{\diamondsuit}+\mathring{\boldsymbol{\mu}}_{b_k}^{\diamondsuit}\right)\\ \mathring{\boldsymbol{u}}_{b_k}^{\diamondsuit}&=\boldsymbol{t}^{\diamondsuit}\mathring{\boldsymbol{u}}_k^{\diamondsuit}\\ \mathring{\boldsymbol{\lambda}}_{b_k}^{\diamondsuit}&=\boldsymbol{S}_{t_k}^{\diamondsuit}\mathring{\boldsymbol{u}}_{b_k}^{\diamondsuit}-\boldsymbol{b}_{p_k}=\mathring{\boldsymbol{\mu}}_{b_k}^{\diamondsuit}-\boldsymbol{Q}_b^{\diamondsuit}\mathring{\boldsymbol{u}}_{b_k}^{\diamondsuit}\end{aligned} \quad (38)$$

Note that $\mathring{\boldsymbol{\gamma}}_{B_k}$ plays no role in the algorithm but it could also be post-processed:

$$\mathring{\boldsymbol{\gamma}}_{B_k} = \boldsymbol{B}^{\diamondsuit^{T+}} \left(\mathring{\boldsymbol{\mu}}_{b_k}^{\diamondsuit} - \boldsymbol{Q}_b^{\diamondsuit} \boldsymbol{A}^{\diamondsuit^T} \mathring{\boldsymbol{v}}_{A_k} \right) \tag{39}$$

where $\boldsymbol{B}^{\diamondsuit^{T+}}$ is a scaled assembling operator commonly employed in the preconditioning step of the FETI method [40, 28]; because of the space splitting properties (24), $\mathring{\boldsymbol{\gamma}}_{B_k}$ does not depend on the choice of the scaling. Of course, if a dual formulation had been preferred then $\mathring{\boldsymbol{\gamma}}_{B_k}$ would have been computed by a FETI-like system (left-multiply (36) by $\boldsymbol{B}^{\diamondsuit}\boldsymbol{F}_t^{\diamondsuit}$).

5.5 Typical algorithm

Algorithm 1 sums up the main steps of the method with the mixed nonlinear local problems and primal global solver. For simplicity reasons, only one increment was considered.

As can be seen in this algorithm, besides classical global convergence criterion ε_{NG} (as in a standard Newton approach), two precision thresholds are used: the local nonlinear thresholds $\varepsilon_{NL}^{\diamondsuit}$ (associated with the Newton processes carried out independently on subdomains) and the global linear threshold of the domain decomposition (Krylov) solver ε_K (here BDD). The convergence criteria are discussed in more detail in the following section.

Other parameters can be the initializations of the various iterative solvers and, for the mixed approach, the choice of the impedance matrices $\boldsymbol{Q}_b^{\diamondsuit}$.

Algorithm 1: Mixed nonlinear approach with BDD tangent solver

Define:
$\boldsymbol{r}_{nl}^{m\diamondsuit}(\boldsymbol{u}^{\diamondsuit}, \boldsymbol{\mu}_b^{\diamondsuit}) = \boldsymbol{f}_{int}^{\diamondsuit}(\boldsymbol{u}^{\diamondsuit}) - \boldsymbol{t}^{\diamondsuit^T}\boldsymbol{Q}_b^{\diamondsuit}\boldsymbol{t}^{\diamondsuit}\boldsymbol{u}^{\diamondsuit} + \boldsymbol{t}^{\diamondsuit^T}\boldsymbol{\mu}_b{}^{\diamondsuit} + \boldsymbol{f}_{ext}^{\diamondsuit}$
Initialization:
$(\boldsymbol{u}_0^{\diamondsuit}, \boldsymbol{\lambda}_{b0}^{\diamondsuit})$ such that $\boldsymbol{B}^{\diamondsuit}\boldsymbol{t}^{\diamondsuit}\boldsymbol{u}_0^{\diamondsuit} = 0$ and $\boldsymbol{A}^{\diamondsuit}\boldsymbol{\lambda}_{b0}^{\diamondsuit} = 0$
Set $k = 0$
Define $\boldsymbol{\mu}_{bk}^{\diamondsuit} = \boldsymbol{\lambda}_{bk}^{\diamondsuit} + \boldsymbol{Q}_b^{\diamondsuit}\boldsymbol{t}^{\diamondsuit}\boldsymbol{u}_k^{\diamondsuit}$
while $\|\boldsymbol{r}_{nl}^{m\diamondsuit}(\boldsymbol{u}_k^{\diamondsuit}, \boldsymbol{\mu}_{bk}^{\diamondsuit})\| + \|\boldsymbol{B}^{\diamondsuit}\boldsymbol{t}^{\diamondsuit}\boldsymbol{u}^{\diamondsuit}\|_B > \varepsilon_{NG}$ **do**
  Local nonlinear step:
  Set $\boldsymbol{u}_{k,0}^{\diamondsuit} = \boldsymbol{u}_k^{\diamondsuit}$ and $j = 0$
  while $\|\boldsymbol{r}_{nl}^{m\diamondsuit}(\boldsymbol{u}_{k,j}^{\diamondsuit}, \boldsymbol{\mu}_{bk}^{\diamondsuit})\| > \varepsilon_{NL}^{\diamondsuit}$ **do**
    $\boldsymbol{u}_{k,j+1}^{\diamondsuit} = \boldsymbol{u}_{k,j}^{\diamondsuit} - \left(\boldsymbol{K}_{t_{k,j}}^{\diamondsuit} + \boldsymbol{t}^{\diamondsuit^T}\boldsymbol{Q}_b^{\diamondsuit}\boldsymbol{t}^{\diamondsuit}\right)^{-1} \boldsymbol{r}_{nl}^{m\diamondsuit}(\boldsymbol{u}_{k,j}^{\diamondsuit}, \boldsymbol{\mu}_{bk}^{\diamondsuit})$
    Set $j = j + 1$
  end
  Linear right-hand side:
  $\boldsymbol{b}_{m_k}^{\diamondsuit} = \boldsymbol{A}^{\diamondsuit^T}\left(\boldsymbol{A}^{\diamondsuit}\boldsymbol{Q}_b^{\diamondsuit}\boldsymbol{A}^{\diamondsuit^T}\right)^{-1}\boldsymbol{A}^{\diamondsuit}\boldsymbol{\mu}_{bk}^{\diamondsuit} - \boldsymbol{t}^{\diamondsuit}\boldsymbol{u}_{k,j}^{\diamondsuit}$
  $\boldsymbol{b}_{p_k}^{\diamondsuit} = (\boldsymbol{S}_{t_{k,j}}^{\diamondsuit} + \boldsymbol{Q}_b^{\diamondsuit})\boldsymbol{b}_{m_k}^{\diamondsuit}$
  Global linear step:
  Set $\mathring{\boldsymbol{v}}_{A_k}^{0} = 0$ and $i = 0$
  while $\|\boldsymbol{b}_{p_k}^{\diamondsuit} - \left(\boldsymbol{A}^{\diamondsuit}\,\boldsymbol{S}_{t_{k,j}}^{\diamondsuit}\boldsymbol{A}^{\diamondsuit^T}\right)\mathring{\boldsymbol{v}}_{A_k}^{i}\| > \varepsilon_K$ **do**
    Make BDD iterations (index i)
  end
  Set $\boldsymbol{u}_{k+1}^{\diamondsuit} = \boldsymbol{u}_k^{\diamondsuit} + \mathring{\boldsymbol{u}}_k^{i\diamondsuit}$ and $\boldsymbol{\lambda}_{b\,k+1}^{\diamondsuit} = \boldsymbol{\lambda}_{b_k}^{\diamondsuit} + \mathring{\boldsymbol{\lambda}}_{b_k}^{i\diamondsuit}$ using (38)
  Set $k = k + 1$
end

6 Error analysis

The aim of this section is to further analyze the effect of the different thresholds and the way they are connected. When possible, we will make reference to the global residual $\boldsymbol{r}$ which is defined as:

$$\boldsymbol{f}_{int}(\boldsymbol{u}) + \boldsymbol{f}_{ext} = \boldsymbol{r} \tag{40}$$

The objective of the solver is that $\|\boldsymbol{r}\| < \varepsilon_{NG}\|\boldsymbol{f}_{ext}\|$ (global nonlinear criterion).

We first show how the convergence criteria for the Local Newton solvers $\varepsilon_{NL}^{\diamondsuit}$ and for the Krylov interface solver ε_K are related for the different approaches. Then we interpret the whole process as an inexact Newton solver [7] and

we consider adapting at each step the criteria ε_K and $\varepsilon_{NL}^{\diamondsuit}$ to the current error status $\|\boldsymbol{r}\|$. ε_K and $\varepsilon_{NL}^{\diamondsuit}$ are interpreted as forcing terms which need to be close to the objective ε_{NG} only when global convergence is almost reached whereas they should be relaxed at the beginning of the process in order to avoid oversolving (see Section 6.4).

6.1 Primal approach

6.1.1 Local Newton solvers

The inner (local) Newton loops are associated with Dirichlet problems. Assuming these conditions are exactly taken into account, the convergence is controlled by the internal node residue:

$$\boldsymbol{f}_{int\,i}^{\diamondsuit}(\boldsymbol{u}^{\diamondsuit}) + \boldsymbol{f}_{ext\,i}^{\diamondsuit} = \boldsymbol{r}_i^{\diamondsuit} \text{ with } \|\boldsymbol{r}_i^{\diamondsuit}\| < \varepsilon_{NL}^{\diamondsuit}\|\boldsymbol{f}_{ext\,i}^{\diamondsuit}\| \tag{41}$$

From the $\boldsymbol{u}^{\diamondsuit}$ computed above, the reactions are defined by the following relation:

$$\boldsymbol{\lambda}_b^{\diamondsuit} := -\left(\boldsymbol{f}_{int\,b}^{\diamondsuit}(\boldsymbol{u}^{\diamondsuit}) + \boldsymbol{f}_{ext\,b}^{\diamondsuit}\right) \tag{42}$$

6.1.2 Link to the global error

In the primal case, the displacement search space is the same as in the non-substructured case (displacements are continuous across the interface). This simplifies the error analysis. Indeed, we directly have:

$$\begin{aligned}\|\boldsymbol{f}_{int}(\boldsymbol{u}) + \boldsymbol{f}_{ext}\| &\leqslant \|\boldsymbol{f}_{int_i}^{\diamondsuit}(\boldsymbol{u}^{\diamondsuit}) + \boldsymbol{f}_{ext_i}^{\diamondsuit}\| + \|\boldsymbol{A}^{\diamondsuit}\left(\boldsymbol{f}_{int_b}^{\diamondsuit}(\boldsymbol{u}^{\diamondsuit}) + \boldsymbol{f}_{ext_b}^{\diamondsuit}\right)\| \\ &\leqslant \|\varepsilon_{NL}^{\diamondsuit}\|\|\boldsymbol{f}_{ext_i}^{\diamondsuit}\| + \|\boldsymbol{A}^{\diamondsuit}\boldsymbol{\lambda}_b^{\diamondsuit}\|\end{aligned} \tag{43}$$

This means that global convergence occurs when both the local Newton criteria and the resulting lack of balance at the interface (the initial residual of the tangent interface solver) are small.

6.1.3 Tangent interface solver

From the point of view of the nonlinear Schur complement, the computation of the reactions suffers from the error made on internal displacements, if $\boldsymbol{S}_{nl}^{\diamondsuit}(\boldsymbol{A}^{\diamondsuit^T}\boldsymbol{u}_{A_k}; \boldsymbol{f}_{ext}^{\diamondsuit})$ denotes the exact reaction (corresponding to $\boldsymbol{r}_i^{\diamondsuit} = 0$), we write:

$$\boldsymbol{\lambda}_b^{\diamondsuit} = \boldsymbol{S}_{nl}^{\diamondsuit}(\boldsymbol{A}^{\diamondsuit^T}\boldsymbol{u}_{A_k}; \boldsymbol{f}_{ext}^{\diamondsuit}) + \boldsymbol{r}_b^{\diamondsuit} \tag{44}$$

In the linear case, the internal residual would propagate on the boundary according to the relation $\boldsymbol{r}_b^{\diamondsuit} = -\boldsymbol{K}_{bi}^{\diamondsuit}\boldsymbol{K}_{ii}^{\diamondsuit^{-1}}\boldsymbol{r}_i^{\diamondsuit}$, so that we can assume that the error on the reaction is of the same order of the inner solver error $\|\boldsymbol{r}_b^{\diamondsuit}\| \simeq \|\boldsymbol{r}_i^{\diamondsuit}\|$.

We then can study the error made on the tangent solution. The computed vector $\mathring{\boldsymbol{u}}_{A_k}$ satisfies the equation:

$$\left(\boldsymbol{A}^{\diamondsuit}\boldsymbol{S}_{t_k}^{\diamondsuit}\boldsymbol{A}^{\diamondsuit^T}\right)\mathring{\boldsymbol{u}}_{A_k} = -\boldsymbol{A}^{\diamondsuit}\boldsymbol{S}_{nl}^{\diamondsuit}(\boldsymbol{A}^{\diamondsuit^T}\boldsymbol{u}_{A_k}; \boldsymbol{f}_{ext}^{\diamondsuit}) - \boldsymbol{A}^{\diamondsuit}\boldsymbol{r}_b^{\diamondsuit} + \boldsymbol{r}_A^L \tag{45}$$

where $\|\boldsymbol{r}_A^L\| < \varepsilon_K\|\boldsymbol{r}_{A_0}^L\|$ controls the error of the linear (Krylov) solver. Since $\|\boldsymbol{A}^{\diamondsuit}\| = O(1)$, we see that the error of the local Newton solver and the global interface iterative solver should be of the same order of magnitude.

6.2 Dual approach

6.2.1 Local Newton solvers

The inner (local) Newton loops are associated with Neumann problems, and the convergence is controlled by the following residue:

$$\boldsymbol{f}_{int}^{\diamondsuit}(\boldsymbol{u}^{\diamondsuit}) + \boldsymbol{f}_{ext}^{\diamondsuit} + \boldsymbol{t}^{\diamondsuit^T}\boldsymbol{B}^{\diamondsuit^T}\boldsymbol{\lambda}_B = \boldsymbol{r}^{\diamondsuit} \text{ with } \|\boldsymbol{r}^{\diamondsuit}\| < \varepsilon_{NL}^{\diamondsuit}\|\boldsymbol{f}_{ext}^{\diamondsuit}\| \tag{46}$$

The output from these computations is the interface displacement. Unfortunately there is no direct control on the associated error $\boldsymbol{\delta}_b^{\diamondsuit}$. A good estimation is provided by the last correction brought to $\boldsymbol{u}_b^{\diamondsuit}$ by the inner Newton loop.

6.2.2 Link to the global error

The difficulty with the dual approach is that the (broken) search space is larger than the original space (where displacements are continuous). In the linear case, a costless processing enables us to obtain a continuous displacement [34]. Such a strategy is not realistic for nonlinear problems.

One possibility is to define an error extended to the broken space (e_{bs}):

$$e_{bs}^2 = \|\boldsymbol{f}_{int}^{\diamondsuit}(\boldsymbol{u}^{\diamondsuit}) + \boldsymbol{f}_{ext}^{\diamondsuit} + \boldsymbol{t}^{\diamondsuit^T}\boldsymbol{B}^{\diamondsuit^T}\boldsymbol{\lambda}_B\|^2 + \|\boldsymbol{B}^{\diamondsuit}\boldsymbol{t}^{\diamondsuit}\boldsymbol{u}^{\diamondsuit}\|_B^2 \tag{47}$$

where $\|\|_B$ is a well chosen norm for the interface discontinuity. We recover an expression where the inner criterion is cumulated with the initial residual of the tangent interface solver.

6.2.3 Tangent interface solver

The tangent interface solution can be written as:

$$\boldsymbol{P}_B^T \left(\boldsymbol{B}^{\diamond} \boldsymbol{F}_{t_k}^{\diamond} \boldsymbol{B}^{\diamond^T} \right) \boldsymbol{P}_B \mathring{\boldsymbol{\lambda}}_{B_k} = -\boldsymbol{P}_B^T \left(\boldsymbol{B}^{\diamond} \boldsymbol{F}_{nl}^{\diamond}(\boldsymbol{B}^{\diamond^T} \boldsymbol{\lambda}_{B_k}; \boldsymbol{f}_{ext}^{\diamond}) + \boldsymbol{B}^{\diamond} \boldsymbol{\delta}_b^{\diamond} \right) + \boldsymbol{r}_B^L \tag{48}$$

where $\|\boldsymbol{r}_B^L\| < \varepsilon_K \|\boldsymbol{r}_{B_0}^L\|$ controls the error of the linear solver. Since $\|\boldsymbol{B}^{\diamond}\| = O(1)$, we see that the forward error of the local Newton solver and the backward error of the global interface iterative solver should be of the same order of magnitude.

6.3 Mixed approach

6.3.1 Local Newton solvers

The inner (local) Newton loops are associated with Robin problems. The convergence is controlled by the following residue:

$$\boldsymbol{f}_{int}^{\diamond}(\boldsymbol{u}^{\diamond}) - \boldsymbol{t}^{\diamond^T} \boldsymbol{Q}_b^{\diamond} \boldsymbol{t}^{\diamond} \boldsymbol{u}^{\diamond} + \boldsymbol{t}^{\diamond^T} \boldsymbol{\mu}_b{}^{\diamond} + \boldsymbol{f}_{ext}^{\diamond} = \boldsymbol{r}^{\diamond} \text{ with } \|\boldsymbol{r}^{\diamond}\| < \varepsilon_{NL}^{\diamond} \|\boldsymbol{f}_{ext}^{\diamond}\| \tag{49}$$

As in the dual approach, the output from these computations is the interface displacement, of which the error $\boldsymbol{\delta}_b^{\diamond}$ can be estimated by the last correction brought to $\boldsymbol{u}_b^{\diamond}$ by the inner Newton loop.

6.3.2 Link to the global error

As in the dual case, the search space is a broken space. This implies the use of an extended norm to evaluate the global error: not only the displacements are not continuous at the interface but neither are the reactions balanced. With reaction $\boldsymbol{\lambda}_b^{\diamond}$ defined by (42), we see that the trace of the local residue can be written as:

$$\boldsymbol{r}_b^{\diamond} = \boldsymbol{Q}_b^{\diamond} \left(\boldsymbol{A}^{\diamond^T} \boldsymbol{v}_A - \boldsymbol{t}^{\diamond} \boldsymbol{u}^{\diamond} \right) + \left(\boldsymbol{B}^{\diamond^T} \boldsymbol{\gamma}_B - \boldsymbol{\lambda}_b^{\diamond} \right) \tag{50}$$

This relation links the two interface errors, which means that the dual criterion (47) is sufficient to monitor the convergence: if both $\|\boldsymbol{r}_b^{\diamond}\|$ and $\|\boldsymbol{B}^{\diamond} \boldsymbol{t}^{\diamond} \boldsymbol{u}^{\diamond}\|_B$ are small, then so must $\|\boldsymbol{A}^{\diamond} \boldsymbol{\lambda}_b^{\diamond}\|$ be.

6.3.3 Tangent interface solver

The tangent interface solution can be written as (in primal form):

$$\left(\boldsymbol{A}^{\diamond} \boldsymbol{S}_{t_k}^{\diamond} \boldsymbol{A}^{\diamond^T} \right) \mathring{\boldsymbol{v}}_{A_k} = \boldsymbol{A}^{\diamond} (\boldsymbol{Q}_b^{\diamond} + \boldsymbol{S}_{t_k}^{\diamond}) \left(\boldsymbol{u}_{b_k}^{\diamond} + \boldsymbol{\delta}_b^{\diamond} - \boldsymbol{A}^{\diamond^T} \boldsymbol{v}_{A_k} \right) + \boldsymbol{r}_A^L \tag{51}$$

where $\|\boldsymbol{r}_A^L\| < \varepsilon_K \|\boldsymbol{r}_{A_0}^L\|$ controls the error of the linear solver. From the definition of $\boldsymbol{\delta}_b^{\diamond}$, we can expect $\|\boldsymbol{A}^{\diamond}(\boldsymbol{Q}_b^{\diamond} + \boldsymbol{S}_{t_k}^{\diamond}) \boldsymbol{\delta}_b^{\diamond}\| \simeq \|\boldsymbol{A}^{\diamond} \boldsymbol{t}^{\diamond} \boldsymbol{r}^{\diamond}\|$ which means that the convergence criterion of the Krylov solver must be of the order of magnitude of the local Newton solvers' criteria.

6.4 Tuning of criteria

The previous subsections established connections between the convergence criteria of the local Newton solvers and of the global interface Krylov solver. We can now analyze the algorithm in the framework of inexact Newton methods [7]. From previous analysis, and using the notations of (26), we solve (25) by the sequence:

$$\begin{aligned} \frac{\partial \boldsymbol{L}}{\partial \boldsymbol{x}}(\boldsymbol{x}_k) \mathring{\boldsymbol{x}}_k &= -\boldsymbol{L}(\boldsymbol{x}_k) + \boldsymbol{r}_k, \text{ with } \|\boldsymbol{r}_k\| \leqslant \varepsilon_k \|\boldsymbol{L}(\boldsymbol{x}_k)\| \\ \boldsymbol{x}_{k+1} &= \boldsymbol{x}_k + \mathring{\boldsymbol{x}}_k \end{aligned} \tag{52}$$

ε_k is called the forcing term, and its value can be updated at each Newton iteration in order to optimize the convergence rate. More precisely, [9] shows that:

1. If $0 \leq \varepsilon_k \leq \varepsilon_{max} < 1 \ \ \forall k$, then the series converges linearly, with an asymptotic rate $a \leqslant \varepsilon_{max}$.
2. If $\lim_{k \to +\infty} \varepsilon_k = 0$, convergence is superlinear.
3. If $\varepsilon_k = O(\|\boldsymbol{L}(\boldsymbol{x}_k)\|)$, convergence is quadratic.

These convergence properties lead to more or less simple expressions for the *forcing terms* (see [9]):

$$\varepsilon_k = \frac{1}{2^{k+1}} \tag{53}$$

$$\varepsilon_k = \gamma \left(\frac{\|\boldsymbol{L}(\boldsymbol{x}_k)\|}{\|\boldsymbol{L}(\boldsymbol{x}_{k-1})\|} \right)^{\alpha}, \ \gamma \in [0,1[, \ \alpha \in]0,2] \tag{54}$$

$$\varepsilon_k = \frac{\left| \|\boldsymbol{L}(\boldsymbol{x}_k)\| - \|\boldsymbol{L}(\boldsymbol{x}_{k-1}) + \boldsymbol{L}'(\boldsymbol{x}_{k-1})\overset{\circ}{\boldsymbol{x}}_{k-1}\| \right|}{\|\boldsymbol{L}(\boldsymbol{x}_{k-1})\|} \tag{55}$$

Expression (53) is easy to implement but does not relate the linear criterion with $\|\boldsymbol{L}(\boldsymbol{x}_k)\|$ at each iteration: the convergence of the series $(\boldsymbol{x}_k)_k$ will theoretically not be better than superlinear. Expressions (54) and (55) are more complicated to implement, but can a priori allow faster convergence rates. For example taking $\alpha = 2$ in expression (54) should lead to quadratic convergence.

7 Assessments

The aim of these assessments is to test the various configurations of the approach to an academic problem with a realistic nonlinear behavior. We are aware that the methods we propose are sensitive to many parameters, among others the type of nonlinearities (with potential instabilities), the partitioning of the domain, the localization of the linearities (whether the nonlinearity spreads amongst subdomains or not), the variants of the solvers, the loading increments, the convergence criteria.

The following study is thus far from pretending to be exhaustive, it simply aims at giving major trends of the methods and at proving that they can be computationally interesting. At the end of the section, we discuss the applicability of the methods to other cases and the expected performance.

7.1 Perfect plasticity problem

We consider a rectangular plate in traction with a central hole (length $L_x = 95$ mm, height $L_y = 31.5$ mm, radius $r = 5$ mm) as depicted in Figure 2 (after taking into account the symmetries). The material is a perfectly plastic steel (Young modulus $E = 210$ GPa, Poisson's ratio $\nu = 0.3$, elasticity limit $\sigma_0 = 420$ MPa), and we assume the plane stress conditions. The mesh is constituted by triangle elements with three nodes, with characteristic length $h = 0.01\, L_y$. The eight subdomains are delimited by concentric circles. They are numbered starting from the center.

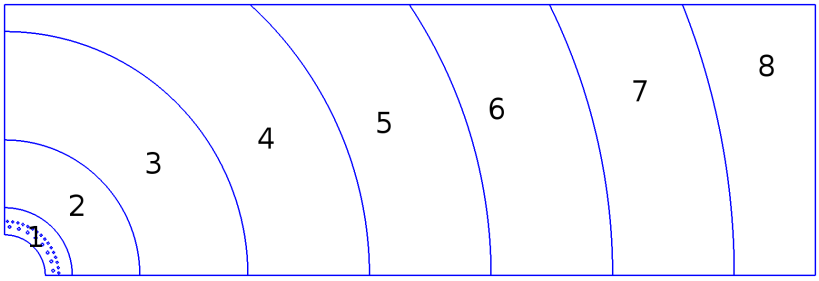


Figure 2: Localized plasticity problem / numbering of subdomains

Small perforations were added near the central hole in order to trigger a large level of localized plasticity. Traction loading is incrementally applied on the perforated plate by imposing a displacement u_D on its right side. Increments are taken as follows, with $u_e = 25$ μm:

$$u_D = [1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 5.75 \quad 6.5]\, u_e$$

The first step, corresponding to an increment of loading $u_D = u_e$, remains elastic. Steps $u_D = 2u_e$ to $u_D = 5u_e$ activate plasticity in only the central subdomain, reaching a maximum of $p = 0.051$ of cumulated plasticity. Plasticity then spreads in the second and third subdomains until $\max(p) = 0.15$.

This sequence of loadings was identified using a classical (outer Newton) approach. It could be reused as is with primal and mixed approaches. Solving a local Neumann problem with perfectly plastic subdomains is known to be a tougher problem than a Dirichlet or a Robin problem. Indeed, it was necessary to adopt a finer loading sequence for the dual approach; the following steps needed to be inserted:

$$u_D = [4.5 \quad 5.25 \quad 5.5 \quad 6 \quad 6.125 \quad 6.25 \quad 6.375]u_e$$

It was verified that, for the primal and mixed approaches, this refined sequence leads to results of comparable quality as the coarse sequence.

In the following, we compare the classical approach (outer Newton, inner Schur-Krylov), the primal and dual nonlinear approaches, and the mixed nonlinear approach with parameter $\boldsymbol{Q}_b$ equal to the stiffness of the neighbors, or $\boldsymbol{Q}_b$ equal to the optimal choice (Schur complement of the remainder of the structure). Of course the latter choice is not computationally realistic, and efficient approximations will be the subject of further studies.

In order to assess the performance, we measure:

- The number of outer Newton loops, since nonlinear approaches shall enable us to follow a faster path to convergence,
- The maximum number of computations of the tangent amongst subdomains. For the classical approach, it is simply equal to the number of outer iterations. The inner Newton loops makes that number larger in non-linear approaches.
- The number of cumulated Krylov (conjugate gradient) iterations which is proportional to the number of exchanges between subdomains.

The aim of the nonlinear approaches is to lower the number of outer Newton and Krylov iterations, at the price of some extra local Newton iterations. Of course the number of computations of the tangent varies from one subdomain to another, depending on the spread and intensity of the nonlinearity, which can cause poor load balancing. This is a real issue which will be the subject of future studies. Nevertheless we believe that the reduction of the number of outer iterations and of the exchanges (Krylov iterations) already makes the proposed methods interesting.

7.1.1 Classical technique

We briefly give the performance for the classical Newton method with a domain decomposition solver for the tangent systems. Note that BDD is used but FETI gives a quite similar number of Krylov iterations. The outer Newton's convergence criterion is set to $\varepsilon_{NG} = 10^{-5}$. For the results given in Table 1, the inner Krylov solver criteria for the condensed system is fixed to $\varepsilon_K = 10^{-6}$.

Increment of loading	u_e	$2u_e$	$5u_e$	$5.75u_e$	$6.5u_e$
Spread of nonlinearity	Elastic	1SD plastifies		Several SD plastify	
Global iterations	1	2	13	17	23
Cumulated Krylov iterations	19	38	255	336	460

Table 1: Performance of the classical approach with BDD solver for the tangent systems, with fixed criteria.

7.1.2 Nonlinear localization with fixed criteria

Before tuning the different thresholds, as developed in section 6.4, a comparison is made between the classical outer Newton and the nonlinear approaches with fixed stopping criteria. In order to achieve a meaningful comparison, thresholds are fixed to typical values:

$$\varepsilon_{NG} = 10^{-5}$$
$$\varepsilon_{NL} = \varepsilon_K = 10^{-6}$$

Relative results of primal, dual and mixed approaches normalized by the classical approach are given in Table 2 for chosen loading increments.

Table 2 clearly indicates a decrease in the number of cumulated global iterations when using the primal and mixed approaches: 20% less iterations for the primal and $\boldsymbol{K}_{bb}$-mixed approaches, 30% less iterations with the optimal Robin condition. For the dual method, the number of global iterations increases because of the steps inserted additionally to insure convergence.

The gain in Krylov iterations is quite similar to the gain in global Newton iterations, because each solution of a linear system requires an almost constant number of Krylov iterations (note that before requiring extra increments, the dual approach needs one Krylov iteration less than the primal approach, hence the $0.95 \simeq 18/19$ value in the table). These figures emphasize the need to adapt the precision of the Krylov solver as a function of the current nonlinear residual.

Regarding local Newton iterations, we see that, except for the dual method which is inefficient in that case, no more than 3 times more computations of the tangent are required for the nonlinear approaches. In particular, the optimal Robin only requires 2.4 times more computations of the tangent.

	Increment of loading	u_e	$2u_e$	$5u_e$	$5.75u_e$	$6.5u_e$
	Spread of nonlinearity	Elastic	SD 1		SD 1-2	SD 1-3
Global Newton	Primal/Classic	1	1	0.77	0.82	0.83
	Dual/Classic	1	1	1.46	2.12	3.35
	Mixed/Classic, $\boldsymbol{Q}_b = \boldsymbol{K}_{bb}$	1	1	0.77	0.82	0.78
	Mixed/Classic, $\boldsymbol{Q}_b$ opti	1	1	0.62	0.65	0.70
Krylov	Primal/Classic	1	1	0.76	0.82	0.82
	Dual/Classic	0.95	0.95	1.36	1.96	3.08
	Mixed/Classic, $\boldsymbol{Q}_b = \boldsymbol{K}_{bb}$	1	1	0.76	0.82	0.78
	Mixed/Classic, $\boldsymbol{Q}_b$ opti	1	1	0.61	0.64	0.69
Local Newton	Primal/Classic	1	2	2.62	2.88	3.04
	Dual/Classic	1	2	5.69	9.24	16.74
	Mixed/Classic, $\boldsymbol{Q}_b = \boldsymbol{K}_{bb}$	1	2	2.69	2.94	3
	Mixed/Classic, $\boldsymbol{Q}_b$ opti	1	2	1.85	2.12	2.43

Table 2: Ratios of cumulated iterations between nonlinearly localized methods and classic method, with fixed stopping criteria

As a conclusion for this test case, the primal and mixed versions of nonlinearly localized methods lead to interesting performance. Moreover they seem to be more effective when plasticity remains localized in only one subdomain. Indeed, global and Krylov numbers of iterations present a strong decrease when comparing them to those of the classic method (ratios from 0.61 to 0.76 at step $u_D = 5u_e$), which means a decrease in communications between subdomains. The cost of this improvement stays limited, with less than 3 times more factorizations at step $5u_e$ (and even less than 2 times more for optimal $\boldsymbol{Q}_b$). This is interesting because, as previously said, local iterations are independent and do not require communications between processors, which results in a gain of time for large -scale solutions.

The dual approach does not provide significant results, since additional increments had to be used to achieve convergence. Thus, it will not be considered in the following part, where better performance is achieved with an adaptation of stopping criteria.

7.1.3 Adaptation of stopping criteria

In order to observe the influence of the tuning of ε_K and ε_{NL} with respect to the current convergence state, the two expressions of section 6.4 are tested with a set of coefficients chosen for their performance (see Table 3). As discussed in section 6, we choose $\varepsilon_K = \varepsilon_{NL}$. The same sequence of load increments as that of the previous section is applied to the perforated plate.

Choice 1	Eq (54)	$\gamma = 0.7$, $\alpha = 1.5$, $\varepsilon_{K_0} = 10^{-6}$
Choice 2	Eq (55)	$\varepsilon_{K_0} = 10^{-4}$

Table 3: Chosen parameters for $\varepsilon_K = \varepsilon_{NL}$

The relative performance for choice 1 is summed up in Table 4, whereas the relative performance for choice 2 is given in Table 5. Depending on the quantity, we compare the variants with adapted criteria (INexact in the tables) to the fixed criteria approaches, or the nonlinear approaches with the classical approach, both with adapted criteria. Note that choice 2 was too aggressive for classic and primal nonlinear localized methods to converge, so that only the mixed approaches were studied.

The aim of adaption is to avoid useless inner iterations without modifying the global iterations. This absence of perturbation corresponds to ratios of 1 in the "Global Newton" rows. This happens almost every time.

Regarding the number of Krylov iterations, the figures in Table 4 are not as good, but still of the same order as the ones in Table 2. Table 5 shows that mixed approaches can be really efficient with a good adaption of criteria. This means that adaption is not in contradiction with nonlinear localization and shall be used to reduce the number of exchanges.

	Increment of loading	u_e	$2u_e$	$5u_e$	$5.75u_e$	$6.5u_e$
	Spread of nonlinearity	Elastic	SD 1		SD 1-2	SD 1-3
Global Newton	Classic IN/Classic	1	1	1	1.06	1
	Primal IN/Primal	1	1	1	1	1
	Mixed IN/Mixed, $\boldsymbol{Q}_b = \boldsymbol{K}_{bb}$	1	1	1	1	1.06
	Mixed IN/Mixed, $\boldsymbol{Q}_b$ opti	1	1	1	1.09	1.06
Krylov	Classic IN/Classic	1	1	0.77	0.77	0.69
	Primal IN/Classic IN	1	1	0.86	0.85	0.89
	Mixed IN/Classic IN, $\boldsymbol{Q}_b = \boldsymbol{K}_{bb}$	1	1	0.86	0.85	0.91
	Mixed IN/Classic IN, $\boldsymbol{Q}_b$ opti	1	1	0.73	0.73	0.79
Local Newton	Classic IN/Classic	1	1	1	1.06	1
	Primal IN/Classic IN	1	2	2.54	2.56	2.78
	Mixed IN/Classic IN, $\boldsymbol{Q}_b = \boldsymbol{K}_{bb}$	1	2	2.31	2.33	2.57
	Mixed IN/Classic IN, $\boldsymbol{Q}_b$ opti	1	2	1.85	2.06	2.30

Table 4: Performance of adapted criteria with choice 1.

	Increment of loading	u_e	$2u_e$	$5u_e$	$5.75u_e$	$6.5u_e$
	Spread of nonlinearity	Elastic	SD 1		SD 1-2	SD 1-3
Global Newton	Mixed IN/Mixed, $\boldsymbol{Q}_b = \boldsymbol{K}_{bb}$	1	1	1	1	1.06
	Mixed IN/Mixed, $\boldsymbol{Q}_b$ opti	1	1	1	1	1
Krylov	Mixed IN/Classic IN, $\boldsymbol{Q}_b = \boldsymbol{K}_{bb}$	1	0.79	0.71	0.67	0.66
	Mixed IN/Classic IN, $\boldsymbol{Q}_b$ opti	1	0.79	0.58	0.56	0.57
Local Newton	Mixed IN/Classic IN, $\boldsymbol{Q}_b = \boldsymbol{K}_{bb}$	1	2	2.23	2.28	2.43
	Mixed IN/Classic IN, $\boldsymbol{Q}_b$ opti	1	2	1.85	1.83	2.04

Table 5: Performance of adapted criteria with choice 2.

Concerning local Newton iterations, which correspond to the extra cost of nonlinear localization techniques, we see that adaption makes it possible to reduce their number, with slightly smaller figures in Tables 4 than in Table 2, and more significant improvement in Table 5 for mixed approaches.

7.2 Discussion

Assessments proved that the nonlinear localization methods were able to reduce the number of Krylov iterations (and then the number of communication) at the price of some extra local computations (factorization), without modifying the number of global (outer Newton) iterations. Unfortunately our `octave-mpi` implementation does not allow reliable time measurements and large simulations. Moreover, the time performance of the methods would be highly dependent on the hardware and in particular on the performance of the network compared to the performance of the cores.

We chose the previous experiment in order to prove that performance is highly dependent on the chosen boundary condition. In particular Neumann boundary conditions for perfectly plastic domains lead to very stiff problems, because in that case plasticity is estimated from the above (plasticity is overestimated and Newton iterations to reduce it are conducted on very unrealistic configurations). During our experimentations, we were able to design configurations with positive hardening where the dual approach was equivalent or even better than the primal.

More generally, it is easy to build test cases where one variant behaves poorly, or even fails. Figure 3 presents two such cases. In the first case, the decomposition makes subdomains very slender and prone to buckling (assuming large transformations). In that case, the method may converge (slowly) to a suboptimal solution [21]. In the second case, the decomposition amplifies the initial deformation, and assuming damageable behavior, subdomains may fail even if

the global problem was well-posed.

On the contrary, any experiment where the chosen boundary condition is barely influenced by the nonlinearity will lead to better parallel performance than a linear problem: an extreme case is an isostatic elastoplastic lattice solved with a dual approach.

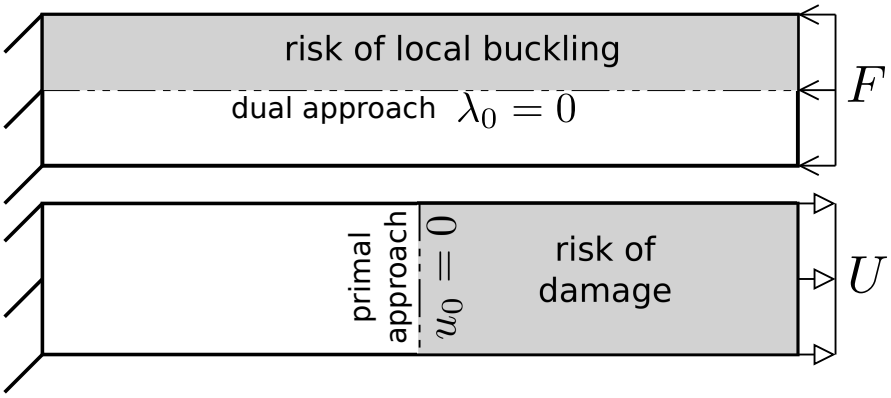


Figure 3: Two critical cases

We can thus draw the following conclusions:

- The method may not converge or follow an inadequate path if local instabilities are possible. Based on the literature of Schwarz methods for nonlinear problems [1, 31], we expect the method to converge unconditionally (assuming small enough increments) in the case of coercive maximal monotone operators (typically positive hardening in small strains) where the existence and interesting properties of nonlinear Schur complements can be established [42].
- The shape of the subdomains shall play an important role in the potential development of nonlinear effects.
- The initial boundary condition is critical, zero may be a very bad guess which triggers unduly complex local nonlinear computations. It is worth considering starting the method by a global elastic prediction.
- Well chosen Robin conditions should help follow the good path (the one of the monolithic solution). In the example above, we could compute the tangent Schur complement of the remainder of the structure which is (first order) optimal, but it is of pure academic interest, since the associated numerical cost would not be acceptable for actual industrial simulations; approximations of that operator is a classical question [23, 15] which we will address in future works. The approximation studied in the example, constituted by the stiffness of the neighbors, is not flexible enough, and it is not expected to give good results when the remainder of the structure is very compliant.

8 Conclusion

In this article, we have proposed nonlinear versions of FETI, BDD and FETI2LM algorithms where nonlinear computations are carried out independently on the subdomains. The starting point is a "nonlinear condensation" of the problems which leads, after linearization, to the parallel evaluation of the residual by nonlinear problems with boundary conditions characteristic of the method and to the solution to a classical DD system using any classical linear solver amongst FETI, BDD or FETI2NL.

The method modifies the nonlinear system to be solved and introduces inner inexact solvers, which implies a careful tuning of the stopping criteria. We proposed to exploit a classical inexact Newton formula.

A first assessment proved the potential of the method with an interesting trade-off between Krylov iterations (which corresponds to network communications) and Local Newton iterations (associated with local CPU computations).

However the method suffers from several drawbacks. First, the performance depends on the chosen boundary condition, which is a strong difference with the linear case. Second, the shape of the subdomains plays an even greater role than in the linear case, since it can trigger unphysical nonlinear effects. Third, in the case of localized nonlinearity, the method makes it possible to concentrate computations on the nonlinear subdomains but linear subdomains are then inactive, leading to poor load balancing (but potential power saving when idling processors enter standby mode). The question of load balancing will be addressed in future work when a more capable implementation is available. Typically we believe that it will be possible to propose an efficient strategy by encompassing the nonlinear problem in a mesh/substructuring adaption loop when a guaranteed precision is required by the user [38].

Much work is thus still needed before the method is reliable. We believe that well-chosen Robin conditions are an important ingredient to make the method stable and efficient.

References

[1] L. Badea. On the Schwarz alternating method with more than two subdomains for nonlinear monotone problems. *SIAM J. Numer. Anal.*, 28(1):179–204, 1991.

[2] Felipe Bordeu, Pierre-Alain Boucard, and Pierre Gosselet. Balancing domain decomposition with nonlinear relocalization: parallel implementation for laminates. In *Proceedings of the first international conference on parallel, distributed and grid computing for engineering*, pages CCP: 90, paper : 4, Pécs, 2009.

[3] D. Brands, A. Klawonn, O. Rheinbach, and J. Schröder. Modelling and convergence in arterial wall simulations using a parallel FETI solution strategy. *Comput. Meth. Appl. Mech. Engrg.*, 11(5):569–583, October 2008.

[4] Xiao-Chuan Cai and David E. Keyes. Nonlinearly preconditioned inexact Newton algorithms. *SIAM J. Sci. Comput.*, 24(1):183–200, 2002.

[5] P.G. Ciarlet. *Linear and nonlinear functional analysis with applications*. SIAM, 2013.

[6] P. Cresta, O. Allix, C. Rey, and S. Guinard. Nonlinear localization strategies for domain decomposition methods: Application to post-buckling analyses. *Computer Methods in Applied Mechanics and Engineering*, 196:1436–1446, 2007.

[7] Ron S Dembo, Stanley C Eisenstat, and Trond Steihaug. Inexact Newton methods. *SIAM Journal on Numerical analysis*, 19(2):400–408, 1982.

[8] Maksymilian Dryja and Wolfgang Hackbusch. On the nonlinear domain decomposition method. *BIT*, 37(2):296–311, 1997.

[9] Stanley C Eisenstat and Homer F Walker. Choosing the forcing terms in an inexact Newton method. *SIAM Journal on Scientific Computing*, 17(1):16–32, 1996.

[10] C. Farhat, M. Lesoinne, P. LeTallec, K. Pierson, and D. Rixen. FETI-DP: a dual-primal unified FETI method - part i: a faster alternative to the two-level FETI method. *International Journal for Numerical Methods in Engineering*, 50(7):1523–1544, 2001.

[11] C. Farhat, K. Pierson, and M. Lesoine. The second generation FETI methods and their application to the parallel solution of large-scale linear and geometrically non-linear structural analysis problems. *Computer Methods in Applied Mechanics and Engineering*, 184(2-4):333–374, 2000.

[12] C. Farhat and F. X. Roux. The dual Schur complement method with well-posed local Neumann problems. *Contemporary Mathematics*, 157:193, 1994.

[13] C. Farhat and F. X. Roux. Implicit parallel processing in structural mechanics. *Computational Mechanics Advances*, 2(1):1–124, 1994. North-Holland.

[14] M. Feistauer and A. Zenisek. Finite element solution of nonlinear elliptic problems. *Numerische Mathematik*, 50:451–475, 1987.

[15] Lionel Gendre, Olivier Allix, and Pierre Gosselet. A two-scale approximation of the Schur complement and its use for non-intrusive coupling. *International Journal for Numerical Methods in Engineering*, 87(9):889–905, 2011.

[16] Pierre Gosselet. Domain decomposition methods for nonlinear problems in structural mechanics. Seminars in mathematical faculty of Essen (Germany), lab. N. Oresme of Caen, lab. MSSMat of école centrale de Paris, lab. GeM of école centrale Nantes (France); slides available on hal.archives-ouvertes.fr, 2011.

[17] Pierre Gosselet and Christian Rey. Non-overlapping domain decomposition methods in structural mechanics. *Archives of computational methods in engineering*, 13(4):515–572, 2007.

[18] Pierre Gosselet, Christian Rey, and Julien Pebrel. Total and selective reuse of Krylov subspaces for the solution to a sequence of nonlinear structural problems. *International Journal for Numerical Methods in Engineering*, 94(1):60–83, 2013.

[19] Pierre Gosselet, Daniel Rixen, François-Xavier Roux, and Nicole Spillane. Simultaneous-FETI and Block-FETI: robust domain decomposition with multiple search directions. *International Journal for Numerical Methods in Engineering*, 104(10):905–927, 2015.

[20] D. Hauer. The p-Dirichlet-to-Neumann operator with applications to elliptic and parabolic problems. *Journal of diffential equations*, 259(8):3615–3655, 2015.

[21] J. Hinojosa, O. Allix, P.A. Guidault, and P. Cresta. Domain decomposition methods with nonlinear localization for the buckling and post-buckling analyses of large structures. *Advances in Engineering Software*, 70:13–24, 2014.

[22] Feng-Nan Hwang and Xiao-Chuan Cai. A class of parallel two-level nonlinear Schwarz preconditioned inexact Newton algorithms. *Computer Methods in Applied Mechanics and Engineering*, 196(8):1603–1611, 2007.

[23] C. Japhet, F. Nataf, and F. Rogier. The optimized order 2 method. application to convection-diffusion problems. *Future Generation Computer Systems*, 18(1):17–30, 2001.

[24] C. Kelley. *Solving Nonlinear Equations with Newton's Method*. Society for Industrial and Applied Mathematics, 2003.

[25] Axel Klawonn, Martin Lanser, and Oliver Rheinbach. Nonlinear FETI-DP and BDDC methods. *SIAM J. Sci. Comput.*, 36(2):A737–A765, 2014.

[26] Axel Klawonn and Oliver Rheinbach. Robust FETI-DP methods for heterogeneous three dimensional elasticity problems. *Computer Methods in Applied Mechanics and EngineerinComputer Methods in Applied Mechanics and Engineering*, 196(8):1400–1414, 2007.

[27] Axel Klawonn, Oliver Rheinbach, and Olof B. Widlund. An analysis of a FETI–DP algorithm on irregular subdomains in the plane. *SIAM J. Numer. Anal.*, 46(5):2484–2504, 2008.

[28] Axel Klawonn and Olof Widlund. FETI and Neumann-Neumann iterative substructuring methods: Connections and new results. *Communications on Pure and Applied Mathematics*, 54(1):57–90, 2001.

[29] P. Ladevèze. Sur une famille d'algorithmes en mécanique des structures. *Comptes Rendus Académie des Sciences - Mécanique, Paris*, 300(2):41–45, 1985.

[30] P. Ladevèze. *Nonlinear Computational Structural Mechanics – New Approaches and Non-Incremental Methods of Calculation*. Springer Verlag, 1999.

[31] P. Ladevèze, D. Néron, and P. Gosselet. On a mixed and multiscale domain decomposition method. *Computer Methods in Applied Mechanics and Engineering*, 196(8):1526–1540, 2007.

[32] Patrick Le Tallec. Domain decomposition methods in computational mechanics. *Comput. Mech. Adv.*, 1(2):121–220, 1994.

[33] Jan Mandel. Balancing domain decomposition. *Communications in Numerical Methods in Engineering*, 9(3):233, 1993.

[34] Augustin Parret-Fréaud, Christian Rey, Pierre Gosselet, and Frédéric Feyel. Fast estimation of discretization error for FE problems solved by domain decomposition. *Computer Methods in Applied Mechanics and Engineering*, 199(49-52):3315–3323, 2010.

[35] J. Pebrel, Christian Rey, and P. Gosselet. A nonlinear dual-domain decomposition method: Application to structural problems with damage. *International Journal for Multiscale Computational Engineering*, 6(3):251, 2008.

[36] Julien Pebrel, Pierre Gosselet, and Christian Rey. Etude du choix des conditions d'interface pour des stratégies non linéaire de décomposition de domaine. In *Actes du neuvième colloque national en calcul des structures*, volume 2, pages 393–398, Giens (Var), 2009.

[37] Christian Rey and Franck Risler. A Rayleigh–Ritz preconditioner for the iterative solution to large scale nonlinear problems. *Numerical Algorithms*, 17(3/4):279–311, 1998.

[38] Valentine Rey, Christian Rey, and Pierre Gosselet. A strict error bound with separated contributions of the discretization and of the iterative solver in non-overlapping domain decomposition methods. *Computer Methods in Applied Mechanics and Engineering*, 270(1):293–303, 2014.

[39] F. Risler and C. Rey. Iterative accelerating algorithms with Krylov subspaces for the solution to large-scale non-linear problems. *Numerical algorithms*, 23:1, 2000.

[40] Daniel J. Rixen and Charbel Farhat. A simple and efficient extension of a class of substructure based preconditioners to heterogeneous structural mechanics problems. *International Journal for Numerical Methods in Engineering*, 44(4):489–516, 1999.

[41] François-Xavier Roux. A feti-2lm method for non-matching grids. In Michel Bercovier, MartinJ. Gander, Ralf Kornhuber, and Olof Widlund, editors, *Domain Decomposition Methods in Science and Engineering XVIII*, volume 70 of *Lecture Notes in Computational Science and Engineering*, pages 121–128. Springer Berlin Heidelberg, 2009.

[42] R.E. Showalter. *Hilbert space methods for partial differential equations*. Monographs and studies in mathematics. Pitman, London, 1977.

[43] R.E. Showalter. *Monotone operators in Banach space and nonlinear partial differential equations*, volume 49 of *Mathematical surveys and monographs*. American mathematical society, 1997.

[44] N. Spillane, V. Dolean, P. Hauret, F. Nataf, C. Pechstein, and R. Scheichl. Abstract robust coarse spaces for systems of PDEs via generalized eigenproblems in the overlaps. *Numer. Math.*, 126(4):741–770, 2014.

D Article [Gosselet *et al.*, 2013], recyclage des espaces de Krylov

Cet article présente une technique de recyclage de l'information numérique dans le cas où une succession de systèmes linéaires (à matrice non constante) doit être résolue. Cela arrive typiquement quand on utilise un solveur de Newton ou lorsqu'on explore un plan d'expérience (par exemple sur des coefficients matériaux). La méthode s'appuie sur une décomposition de domaine de type FETI (ou BDD) avec augmentation du solveur de Krylov (ici gradient conjugué). L'espace d'augmentation est obtenu par une analyse de l'espace de Krylov généré lors de la résolution du système linéaire précédent. Un critère permet de sélectionner les vecteurs de Ritz qui approchent le mieux les véritables vecteurs propres du système aux valeurs propres généralisées constitué par l'opérateur courant et son préconditionneur. L'espoir est que ces vecteurs restent pertinents dans la nouvelle configuration de calcul.

Il est à noter que cette méthode donne une heuristique pour approcher à coût nul les vecteurs propres qui pénalisent la convergence et qui sont typiquement détectés par des méthodes *a priori* de type GENEO [Spillane *et al.*, 2014].

Total and selective reuse of Krylov subspaces for the resolution of sequences of nonlinear structural problems

P. Gosselet, C. Rey, J. Pebrel
LMT Cachan, ENS Cachan/CNRS/UPMC/PRES UniverSud Paris
61 Avenue du Président Wilson, 94235 Cachan, France

November 4, 2016

Abstract

This paper deals with the definition and optimization of augmentation spaces for faster convergence of the conjugate gradient method in the resolution of sequences of linear systems. Using advanced convergence results from the literature, we present a procedure based on a selection of relevant approximations of the eigenspaces for extracting, selecting and reusing information from the Krylov subspaces generated by previous solutions in order to accelerate the current iteration. Assessments of the method are proposed in the cases of both linear and nonlinear structural problems.

Keywords: Krylov solvers; multiresolution; model reduction.

1 Introduction

Accelerating the convergence of Krylov iterative solvers [44] is an old issue which has returned to the spotlight because of the increasing number of applications for which these are preferred to direct solvers today. Traditional approaches aim at improving the condition number by using frameworks in which efficient preconditioners exist (e.g. domain decomposition methods [25, 17]), or for which good initialization vectors [19], relevant augmentation subspaces [6, 43, 5] or suitable block strategies (see [1] for a very general block-Lanczos algorithm) are available. For instance:

- For 3D elasticity problems, domain decomposition methods come with "physical" augmentation associated with the global equilibrium of floating substructures (rigid body motions), which makes the methods scalable [13, 30]; for plate and shell problems, additional augmentation through "corner modes" [11, 9, 26] is required.
- For structures with repeated patterns, block strategies are possible [20].
- For restarted algorithms, one can use deflation or augmentation [33, 7], or block techniques [3].
- For problems with multiple right-hand sides, deflation [10, 46, 8] is a rather classical approach.

The problem with these techniques is that they require some *a priori* information which is seldom available, except in specific cases.

Many recent works present theoretical and practical comparisons of the numerous algorithms which have been developed in connection with these ideas [48, 47].

Multiresolution approaches form a general framework in which numerical information is available to accelerate the convergence of Krylov solvers. Multiresolution refers to situations in which the solution of a mechanical problem cannot be achieved through the resolution of a single linear system. For example, calculating the solution of a nonlinear or time-dependent problem or exploring a design of experiment during an optimization procedure requires the resolution of sequences of linear systems. Multiresolution is more general than using multiple right-hand sides because the matrices themselves are likely to change from one system to another (multiple right-hand and left-hand sides). Thus, the problem consists in solving a k-indexed family of large, sparse, linear $n \times n$ systems of the form:

$$\boldsymbol{A}^{(k)} x^{(k)} = b^{(k)} \tag{1}$$

Although different, the systems are assumed to be similar to one another. This similarity can be defined in several ways: in terms of rank, by the fact that $\text{rank}(\boldsymbol{A}^{(k)} - \boldsymbol{A}^{(k-1)}) \ll \text{rank}(\boldsymbol{A}^{(k)})$; or in a spectral sense by the fact that the eigenspaces remain stable from one system to another; or in terms of the Krylov subspaces generated [4]. The first case can be dealt with easily, even with direct solvers, by using the Sherman-Morrison formula; the second case requires augmentation strategies in order to eliminate the most penalizing part of the spectrum and improve the active condition number[1] [39, 16, 38, 50]; and the last case calls for preconditioning techniques [40, 41].

While most of the studies of Krylov methods for multiresolution (often referred to as the recycling of Krylov subspaces) are set in the framework of GMRes/MinRes [38, 50], we chose to work on the specific case of the resolution of symmetric, positive definite systems using conjugate gradients (CGs), in which the convergence is under control and related to easily calculated spectral properties [49]. In earlier works, the authors developed efficient preconditioners based on previous Krylov subspaces [40, 41] which took advantage of the conjugation properties of CGs, but did not extract the most interesting part of the information available in the Krylov subspaces, and they proposed augmentation techniques using Ritz vectors [39]. Typically, these works were aimed at nonlinear mechanical systems solved by Newton-Raphson linearization and FETI or BDD domain decomposition [18, 27].

The recycling of Krylov subspaces can also be analyzed from the model reduction point of view. Since Krylov solvers satisfy Petrov-Galerkin conditions, they share many common points with strategies based on Karhunen-Loeve expansion [31, 42, 36]. These similarities are well-known [15, 14, 21]. But our objective is not to develop reduced models of mechanical systems in order to perform fast but coarse analyzes; it is to define, improve and reuse reduced models in order to carry out calculations both rapidly and accurately.

In this paper, we undertake a more in-depth investigation of augmentation using a selection of post-processed Ritz vectors. In Section 2, we begin with a detailed presentation of the theoretical framework of the augmented preconditioned conjugate gradient method; then, in Section 3, we propose a first reuse algorithm in a multiresolution framework (TRKS); in Section 4, we improve this algorithm by proposing a procedure for selecting the "best" Ritz vectors (SRKS and "cluster"); finally, in Section 5, we propose an evaluation of the method in the case of nonlinear mechanics and parametric problems, using domain decomposition methods [17] to define efficient preconditioners.

2 The augmented preconditioned conjugate gradient method

2.1 Algorithm and properties

Let us consider the linear problem

$$\boldsymbol{A}x = b, \tag{2}$$

where $\boldsymbol{A}$ is an $n \times n$ symmetric positive definite matrix, and let us study the resolution of this system using the augmented preconditioned conjugate gradient algorithm. With $\boldsymbol{M}$ being the $n \times n$ symmetric positive definite matrix of the preconditioner, we introduce the following notations:

$$\begin{aligned} i = 0 \dots m \qquad & \text{the iteration number} \\ x_i \qquad & \text{the } i^{th} \text{ approximation} \\ r_i = b - Ax_i = A(x - x_i) \qquad & \text{the } i^{th} \text{ residual} \end{aligned} \tag{3}$$

With no loss of generality, the presentation can be limited to the case of a zero initial guess $x_{00} = 0$. (Otherwise, one can set $b \leftarrow b - \boldsymbol{A}x_{00}$.)

Let $\mathcal{C}$ be a subspace of $\mathbb{R}^n$ of dimension n_c, and let Matrix $\boldsymbol{C} = [c_1, \dots, c_{n_c}]$ be a basis of $\mathcal{C}$. The search principle of the augmented left-preconditioned conjugate gradient is:

$$\left\{ \begin{array}{ll} \text{find} & x_i \in \mathcal{K}_i(\boldsymbol{M}^{-1}\boldsymbol{A}, \mathcal{C}, \boldsymbol{M}^{-1}r_0) \\ \text{such that} & r_i \perp \mathcal{K}_i(\boldsymbol{M}^{-1}\boldsymbol{A}, \mathcal{C}, \boldsymbol{M}^{-1}r_0) \end{array} \right. \tag{4}$$

where $\mathcal{K}_i(\boldsymbol{M}^{-1}\boldsymbol{A}, \mathcal{C}, \boldsymbol{M}^{-1}r_0)$ is the augmented Krylov subspace associated with preconditioned operator $\boldsymbol{M}^{-1}\boldsymbol{A}$ and augmentation subspace $\mathcal{C}$:

$$\mathcal{K}_i(\boldsymbol{M}^{-1}\boldsymbol{A}, \mathcal{C}, \boldsymbol{M}^{-1}r_0) = \text{span}\left(\boldsymbol{M}^{-1}r_0, \dots, (\boldsymbol{M}^{-1}\boldsymbol{A})^{(i-1)}\boldsymbol{M}^{-1}r_0\right) \oplus \mathcal{C} \tag{5}$$

[1] "active" referring to the part of the spectrum of the matrix which is solicited by the right-hand side.

A classical implementation relies on the definition of a convenient initialization and projector pair (x_0,$\boldsymbol{P}$):

$$x = x_0 + \boldsymbol{P}y \quad \begin{cases} \boldsymbol{C}^T r_0 = 0 & \Leftarrow \quad x_0 = \boldsymbol{C}(\boldsymbol{C}^T\boldsymbol{A}\boldsymbol{C})^{-1}\boldsymbol{C}^T b \\ \boldsymbol{C}^T\boldsymbol{A}\boldsymbol{P} = 0 & \Leftarrow \quad \boldsymbol{P} = \boldsymbol{I} - \boldsymbol{C}(\boldsymbol{C}^T\boldsymbol{A}\boldsymbol{C})^{-1}\boldsymbol{C}^T\boldsymbol{A} \end{cases} \tag{6}$$

One should note that since $\boldsymbol{A}\boldsymbol{P} = \boldsymbol{P}^T\boldsymbol{A}\boldsymbol{P} = \boldsymbol{P}^T\boldsymbol{A}$ augmentation preserves symmetry. One should also note that $\boldsymbol{P}\boldsymbol{C} = 0$. The system to be solved is:

$$\boldsymbol{A}\boldsymbol{P}y = (\boldsymbol{P}^T\boldsymbol{A}\boldsymbol{P})y = r_0 = \boldsymbol{P}^T b \tag{7}$$

The $\boldsymbol{C}$-augmented, $\boldsymbol{M}$-preconditioned conjugate gradient technique (APCG) implemented by projection is presented in Algorithm 1. (For the sake of simplicity, the methods will be described assuming exact arithmetic, even though they are compatible with more realistic full reorthogonalization [28].)

Algorithm 1: APCG($\boldsymbol{A}, \boldsymbol{M}, \boldsymbol{C}$, b)

Calculate $\boldsymbol{A}\boldsymbol{C}$, $(\boldsymbol{C}^T\boldsymbol{A}\boldsymbol{C})^{-1}$; $(\boldsymbol{P} = \boldsymbol{I} - \boldsymbol{C}\left(\boldsymbol{C}^T\boldsymbol{A}\boldsymbol{C}\right)^{-1}\boldsymbol{C}^T\boldsymbol{A})$;
$x_0 = \boldsymbol{C}(\boldsymbol{C}^T\boldsymbol{A}\boldsymbol{C})^{-1}\boldsymbol{C}^T b$;
$r_0 = b - \boldsymbol{A}x_0 = \boldsymbol{P}^T b$;
$z_0 = \boldsymbol{P}\boldsymbol{M}^{-1}r_0$, $w_0 = z_0$;
for $j = 1, \ldots, m$ **do**
  $\alpha_{j-1} = (r_{j-1}, w_{j-1})/(\boldsymbol{A}w_{j-1}, w_{j-1})$
  $x_j = x_{j-1} + \alpha_{j-1}w_{j-1}$
  $r_j = r_{j-1} - \alpha_{j-1}\boldsymbol{A}w_{j-1}$
  $z_j = \boldsymbol{P}\boldsymbol{M}^{-1}r_{j+1}$
  $w_j = z_j - \beta_j w_{j-1}$
  $\beta_j = (\boldsymbol{A}w_{j-1}, z_j)/(w_{j-1}, \boldsymbol{A}w_{j-1})$
end

The following basic relations hold:

$$\begin{aligned} (r_i, z_j) &= 0, \quad i \neq j \\ (w_i, \boldsymbol{A}w_j) &= 0, \quad i \neq j \end{aligned} \tag{8}$$

With $\boldsymbol{W}_i = [w_0, \ldots, w_{i-1}]$ and $\boldsymbol{Z}_i = [z_0, \ldots, z_{i-1}]$ being two bases of $\mathcal{K}_i(\boldsymbol{P}\boldsymbol{M}^{-1}\boldsymbol{A}, z_0)$, the projector enables the spaces to be divided orthogonally:

$$\mathcal{K}_i(\boldsymbol{M}^{-1}\boldsymbol{A}, \mathcal{C}, \boldsymbol{M}^{-1}r_0) = \mathcal{K}_i(\boldsymbol{P}\boldsymbol{M}^{-1}\boldsymbol{A}, z_0) \overset{\perp_{\boldsymbol{A}}}{\oplus} \mathcal{C} \tag{9}$$

Of course, in the absence of optional constraints ($\boldsymbol{C} = 0$, $\boldsymbol{P} = \boldsymbol{I}$), APCG reduces to standard preconditioned conjugate gradients PCG($\boldsymbol{A}$,$\boldsymbol{M}$, b); if, in addition, $\boldsymbol{M}^{-1} = \boldsymbol{I}$, it becomes a standard conjugate gradient algorithm CG($\boldsymbol{A}$, b).

Let us recall a first result which was proven in [6] for the case of non-preconditioned augmented conjugate gradients.

Proposition 1. *Let* $\mathcal{V} = \mathrm{Range}(\boldsymbol{P})$

- *APCG($\boldsymbol{A}, \boldsymbol{I}, \boldsymbol{C}$, b) is equivalent to CG($\boldsymbol{P}^T\boldsymbol{A}\boldsymbol{P}_{|\boldsymbol{A}\mathcal{V}}$, $\boldsymbol{P}^T b$) in the sense that both generate the same residuals. x_i, the i^{th} APCG approximation, is connected to y_i, the i^{th} CG approximation, by $x_i = x_0 + \boldsymbol{P}y_i$.*
- *APCG($\boldsymbol{A}, \boldsymbol{I}, \boldsymbol{C}$, b) does not break down; it converges, and its asymptotic convergence rate is governed by the condition number $\kappa\left(\boldsymbol{P}^T\boldsymbol{A}\boldsymbol{P}_{|\boldsymbol{A}\mathcal{V}}\right) \leqslant \kappa\left(\boldsymbol{A}\right)$.*

Consequently, augmentation strategies never decrease the asymptotic convergence rate. The following corollary is straightforward:

Corollary 1. *Let $\boldsymbol{D} = [d_1, \ldots, d_{m_d}]$ be a set of m_d linearly independent vectors such that $\boldsymbol{E} = [\boldsymbol{C}, \boldsymbol{D}]$ is a full column rank matrix. Let $\boldsymbol{P}_{\boldsymbol{E}} = \boldsymbol{I} - \boldsymbol{E}(\boldsymbol{E}^T\boldsymbol{A}\boldsymbol{E})^{-1}\boldsymbol{E}^T\boldsymbol{A}$, and let $\mathcal{V}_{\boldsymbol{E}}$ be the range of $\boldsymbol{P}_{\boldsymbol{E}}$.*

- *APCG($\boldsymbol{A}, \boldsymbol{I}, \boldsymbol{E}$, b) is equivalent to APCG($\boldsymbol{P}^T\boldsymbol{A}\boldsymbol{P}$, $\boldsymbol{I}$, $\boldsymbol{D}$, $\boldsymbol{P}^T b$) in the sense that both generate the same residual. $x_i^{\boldsymbol{E}}$, the i^{th} approximation of APCG($\boldsymbol{A}, \boldsymbol{I}, \boldsymbol{E}$, b), is connected to $x_i^{\boldsymbol{D}}$, the i^{th} approximation of APCG($\boldsymbol{P}^T\boldsymbol{A}\boldsymbol{P}$, $\boldsymbol{I}$, $\boldsymbol{D}$, $\boldsymbol{P}^T b$), by $x_i^{\boldsymbol{E}} = x_0 + \boldsymbol{P}x_i^{\boldsymbol{D}}$.*

- *The asymptotic convergence rate is governed by* $\kappa\left({\boldsymbol{P_E}}^T \boldsymbol{A} \boldsymbol{P_{E|\mathcal{AV}_E}}\right) \leqslant \kappa\left(\boldsymbol{P}^T \boldsymbol{A} \boldsymbol{P}_{|\boldsymbol{A}\mathcal{V}}\right) \leqslant \kappa\left(\boldsymbol{A}\right)$.

In conclusion, an increase in the size of the augmentation can only improve the asymptotic rate of convergence. (In the worst case, it leaves it unchanged.)

Now let us focus on the effect of preconditioning. Since $\boldsymbol{M}$ is a symmetric positive definite matrix, it can be factorized in Cholesky form $\boldsymbol{M} = \boldsymbol{L}\boldsymbol{L}^T$ (where $\boldsymbol{L}$ denotes a lower triangular matrix with positive diagonal coefficients). Let us introduce the notation:

$$\begin{array}{l} \hat{\boldsymbol{A}} = \boldsymbol{L}^{-1}\boldsymbol{A}\boldsymbol{L}^{-T} \quad ; \quad \hat{b} = \boldsymbol{L}^{-1} b \quad ; \quad \hat{x} = \boldsymbol{L}^T x \\ \hat{\boldsymbol{C}} = \boldsymbol{L}^T \boldsymbol{C} \quad ; \quad \hat{\boldsymbol{P}} = \boldsymbol{I} - \hat{\boldsymbol{C}}(\hat{\boldsymbol{C}}^T \hat{\boldsymbol{A}} \hat{\boldsymbol{C}})^{-1} \hat{\boldsymbol{C}}^T \hat{\boldsymbol{A}} \end{array} \tag{10}$$

Then, the following equivalence between preconditioned and non-preconditioned augmented conjugate gradients holds:

Proposition 2. *APCG(*$\boldsymbol{A}, \boldsymbol{M}, \boldsymbol{C}$*, b) is equivalent to APCG(*$\hat{\boldsymbol{A}}$*,* $\boldsymbol{I}$*,* $\hat{\boldsymbol{C}}$*,* $\hat{b}$*) with* $\hat{r} = \boldsymbol{L}^{-1} r = \hat{z} = \boldsymbol{L}^T z$*,* $\hat{w} = \boldsymbol{L}^T w$*,* $\hat{\alpha} = \alpha$ *and* $\hat{\beta} = \beta$*. Its asymptotic convergence rate is governed by* $\kappa\left(\hat{\boldsymbol{P}}^T \hat{\boldsymbol{A}} \hat{\boldsymbol{P}}_{|\hat{\boldsymbol{A}}\hat{\mathcal{V}}}\right) \leqslant \kappa\left(\hat{\boldsymbol{A}}\right)$.

Proof. Since $\hat{\boldsymbol{P}} = \boldsymbol{L}^T \boldsymbol{P} \boldsymbol{L}^{-T}$, we obtain directly $\hat{x_0} = \hat{\boldsymbol{C}}(\hat{\boldsymbol{C}}^T \hat{\boldsymbol{A}} \hat{\boldsymbol{C}})^{-1} \hat{\boldsymbol{C}}^T \hat{b} = \boldsymbol{L}^T x_0$, $\hat{r_0} = \hat{b} - \hat{\boldsymbol{A}} \hat{x_0} = \boldsymbol{L}^{-1} r_0 = \hat{z_0} = \boldsymbol{L}^T z_0$ and $\hat{w_0} = \boldsymbol{L}^T w_0$. By induction, it follows that $\hat{\alpha}_{j-1} = (\hat{r}_{j-1}, \hat{z}_{j-1})/(\hat{\boldsymbol{A}} \hat{w}_{j-1}, \hat{w}_{j-1}) = \alpha_{j-1}$, $\hat{r_j} = \boldsymbol{L}^{-1} r_j$, $\hat{\beta}_j = (\hat{\boldsymbol{A}} \hat{w}_{j-1}, \hat{z}_j)/(\hat{\boldsymbol{A}} \hat{w}_{j-1}, \hat{w}_{j-1}) = \beta_j$ and $\hat{w_j} = \boldsymbol{L}^T w_j$. Proposition 1 provides the inequality concerning the asymptotic convergence rate. □

Putting these propositions together, APCG($\boldsymbol{A}, \boldsymbol{M}, \boldsymbol{C}$, b) is equivalent to CG($\boldsymbol{L}^{-1}\boldsymbol{P}^T \boldsymbol{A} \boldsymbol{P} \boldsymbol{L}^{-T}$, $\boldsymbol{L}^{-1}\boldsymbol{P}^T b$). All these results lead us to propose an efficient augmentation by analogy with an equivalent, simpler system solved by classical conjugate gradients.

2.2 Interpretation and choice of the augmentation

From a "constraint" point of view, the projection guarantees the $\boldsymbol{C}$-orthogonality of the residual throughout the iterations ($\boldsymbol{C}^T r_j = 0$). For example, in the FETI domain decomposition method, the residual is the displacement jump between the subdomains; in the case of shell and plate problems, matrix $\boldsymbol{C}$ is introduced to enforce the continuity of the displacement at the corner points [12]. In the BDD domain decomposition method, matrix $\boldsymbol{C}$ is associated with the rigid body motions of floating substructures and, therefore, local Neumann problems in the preconditioner are always well-posed [29]; for shell and plate problems, the matrix is enriched by corner mode corrections [26]. In both cases, matrix $(\boldsymbol{C}^T \boldsymbol{A} \boldsymbol{C})^{-1}$, called a coarse grid matrix, plays a crucial role in the scalability of these methods.

From a "spectral" point of view, augmentation can be used to decrease the active condition number ("active" referring to eigenelements solicited by the right-hand side) and, thus, improve the asymptotic convergence rate. This is called a deflation strategy [8, 5], which boils down to building matrix $\boldsymbol{C}$ by using (approximate) eigenvectors associated with the lowest eigenvalues. Obviously, when $\boldsymbol{C}$ consists of the n_c eigenvectors associated with the lowest eigenvalues ($\lambda_1 \leqslant \ldots \leqslant \lambda_{n_c} \leqslant \ldots \leqslant \lambda_n$), the condition number decreases strictly: $\kappa\left(\boldsymbol{P}^T \boldsymbol{A} \boldsymbol{P}_{\boldsymbol{A}\mathcal{V}}\right) = \frac{\lambda_n}{\lambda_{n_c}} < \frac{\lambda_n}{\lambda_1}$.

From a "model reduction" point of view, subspace $\mathcal{C}$ represents a "macro" (or coarse) space in which the macro part of the solution is calculated directly during the initialization while the "micro" part of the solution, when required, is obtained during the iterations.

2.3 Estimation of computation costs

With regard to the numerical cost of augmentation, the main operations for the construction of the projector are: (i) the block product $\boldsymbol{A}\boldsymbol{C}$ (and assembly with neighbors for domain decomposition methods), (ii) the block dot-product ($\boldsymbol{C}^T \boldsymbol{A} \boldsymbol{C}$) (plus an all-to-all sum for domain decomposition methods), and (iii) the factorization of the fully-populated coarse matrix ($\boldsymbol{C}^T \boldsymbol{A} \boldsymbol{C}$). Then, the application of the projector consists simply of (i) one block dot-product ($(\boldsymbol{A}\boldsymbol{C})^T x$) (plus an all-to-all exchange), (ii) the resolution of the coarse problem, and (iii) the matrix-vector product ($\boldsymbol{C}\alpha$).

Thus, provided that the number of columns of matrix $\boldsymbol{C}$ is small, the main cost is related to the calculation of $\boldsymbol{A}\boldsymbol{C}$. One must bear in mind that block operations (on "multivectors") are comparatively much faster than single vector operations, especially when the matrices are sparse (because data fetching is factorized). In a

domain decomposition context, product $\boldsymbol{AC}$ corresponds to the resolution of Dirichlet or Neumann problems in substructures, which makes the simultaneous treatment of many columns very efficient (and minimizes the number of exchanges). One must also remember that a conjugate gradient iteration involves a preconditioning step which may be expensive. (The cost is comparable to that of an operator product in optimal domain decomposition methods.) Thus, the additional cost of augmentation relative to the cost of one iteration depends on many parameters (the size of the problem, the number of augmentation vectors, the number of subdomains, the preconditioner chosen...). Typically, in the examples presented in this paper, we found that, using an optimal preconditioner, the CPU cost of between 4 and 7 augmentation vectors (depending on the hardware configuration) cost no more than one CG iteration.

A question which is not addressed in this paper is the verification of the full-rank property of matrix $\boldsymbol{C}$, which affects the quality of the factorization of matrix $(\boldsymbol{C}^T\boldsymbol{AC})$. Strategies to correct a dependence among the columns of matrix $\boldsymbol{C}$ due to inexact arithmetic can be found in [1].

3 Total reuse of Krylov subspaces

In this section, we show how it is possible to define efficient augmentation strategies in a multiresolution context. Let us consider the sequence of linear systems:

$$\boldsymbol{A}^{(k)}x^{(k)} = b^{(k)} \quad , \quad k = 1, \dots p \tag{11}$$

where $A^{(k)}$ is an $n \times n$ symmetric positive definite matrix and $b^{(k)}$ is the right-hand side. Each linear system is solved using an augmented preconditioned conjugate gradient algorithm APCG($\boldsymbol{A}^{(k)}$, $\boldsymbol{M}^{(k)}$, $\boldsymbol{C}^{(k)}$, $b^{(k)}$). Let $m^{(k)}$ be the number of iterations which is necessary to reach convergence, and let

$$\boldsymbol{W}_m^{(k)} = \left[w_0^{(k)}, \dots, w_{m^{(k)}-1}^{(k)}\right] \tag{12}$$

be a basis of the associated Krylov subspace.

As explained in the previous section, augmentation never increases the condition number which governs the asymptotic convergence rate. More precisely, the presence of active eigenvectors of the current preconditioned problem in $\boldsymbol{C}^{(k)}$ may increase the efficiency of the iterative solver significantly. Classical strategies can be used in the case of invariant preconditioned operators ($\boldsymbol{A}^{(k)} = \boldsymbol{A}$, $\boldsymbol{M}^{(k)} = \boldsymbol{M}$) and multiple right-hand sides.

It is more difficult to define efficient strategies in the general case of varying operators with no information available on their evolution. A simple and natural idea is to reuse previous Krylov subspaces. A first algorithm which reuses all the previous Krylov subspaces is Total Reuse of Krylov Subspaces (TRKS) (Algorithm 2), which needs only a few comments:

- Since (according to (8)) ${\boldsymbol{C}^{(k)}}^T\boldsymbol{A}^{(k)}\boldsymbol{W}_m^{(k)} = 0$, the vectors of the concatenated matrix $\boldsymbol{C}^{(k+1)}$ are linearly independent. Therefore, APCG($\boldsymbol{A}^{(k+1)}$, $\boldsymbol{M}^{(k+1)}$, $\boldsymbol{C}^{(k+1)}$, $b^{(k+1)}$) does not break down and converges.
- The previous Krylov subspaces are fully reused through concatenation without post-processing; the only downside is that the memory requirements increase due to the need to save the Krylov subspaces.
- If the number of columns of matrix $\boldsymbol{C}^{(k)}$ becomes too large, the method may become computationally inefficient, even though the number of iterations decreases considerably. Nevertheless, TRKS probably leads to the best reduction in the number of iterations achievable by reusing Krylov subspaces. Therefore, it can be used as a reference in terms of the reduction of the number of iterations for any other algorithm based on a reuse of Krylov subspaces.
- One possible way to reduce the cost of TRKS without reducing the size of $\boldsymbol{C}$ consists in using approximate solvers, as in the IRKS strategy [41].

In order to reduce the cost associated with the total reuse of Krylov subspaces, we propose to work on extracted sub-subspaces, an operation often referred to as the recycling of Krylov subspaces. The objective is to retain the smallest number of independent vectors which achieve the greatest decrease in the number of iterations. Clearly, the most effective approach would be to calculate approximate eigenvectors from the previous Krylov subspaces for the current operator. However, because of the variability of the operators, the extraction of such information would be extremely time consuming and would affect the global efficiency. Conversely, approximate eigenvectors of previous problems can be calculated from the associated Krylov subspaces at nearly

Algorithm 2: TRKS-APCG

Initialize $\boldsymbol{C}^{(0)} = \boldsymbol{C}_0$ (an $n \times m_0$ full-rank matrix);
for $k = 0, \ldots, p-1$ **do**
  Solve $\boldsymbol{A}^{(k)} x^{(k)} = b^{(k)}$
    with APCG($\boldsymbol{A}^{(k)}$, $\boldsymbol{M}^{(k)}, \boldsymbol{C}^{(k)}$, $b^{(k)}$);
  Define $\boldsymbol{W}_m^{(k)} = \left[\ldots, w_j^{(k)}, \ldots\right]_{0 \leqslant j < m^{(k)}}$;
  Concatenate: $\boldsymbol{C}^{(k+1)} = \left[\boldsymbol{C}^{(k)}, \boldsymbol{W}_m^{(k)}\right]$
end

no cost. In the following section, we describe an efficient algorithm for the extraction of such approximation vectors along with a simple selection procedure to recycle only a few of these vectors. Of course, the performance of our method depends on the stability of the eigenspaces from one system to another. This topic, especially concerning the lower part of the spectrum, is discussed in [24].

4 Selective recycling of Krylov subspaces

The standard convergence of conjugate gradients corresponds to an asymptotic convergence rate. Using this property to predict the number of iterations n_{ϵ} which is required to reach an accuracy level ϵ_{cg} leads to a huge overestimation. Indeed, one has:

$$\begin{aligned} &\frac{\|x_i - x\|_{\mathbb{A}}}{\|x_0 - x\|_{\mathbb{A}}} \leqslant 2(\sigma_{1,n})^i \leqslant \epsilon_{cg} \quad \Rightarrow \quad i \geqslant n_{\epsilon} = \frac{\ln(\epsilon_{cg}/2)}{\ln(\sigma_{1,n})} \\ &\text{with } \sigma_{r,s} = \frac{\sqrt{\kappa_{r,s}} - 1}{\sqrt{\kappa_{r,s}} + 1} \text{ and } \kappa_{r,s} = \frac{\lambda_r}{\lambda_s} \end{aligned} \tag{13}$$

This result alone cannot explain the improvement in the convergence rate observed during the iteration process. This superconvergence phenomenon can be explained by a study of the convergence of Ritz values [33] which enables one to define an instantaneous convergence rate [49]. This explanation can be improved by a study of the influence of the distribution of the eigenvalues [35, 2].

The objective of recycling Krylov subspaces is to find the best augmentation space in order to trigger superconvergence quickly. This section is organized as follows: we start with a review of Ritz eigenelement analysis and continue with a brief presentation of the improved convergence results; these results lead to a number of selection strategies, which will be assessed in Section 5.

4.1 Ritz analysis: theory and practical calculation

For $0 \leqslant i < m$, Ritz vectors ($\hat{y}_m^i$) and values (θ_m^i) are approximations of the eigenvectors and eigenvalues of the symmetric positive definite matrix $\hat{\boldsymbol{A}}$; their definition is similar to that of the iterates in the conjugate gradient algorithm (4)

$$\left\{ \begin{array}{ll} \text{find} & (\hat{y}_m^i, \theta_m^i) \in \mathcal{K}_m(\hat{\boldsymbol{A}}, \hat{v}_0) \times \mathbb{R} \\ \text{such that} & \hat{\boldsymbol{A}} y_m^i - \theta_m^i y_m^i \perp \mathcal{K}_m(\hat{\boldsymbol{A}}, \hat{v}_0) \end{array} \right. \tag{14}$$

The symmetric Lanczos algorithm [45] enables one to build a particular orthonormal basis of $\mathcal{K}_m(\hat{\boldsymbol{A}}, \hat{v}_0)$, denoted $\boldsymbol{V}_m$. Then, the search principle becomes:

$$y_m^i = \hat{\boldsymbol{V}}_m q_m^i, \quad \hat{\boldsymbol{V}}_m^T \hat{\boldsymbol{A}} \hat{\boldsymbol{V}}_m q_m^i = \theta_m^i q_m^i \tag{15}$$

The Lanczos basis $\hat{\boldsymbol{V}}_m$ makes the Hessenberg matrix $\hat{\boldsymbol{H}}_m = \hat{\boldsymbol{V}}_m^T \hat{\boldsymbol{A}} \hat{\boldsymbol{V}}_m$ symmetrical and tridiagonal. $\hat{\boldsymbol{V}}_m$ and $\hat{\boldsymbol{H}}_m$ can be recovered directly from the conjugate gradient coefficients [44]:

$$\begin{aligned} &\left\{ \begin{array}{l} \hat{\boldsymbol{V}}_m = \left(\ldots, (-1)^j \frac{\hat{r}_j}{\|\hat{r}_j\|}, \ldots\right)_{0 \leqslant j < m} \\ \hat{\boldsymbol{H}}_m = \text{tridiag}(\eta_{j-1}, \delta_j, \eta_j)_{0 \leqslant j < m} \end{array} \right. \\ &\text{with } \delta_0 = \frac{1}{\alpha_0}, \ \delta_j = \frac{1}{\alpha_j} + \frac{\beta_{j-1}}{\alpha_{j-1}}, \ \eta_j = \frac{\sqrt{\beta_j}}{\alpha_j} \end{aligned} \tag{16}$$

Since matrix $\hat{\boldsymbol{H}}_m$ is symmetrical and tridiagonal, its eigenelements $(\theta_j^m, q_j^m)_{1\leqslant j\leqslant m}$ can be calculated easily, for example using a Lapack procedure. Let us define $\boldsymbol{\Theta}_m = \mathrm{diag}(\theta_m^1 \leqslant \ldots \leqslant \theta_m^m)$ and $\boldsymbol{Q}_m = \left[q_m^1, \ldots, q_m^m\right]$ such that $\hat{\boldsymbol{H}}_m = \boldsymbol{Q}_m \boldsymbol{\Theta}_m {\boldsymbol{Q}_m}^T$. $\boldsymbol{\Theta}_m$ and $\hat{\boldsymbol{Y}}_m = \hat{\boldsymbol{V}}_m \boldsymbol{Q}_m$ are the Ritz values and associated Ritz vectors, which are approximations of the eigenelements of operator $\hat{\boldsymbol{A}}$ and satisfy:

$$\hat{\boldsymbol{Y}}_m^T \hat{\boldsymbol{A}} \hat{\boldsymbol{Y}}_m = \boldsymbol{\Theta}_m \qquad \text{and} \qquad \hat{\boldsymbol{Y}}_m^T \hat{\boldsymbol{Y}}_m = \boldsymbol{I}_m$$

We presented Ritz analysis for the equivalent symmetric system described previously because symmetry simplifies the calculation of eigenelements, but the analysis can be transferred back to the left-preconditioned system using the following transformation rules:

$$\begin{aligned} \boldsymbol{V}_m &= \boldsymbol{L}^{-T} \hat{\boldsymbol{V}}_m = \left[\ldots, (-1)^j \frac{z_j}{(r_j, z_j)^{1/2}}, \ldots\right] \\ \boldsymbol{H}_m &= \hat{\boldsymbol{H}}_m = \boldsymbol{V}_m^T \boldsymbol{A} \boldsymbol{V}_m \\ \boldsymbol{Y}_m &= \boldsymbol{L}^{-T} \hat{\boldsymbol{Y}}_m = \boldsymbol{V}_m \boldsymbol{Q}_m \end{aligned} \tag{17}$$

The Ritz vectors are the solution of a generalized eigenproblem and satisfy the following orthogonality properties:

$$\boldsymbol{Y}_m^T \boldsymbol{A} \boldsymbol{Y}_m = \boldsymbol{\Theta}_m \qquad \text{and} \qquad \boldsymbol{Y}_m^T \boldsymbol{M} \boldsymbol{Y}_m = \boldsymbol{I}_m \tag{18}$$

One can show that when m increases the Ritz values converge toward the eigenvalues of $\hat{\boldsymbol{A}}$, and that the convergence is either from above or from below depending on their rank [49, 51]:

$$\theta_m^1 \geqslant \theta_{m-1}^1 \geqslant \theta_m^2 \geqslant \ldots \geqslant \theta_m^{m-1} \geqslant \theta_{m-1}^{m-1} \geqslant \theta_m^m \tag{19}$$

In addition, in the case of clearly distinct eigenvalues, the convergence of a Ritz value results in the convergence of the associated Ritz vector.

4.2 Relation between the convergence of conjugate gradients and the convergence of the Ritz values

In [49], the superconvergence phenomenon is explained by the convergence of the Ritz values through the definition, at each iteration, of a instantaneous convergence rate associated with the part of the spectrum that is not yet approximated correctly by the Ritz values: at a given conjugate gradient iteration, one can find a deflated system (with some of its extreme eigenvalues removed) with similar behavior. Let $[\lambda_l, .., \lambda_r]$ be the spectrum of the deflated operator. The equivalent convergence rate is:

$$\|x - x_{i+1}\|_{\mathbb{A}} \leqslant \; F_{i,l,r} \; 2 \; \sigma_{l,r} \|x - x_i\|_{\mathbb{A}} \tag{20}$$

where $F_{i,l,r}$ quantifies the convergence of the l smallest and r largest Ritz values to the extreme eigenvalues:

$$\begin{aligned} F_{i,l,r} &= \max_{l'>l} J_{l,l'}^{(i)} \; \max_{r'\geqslant r} L_{r,r'}^{(i)} \\ J_{l,l'}^{(i)} &= \prod_{j=1}^{l} \left|1 - \frac{\lambda_{l'}}{\lambda_j}\right| \left|1 - \frac{\lambda_{l'}}{\theta_j^i}\right|^{-1} \\ L_{r,r'}^{(i)} &= \prod_{j=1}^{r} \left|1 - \frac{\lambda_{n-r'}}{\lambda_{n+1-j}}\right| \left|1 - \frac{\lambda_{n-r'}}{\theta_{i+1-j}^i}\right|^{-1} \end{aligned}$$

Since this result holds for every pair (l, r), the effective convergence rate at Iteration i corresponds to the pair (l, r) which minimizes $\sigma_{i,l,r} = F_{i,l,r}\sigma_{l,r}$.

Then, after some iterations, the superconvergent conjugate gradient algorithm behaves very much like a conjugate gradient algorithm augmented by the extreme eigenvectors which are associated with the converged Ritz values. In a multiresolution context, provided the linear systems have similar spectral properties, the Ritz vectors associated with the converged Ritz values obtained for one system should define a viable augmentation space for the subsequent resolutions.

4.3 Effect of the distribution of the eigenvalues

The effect of the distribution of the eigenvalues on the convergence of conjugate gradients was studied in [35, 2]. The results take into account the fact that preconditioning often leads to clustered eigenvalues as opposed to uniformly distributed eigenvalues, as can be seen in Figure 1.

In addition to other results, the authors showed that if a spectrum consists of p isolated eigenvalues in the high part of the spectrum, p isolated eigenvalues in the low part of the spectrum and $n-2p$ uniformly distributed central eigenvalues, then the conjugate gradient convergence takes the form:

$$n_\epsilon \geqslant \tilde{n}_\epsilon = 2p + \text{int}\left(\frac{\ln\left(\epsilon_{cg}/2\right)}{\ln\ \sigma_{p+1,n-p}} - \frac{\sum_{i=1}^{p} \ln\left(\frac{\lambda_{n-p+i}}{4\lambda_i}\left(1-\frac{\lambda_i}{\lambda_{n-p+i}}\right)\right)}{\ln\ \sigma_{p+1,n-p}} \right) \tag{21}$$

The convergence rate is approximately equal to the classical convergence rate for the central part, plus one iteration per higher eigenvalue and a little more than one iteration per lower eigenvalue. These results can be combined with the work by Jiao [51, 22] on the convergence of Ritz values. In general, since the method is related to the power iteration method, a correct approximation by the Ritz values is obtained first for the highest eigenvalues, then for the lowest part of the spectrum, resulting in superconvergence (which is governed by the asymptotic convergence rate of the reduced spectrum).

4.4 Selection procedures

The results of Section 4.2 lead to a first proposal of a selection procedure for converged Ritz vectors: convergence is identified by the stagnation of the Ritz values; if the conjugate gradient algorithm converges at iteration m, the Ritz values are calculated for the previous two states $\boldsymbol{\Theta}_m$ and $\boldsymbol{\Theta}_{m-1}$. Once ranked, the m most recent Ritz values $\boldsymbol{\Theta}_m$ are compared to the $m-1$ previous values according to the following criteria:

$$\begin{cases} \theta_m^j \text{ has converged if } \dfrac{|\theta_m^j - \theta_{m-1}^j|}{|\theta_m^j|} \leqslant \varepsilon, \qquad 1 \leqslant j \leqslant m-1 \\ \theta_m^{m-j} \text{ has converged if } \dfrac{|\theta_m^{m-j} - \theta_{m-1}^{m-1-j}|}{|\theta_m^{m-j}|} \leqslant \varepsilon, \qquad 0 \leqslant j \leqslant m-2 \end{cases} \tag{22}$$

where ε is a user parameter which is easy to adjust since the criterion is generally either very high (before the convergence of the Ritz value) or very small (after convergence). Figure 1 illustrates that property with the simple example of the operator associated with the decomposition of a linear elastic cube into ten subdomains; in that case, the higher half of the spectrum has converged.

The principle of the selective recycling of Krylov subspaces (SRKS-APCG) is described in Algorithm 3. Basically, in addition to the memory required by APCG, the SRKS-APCG algorithm requires storage for m n-vectors $(z_j)_{j=1,m}$. One should note that the selected vectors are normalized by the square root of the associated Ritz value in order to improve the condition number of the coarse matrix. (If operator $\boldsymbol{A}$ remained constant, matrix $(\boldsymbol{C}^T\boldsymbol{A}\boldsymbol{C})$ would be the identity matrix.)

For better computational efficiency, a restart parameter can be introduced in order to limit the size of the augmentation space associated with parameter n_{clim} in Algorithm 3. This limit size can be set after a complexity analysis under the assumption that all non-augmented systems would be solved in the same number of iterations. However, we did not use such a restart procedure in our experiments.

In order to be even more selective, we propose a reselection strategy based on a prediction of the efficiency of the retained vectors. Indeed, the results of Section 4.3 in terms of the effect of the distribution of the eigenvalues lead us to retain only the converged Ritz vectors which belong to the external part of the spectrum:

- this is known to be the first part of the spectrum whose approximation by Ritz values is good;
- since the convergence of Ritz vectors is identified by the stagnation of the associated Ritz values, the fact that the external Ritz values are distinct ensures that the Ritz vectors approximate the eigenvectors correctly [51];
- while choosing vectors in the dense central zone does not modify the shape of the spectrum and does not improve convergence, selecting the external part of the spectrum triggers superconvergence instantly.

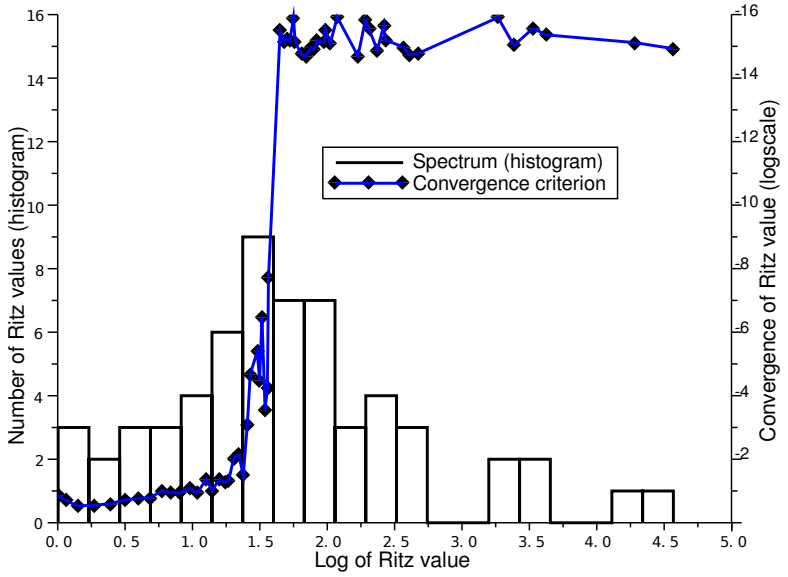


Figure 1: Ritz spectrum and convergence of the Ritz values

Algorithm 3: SRKS-APCG

Initialize $C^{(0)} = \boldsymbol{C}_0$ (full column rank matrix);
for $k = 0, \ldots, p-1$ **do**
- Solve $\boldsymbol{A}^{(k)} x^{(k)} = b^{(k)}$ with APCG$(\boldsymbol{A}^{(k)}, \boldsymbol{M}^{(k)}, \boldsymbol{C}^{(k)}, b^{(k)})$;
- Define $\boldsymbol{V}_m = \left[\ldots, (-1)^j \frac{z_j}{(r_j, z_j)^{1/2}}, \ldots\right]_{0 \leqslant j < m}$;
- Define $\boldsymbol{H}_m = \text{tridiag}(\eta_{j-1}, \delta_j, \eta_j)_{0 \leqslant j < m}$ $\delta_0 = \frac{1}{\alpha_0}$, $\delta_j = \frac{1}{\alpha_j} + \frac{\beta_{j-1}}{\alpha_{j-1}}$, $\eta_j = \frac{\sqrt{\beta_j}}{\alpha_j}$;
- Compute eigenelements $(\boldsymbol{Q}_m, \boldsymbol{\Theta}_m)$ of $\boldsymbol{H}_m$ $(\theta_m^1 \geqslant \ldots \geqslant \theta_m^m)$;
- Compute $\boldsymbol{Y}_m = \boldsymbol{V}_m \boldsymbol{Q}_m = \left[y_m^1, \ldots, y_m^m\right]$;
- Extract $\boldsymbol{H}_{m-1} = \text{tridiag}(\eta_{j-1}, \delta_j, \eta_j)_{0 \leqslant j < m-1}$;
- Compute eigenvalues (θ_{m-1}^j) of $\boldsymbol{H}_{m-1}$;

  for $j = 1, \ldots, m-1$ **do**
  - $\boldsymbol{C} = \left[\boldsymbol{C}, \frac{y_m^j}{\sqrt{|\theta_m^j|}}\right]$ if $|\theta_m^j - \theta_{m-1}^j| \leqslant \varepsilon |\theta_m^j|$;
  - $\boldsymbol{C} = \left[\boldsymbol{C}, \frac{y_m^{j+1}}{\sqrt{|\theta_m^{j+1}|}}\right]$ if $|\theta_m^{j+1} - \theta_{m-1}^j| \leqslant \varepsilon |\theta_m^{j+1}|$

  end
- Concatenate $\boldsymbol{C}^{(k+1)} = [\boldsymbol{C}^{(k)}; \boldsymbol{C}]$, $\boldsymbol{C} = [0]$;
- If $\dim(\boldsymbol{C}^{(k+1)}) \geqslant n_{c_{lim}}$, then $\boldsymbol{C}^{(k)} = \boldsymbol{C}^{(0)}$

end

In order to select only the external part of the spectrum, we implemented the cluster identification algorithm proposed in [32]. This algorithm seeks the piecewise constant distribution which is nearest (in a least squares sense) to the distribution of the distances among the sorted eigenvalues. The only parameter required is the minimum size of the cluster, which we set at one-fifth the number of preselected vectors. As will be shown in the next section, the performance achieved with this reselection algorithm is not outstanding, but some results in terms of gain per augmentation vector are worth considering.

5 Numerical assessments

We present three numerical experiments. Two concern the evaluation of a structure made of random materials, as is the case in a Monte-Carlo simulation. In the first case, the materials are elastic; in the second case, which is a nonlinear problem, they are elastic-plastic. The last case is a large displacement problem, which raises specific difficulties.

The methods were implemented in the ZEBULON code [34] and parallelism was introduced using MPI. The calculations were performed on the LMT-Cachan cluster, which consists of dual quadcore and dual hexacore processors connected by a gigabit network. The calculations were always carried out on homogeneous sets of processors which were entirely dedicated to one task which fit entirely in memory, so swapping was not necessary. In each case, we indicate the CPU time which measures the amount of work performed for one subdomain. The Wall Clock Time (WCT), a global measure which is more sensitive to external perturbations induced by the operating system and the presence of other users, was considered to be unreliable in many cases; so we mention it only for the first set of experiments. One should note that the gains calculated with WCT were always greater.

The CPU plots show the total time as well as the time dedicated to augmentation (preparation of the coarse operator, initialization and projections); the difference represents the iterations of the solver.

All the calculations used a dual formulation of the interface problem through domain decomposition (FETI). The convergence was evaluated using the norm of the residual (which corresponds to the displacement gaps at the interfaces) normalized by the condensed right-hand side. Classically for such structural problems, total reorthogonalization was used to enforce the $\boldsymbol{A}$-conjugation of the search directions. (The case without reorthogonalization is discussed briefly in the first example.)

5.1 The case of a sequence of linear systems

We considered a cube (of side 50 mm) with $4 \times 4 \times 4 = 64$ small cubic inclusions (of side 5.5 mm). A slice through this structure is shown in Figure 2. The cube was clamped over one side, and the opposite side plus another side were subjected to uniform pressure. The mesh consisted of $125,000$ linear hexahedral elements for a total of $400,000$ degrees of freedom. Three automatic decompositions (into 12, 48 and 96 subdomains) were performed using the Metis algorithm [23] (see Figure 3). The resulting interface system contained $54,000$ unknowns for the 12-subdomain decomposition, $96,000$ unknowns for the 48-subdomain decomposition and $133,000$ unknowns for the 96-subdomain decomposition. All the materials were isotropic, linear and elastic, and were characterized by their Young's modulus and Poisson's coefficient. The material properties of each inclusion and of the matrix were chosen randomly following a normal law with a relative standard deviation equal to 10%, leading to a $\pm 23\%$ variation range about the nominal value. The average Young's modulus was 200 MPa for the matrix and $20,000$ MPa for each inclusion, and the average Poisson's coefficient was 0.27 for the matrix and 0.35 for each inclusion. The objective was to perform the calculations for 40 draws of the 130 coefficients.

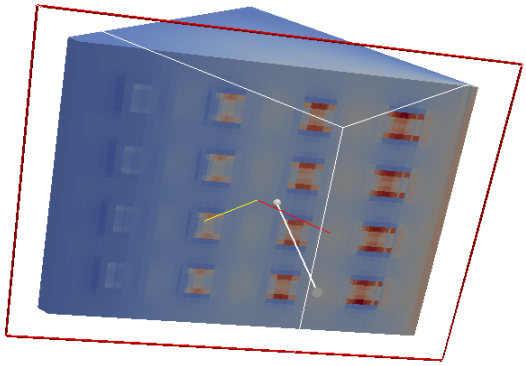


Figure 2: A slice through the heterogeneous cube (shear stress)

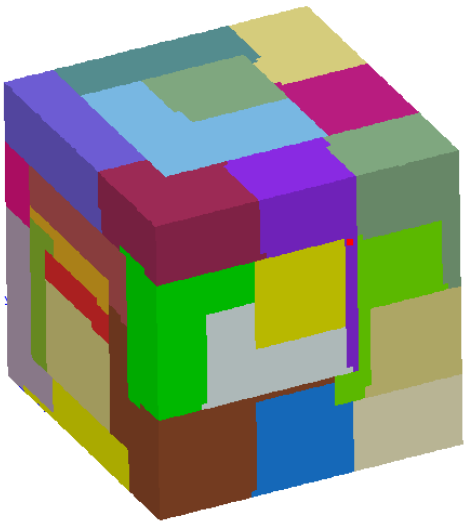


Figure 3: The decomposition into 48 subdomains

			12 subdomains						
accur.	precond.		avg. # it	avg. n_c	max n_c	avg. total CPU	avg. CPU aug.	avg. total WCT	avg. WCT aug.
10^{-3}	Dirichlet	cg (no reo.)	70	—	—	23.5	0	31.8	0
		cg	44.3	—	—	16.0	0	24.5	0
		trks	2.4	77.8	96	7.1	6.3	9.1	7.6
		srks6	20.5	41.3	50	11.0	3.8	15.6	4.7
		srks14	25.6	24.6	27	**11.7**	2.7	**17.1**	3.5
		clust14	**30.6**	16.8	24	13.8	2.2	25.3	3.2
	Lumped	cg (no reo.)	145	—	—	27.7	0	40.3	0
		cg	68.1	—	—	13.6	0	24.0	0
		trks	**0.4**	71.8	74	**5.7**	5.6	**6.8**	6.7
		srks6	27.4	81	108	12.3	6.5	18.0	8.0
		srks14	32.0	59.6	71	11.8	5.1	18.1	6.4
		clust14	51.1	20	39	12.8	2.3	21.7	3.2
10^{-6}	Dirichlet	cg (no reo.)	174	—	—	58.3	0	85.8	0
		cg	84.4	—	—	30.0	0	49.1	0
		trks	13.2	382.9	551	46.5	41.3	60.4	52.8
		srks6	36.7	104.3	142	22.3	8.6	35.7	11.0
		srks14	42.8	72.6	87	22.7	6.3	36.2	8.1
		clust14	**60.8**	35.2	77	**25.7**	3.6	**39.8**	4.9
	Lumped	cg (no reo.)	>400	—	—	>78	0	>110	0
		cg	147.7	—	—	31.7	0	61.0	0
		trks	16.3	516.7	735	70.2	65.7	93.3	85.9
		srks6	54.2	225.7	311	33.4	20.5	49.4	25.6
		srks14	60.6	170.2	216	29.2	15.0	44.2	18.3
		clust14	**129.4**	20	39	**30.4**	2.6	**55.6**	4.0

Table 1: Performance summary for the cube with inclusions

We used a dual formulation (FETI) with both a Dirichlet (optimal) and a lumped preconditioner, leading to 10^{-3} and 10^{-6} APCG accuracy respectively. We considered the following algorithms: conjugate gradients (cg), total reuse of Krylov subspaces (trks) and selective reuse of Krylov subspaces with two values of the criterion, $\varepsilon = 10^{-6}$ (srks6) and $\varepsilon = 10^{-14}$ (srks14). In addition, in the last case ($\varepsilon = 10^{-14}$), we also attempted to further refine the selection by not selecting the converged Ritz values contained in the central cluster (identified by the algorithm proposed by [32]); this method is labeled (clust14).

5.1.1 Comparison of the strategies

The results for the 12-subdomain decomposition are summarized in Table 1, which gives the average number of APCG iterations to convergence, the average size of the augmentation space, the final size of the augmentation space, the average CPU and wall clock times per system, from which we also deduced the augmentation time (operator preparation and projection). Computations were conducted one dual hexacore processor (one subdomain per core). When the average and final sizes of the augmentation space are close, this means that most of the augmentation space was identified with the first systems. For a given configuration (accuracy and preconditioner), the figures in bold in the three columns 'average number of iterations', 'average CPU time' and 'average wall clock time' indicate the best strategy in terms of gain per unit augmentation vector compared to CG.

For the 12-subdomain decomposition, Figures (4, 6, 8) (for an objective of 10^{-3} accuracy) and Figures (5, 7, 9)) (for an objective of 10^{-6} accuracy) give the evolutions of the number of APCG iterations to convergence for each linear system, the dimension of the augmentation space n_c and the the CPU time for the resolution of each system, with both lumped and Dirichlet preconditioners.

Without full reorthogonalization, the performance was very poor and led to about twice the number of

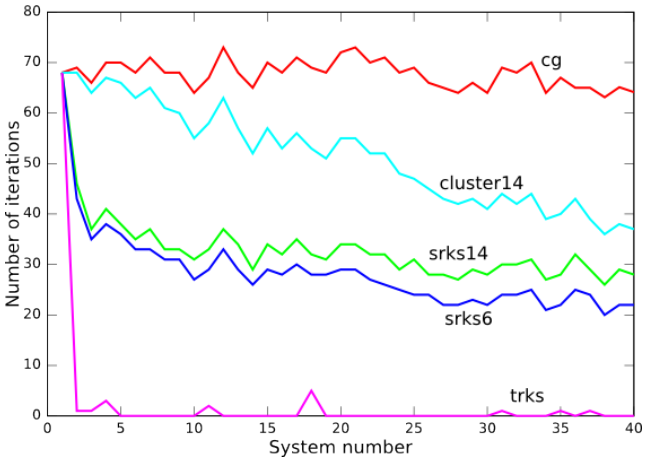


Figure 4: Cube 12 subdomains, lumped, 10^{-3}*accuracy*, number of iterations per linear system

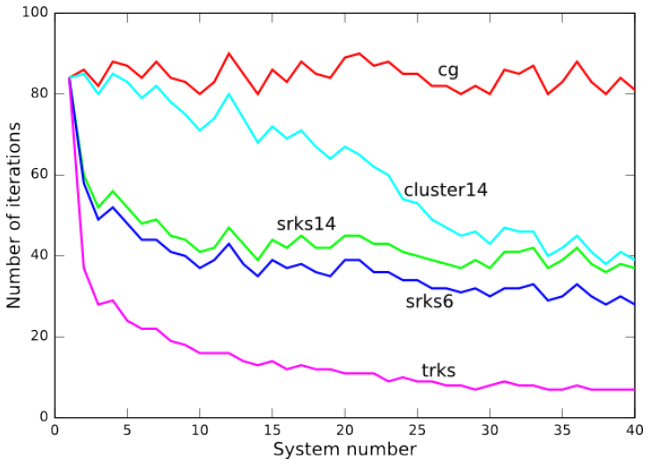


Figure 5: Cube 12 subdomains, Dirichlet, 10^{-6}*accuracy*, number of iterations per linear system

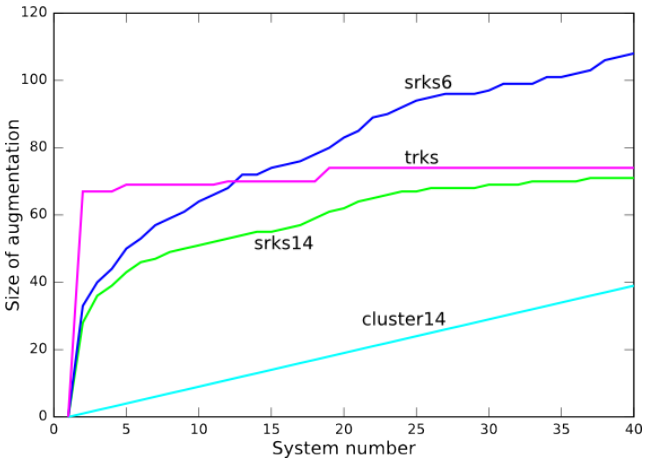


Figure 6: Cube 12 subdomains, lumped, 10^{-3}*accuracy*, dimension of the augmentation space

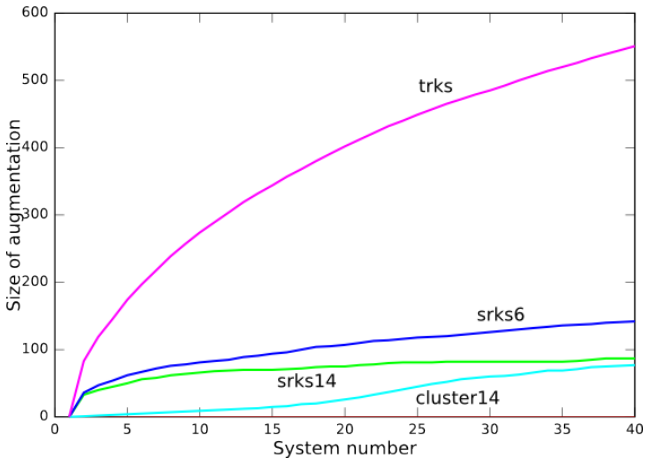


Figure 7: Cube 12 subdomains, Dirichlet, 10^{-6}*accuracy*, dimension of the augmentation space

iterations of the recommended fully reorthogonalized conjugate gradients. This was expected because systems resulting from domain decomposition formulations are known to often require full reorthogonalization [13]. Furthermore, one should note that the additional iterations carried out in the non-reorthogonalized case led to vector sets which made the Ritz analysis more complex due to the appearance of nonphysical, multiple eigenvalues. The non-reorthogonalized approach was no longer considered in the other examples.

With the TRKS approach, two types of behavior were observed. In the low-accuracy case (10^{-3}), for both preconditioners (but especially for the lumped preconditioner), the size of the augmentation space reached a plateau, which means that the augmentation space contained almost all the required information; the gains in terms of both the number of iterations ($> 90\%$) and the CPU time ($> 55\%$) were excellent. In the high-accuracy case (10^{-6}), the size of the augmentation space never stabilized; therefore, even though the number of iterations decreased drastically, the CPU time increased. Table 2 gives extended performance results for TRKS which confirm this analysis. The gains are given relative to conjugate gradients. The efficiency of augmentation is defined by the average decrease in the number of iterations per augmentation vector; the higher the required accuracy, the less efficient the TRKS approach. These results justify our decision to select the subspaces so that the dimension of the augmentation space would remain under control.

The SRKS14 approach succeeded in limiting the size of the augmentation space and led to a satisfactory decrease in the number of iterations. As can be seen on the figures, SRKS6 did not stabilize the augmentation space as efficiently and behaved half way between TRKS and SRKS14; therefore, we will choose SRKS14 as our reference algorithm from now on.

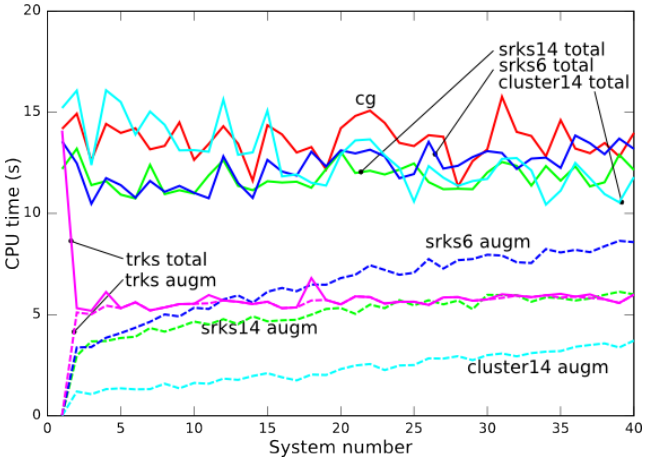


Figure 8: Cube 12 subdomains, lumped, $10^{-3} accuracy$, CPU time per linear system

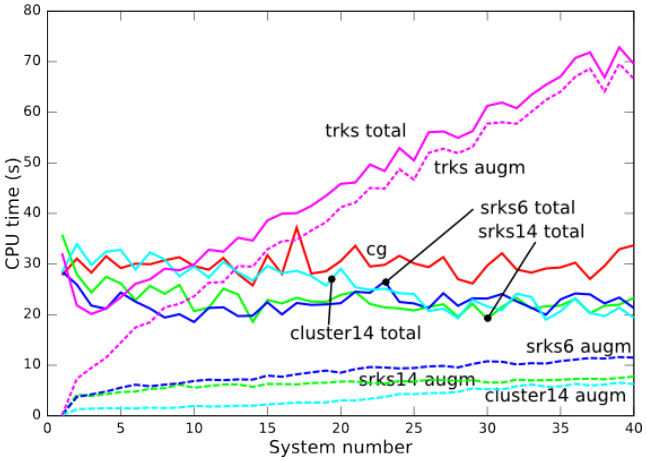


Figure 9: Cube 12 subdomains, Dirichlet, $10^{-6} accuracy$, CPU time per linear system

The cluster strategy as it stands today gave unsatisfactory results: even though it often led to the best gain per augmentation vector, it seemed to impair the selection of useful vectors and allow much less reduction in the number of iterations than SRKS. After the resolution of many systems, it tended to lead to the same augmentation space as SRKS.

To confirm that hypothesis, we compared the spaces C_{SRKS} and $C_{cluster}$ after the 40 resolutions for the low-accuracy Dirichlet case. We used the following procedure: first, the vectors were orthonormalized using SVD: $C = U\Sigma V^T$; then SVD was applied to the concatenated matrix $[U_{SRKS}, U_{cluster}]$. A plot of the singular values is shown in Figure 10. Independent spaces would lead to a constant value equal to 1, while for nested spaces the common space would lead to $\{\sqrt{2}, 0\}$ pairs of singular values. One can observe that the spaces are not exactly nested, but come quite close.

In conclusion, the cluster strategy is not mature yet, but it is promising. It was not considered for the following experiments because, due to the larger number of systems involved, it would behave quite similarly to SRKS.

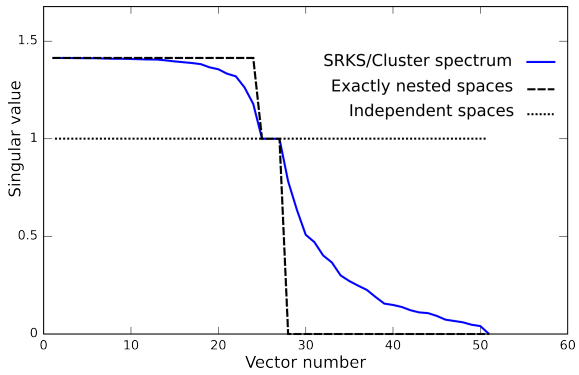


Figure 10: Singular values of $[C_{SRKS}, C_{cluster}]$

5.1.2 Study of SRKS14 in various configurations

Table 3 shows the relative performance of SRKS14 as a function of the number of subdomains, of the preconditioners and of the accuracy. The efficiency of the augmentation is defined as the decrease in the average number of iterations per augmentation vector. One can observe that the efficiency ranged between 0.5 and 0.85 and was best for the lower accuracy and the improved preconditioner. For these spectra in which there exist no small isolated eigenvalues (which could lead to efficiencies greater than 1), such results are consistent

accur.	precond.	# subdomains	iteration gain	CPU gain	efficiency of augmentation
10^{-3}	lumped	12	99.4%	58.2%	0.94
		48	97.7%	60.1%	0.97
	Dirichlet	12	94.7%	55.5%	0.54
		48	96.2%	62.6%	0.69
10^{-6}	lumped	12	89%	-121.2%	0.25
		48	90.2%	-162.3%	0.37
	Dirichlet	12	84.4%	-55.1%	0.19
		48	83.3%	-90.7%	0.21

Table 2: Relative performance of TRKS

with the theory (see Section 4.3). In the next section, we will see that this moderate efficiency does not preclude significant CPU improvements.

The gains in terms of the number of iterations were relatively stable, typically between 50% and 60% in the high-accuracy case.

precond.	accur.	# subdomains	CG avg. # iterations	avg. n_c	iteration gain	efficiency of augmentation
lumped	10^{-3}	12	68.1	59.6	52.9%	0.6
		48	43.7	28.2	54.2%	0.84
	10^{-6}	12	147.7	170.2	59%	0.51
		48	162.7	186.3	62.5%	0.55
Dirichlet	10^{-3}	12	43.3	24.6	42.2%	0.76
		48	50.7	31	48.6%	0.79
		96	67.4	51.1	51.7%	0.68
	10^{-6}	12	84.4	72.6	49.3%	0.57
		48	116.	111.5	57.2%	0.6
		96	140.7	141.9	60.3%	0.6

Table 3: Iteration gains for SRKS14

5.1.3 Influence of the hardware configuration on the CPU gains

Now, let us study the performance of SRKS14 in terms of CPU time for the same decomposition into 48 subdomains, but using different hardware configurations:

1. Configuration A corresponds to 4 dual hexacore nodes with 1 subdomain per core;
2. Configuration B corresponds to 6 dual quadcore nodes with 1 subdomain per core;
3. Configuration C corresponds to 3 dual quadcore nodes with 2 subdomains per core;
4. Configuration D corresponds to 2 dual quadcore nodes with 3 subdomains per core.

One can note that the processors in Configuration A were different from those used in the other cases. In all the cases, the memory was sufficient to avoid swapping. The results are given in Table 4 for the Dirichlet preconditioner and in Table 5 for the lumped preconditioner. One can see that Configurations B,C and D had similar performances and were slower than Configuration A due to the different memory technology.

One interesting factor is the ratio of the average CPU cost of an iteration to the average CPU cost of an augmentation vector (the last columns of Table 4 and 5). One can see that in Configuration A, 4 augmentation vectors cost no more than one iteration; in the other configurations 7 augmentation vectors cost no more than one iteration. Since we saw that one needs about $1/0.6 \simeq 1.6$ augmentation vectors to save one iteration, the

advantage of augmentation is clear. Indeed, we observe a 32% CPU improvement in Configuration A and a 40% to 50% improvement in the other configurations.

Note that when the lumped preconditioner is used the equivalent cost of an iteration is only 2.8 augmentation vectors in Configuration A and 4.5 augmentation vectors in Configuration D (see Table 5). Since the efficiency of the augmentation vectors is less when this inexpensive preconditioner is used (in the high accuracy case), so is the CPU improvement.

Configuration	CG avg. CPU	CPU gain	CPU per iteration / CPU per augm. vector
A	9.5	32.6%	4
B	25.7	41.5%	6.7
C	29.7	48.9%	7.7
D	29.9	47.6%	7.8

Table 4: CPU performance of SRKS14 for 10^{-6} accuracy with the Dirichlet preconditioner

Configuration	CG avg. CPU	CPU gain	CPU per iteration / CPU per augm. vector
A	10.7	22.4%	2.8
D	27.4	33.3%	4.5

Table 5: CPU performance of SRKS14 for 10^{-6} accuracy with the lumped preconditioner

The ratio of the CPU time per iteration to the CPU time per augmentation vector for SRKS (Column 4 of the previous tables) turned out to be relatively stable for a given machine with a given preconditioner. This is due to the stability of the size of the augmentation space which prevented the cost from soaring (as would happen with TRKS). Thus, the CPU performance can be deduced from the iteration gains and the augmentation efficiency (see Table 3). For instance, the CPU gain for SRKS with the 96-subdomain decomposition was slightly greater than 50%.

5.2 The case of a sequence of nonlinear problems

Now let us consider a hexahedral holed plate ($10 \times 10 \times 0.2$ mm with a center hole of radius 1 mm, see Figure 11) subjected to unidirectional tension (a prescribed normal displacement). The plate was discretized into $61,000$ linear hexahedral elements for a total of $41,000$ degrees of freedom. The structure was divided into 8 subdomains using the Metis algorithm, which resulted in an interface system with $3,000$ unknowns. The problem was solved using one 8-core processor (one subdomain per core). Elastic-plastic behavior with nonlinear isotropic hardening and a Von Mises'-type plasticity criterion was assumed. Denoting σ the Cauchy stress tensor, $\epsilon(u)$ the symmetric gradient of the displacement field u, and $\mathcal{K}$ the Hooke tensor, the material law can be written as:

$$\left\{\begin{array}{l} \epsilon(u) = \epsilon^e + \epsilon^p, \qquad \sigma = \mathcal{K} : \epsilon^e \\ \text{if } f(\sigma) = 0 \text{ then } \dot{\epsilon}^p = \lambda f_{,\sigma} \\ \text{if } f(\sigma) \leqslant 0 \text{ then } \dot{\epsilon}^p = 0 \\ f(\sigma) = \sqrt{\frac{3}{2}\,\sigma : \sigma} - \left(R_0 + Q\left(1 - e^{-b\lambda}\right)\right) \end{array}\right. \tag{23}$$

The coefficients were assigned a normal law with a 10% relative standard deviation, which implied variations of up to $\pm 23\%$ in the coefficients. The mean values of the material parameters were: $E = 200,000$ MPa, $\nu = 0.3$, $R_0 = 300$ MPa, $b = 22$ and $Q = 170$ MPa. The loading was applied in two steps: first, a single increment to reach the elastic limit; then, 16 equal increments in order to multiply the prescribed displacement by 4. The objective of the study was to analyze 21 configurations.

Again, the linear solver used was FETI with a Dirichlet or lumped preconditioner. The accuracy objective for the linear systems was set at 10^{-6}. (The accuracy must be high for the nonlinear process to run well). Because of the approximations, not all the methods converged in the same number of Newton iterations; on average, one nonlinear analysis required the resolution of 95 tangent systems. Table 6 summarizes the performances of the various methods; Figure 12 shows the evolution of the average number of APCG iterations with the lumped preconditioner during the sequence of linear systems; Figure 13 shows the evolution of the size of the

augmentation space; Figures 14 and 15 show the evolutions of the average CPU time and wall clock time for the resolution of one linear system along with the evolution of the average augmentation time (operator creation and projection).

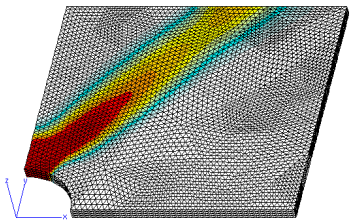


Figure 11: The holed plate example (plastic strain)

precond.	method	avg. # it	avg. n_c	max n_c	avg. CPU	avg. WCT
Dirichlet	cg	25.6	–	–	1.21	3.03
Dirichlet	trks*	1.4	358	492	2.66	9.83
Dirichlet	srks14	**16.1**	17	19	**0.98**	**2.35**
lumped	cg	41.4	–	–	1.03	3.24
lumped	trks*	1.2	520	695	4.72	6.98
lumped	srks14	**19.1**	43	45	**0.87**	**2.08**

* calculation too slow, was stopped before all the systems were solved

Table 6: Holed plate, performance summary

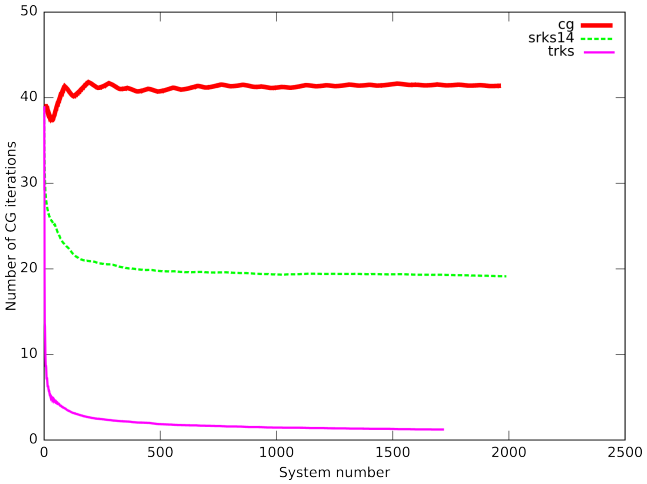


Figure 12: Plate, lumped – avg. # it. / linear system

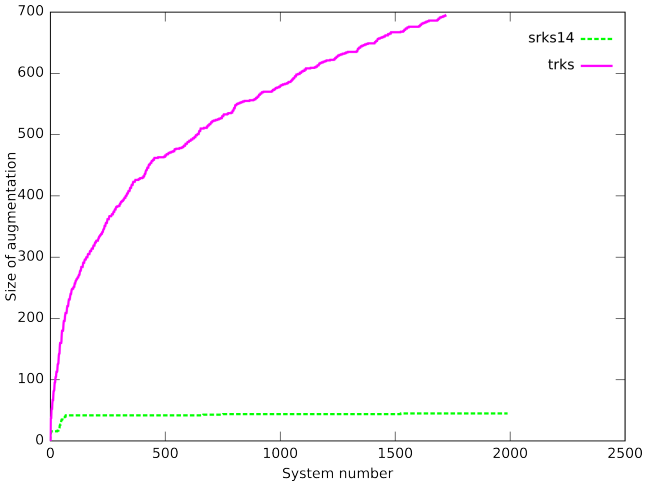


Figure 13: Plate, lumped – dimension of aug. space

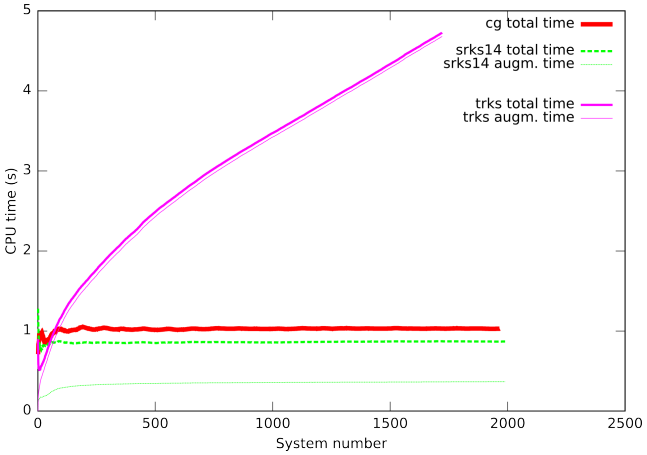


Figure 14: Plate, lumped – avg. CPU time / system

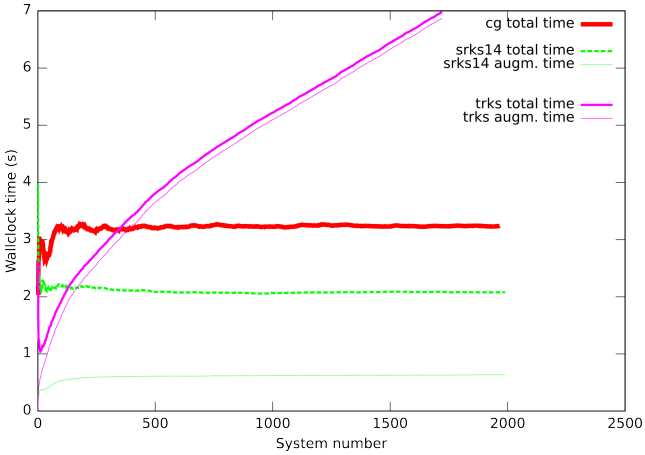


Figure 15: Plate, lumped – avg. wall clock time / system

This test leads to conclusions similar to the previous ones. More specifically, one can observe that the SRKS augmentation space selected after the first nonlinear configuration remained stable. Conversely, since the TRKS augmentation space never reached a plateau, the solutions of the linear systems did not belong to a common space. SRKS was the most efficient method, leading to a 20% CPU gain and a 36% wall clock time improvement.

5.3 The case of a large displacement problem

Finally, let us consider the problem of the buckling of a straight heterogeneous beam with a circular cross section (length/diameter ratio equal to 30), clamped at one end and subjected to an axial pressure at the other, with no radial displacement. The heterogeneities consisted of five straight fibers whose stiffness was $1,000$ times that of the matrix. The problem was formulated in the updated Lagrangian framework, assuming linear elastic behavior (characterized by the Young's modulus and Poisson's coefficient) in the current configuration. The beam was discretized into $90,000$ linear hexahedral finite elements for a total of $300,000$ degrees of freedom. It was divided into 10 subdomains using the Metis algorithm, leading to an interface system with $16,000$ unknowns. A single 12-core processor was used (1 subdomain per core, leaving 2 inactive cores). The pressure was applied incrementally up to the configuration shown in Figure 18, in which the maximum axial displacement was about 3% of the total length. 12 increments were used, leading to the resolution of about 30 tangent linear systems.

We used a FETI solver with a Dirichlet preconditioner and an "identity" projector. The FETI convergence criterion was set to 10^{-6}. Figure 16 shows the evolution of the number of conjugate gradient iterations required for the resolution of each linear system. Figure 17 shows the evolution of the size of the augmentation space. Three algorithms were tested: classical conjugate gradients, total reuse of subspaces, and selective reuse of

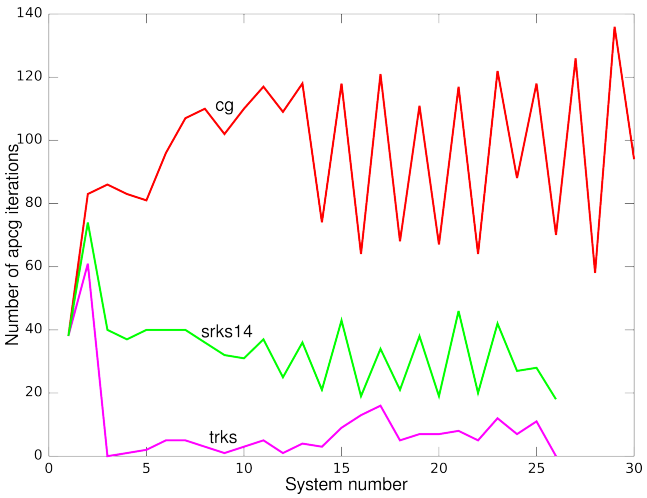


Figure 16: Buckling of the heterogeneous beam: number of iterations per linear system

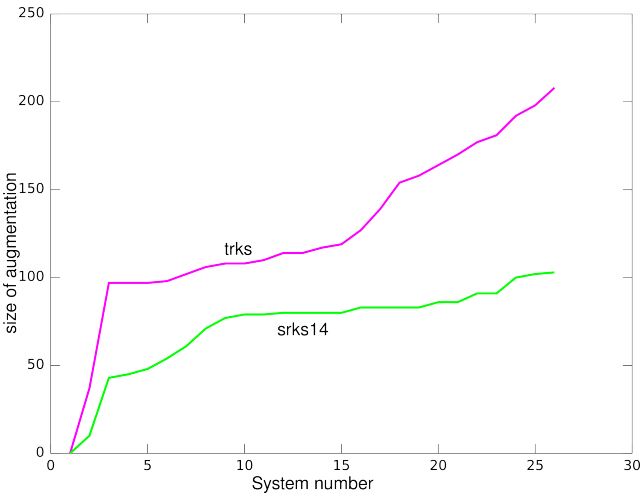


Figure 17: Buckling of the heterogeneous beam: dimension of the augmentation space for each linear system

subspaces ($\varepsilon = 10^{-14}$). Table 7 summarizes the main results.

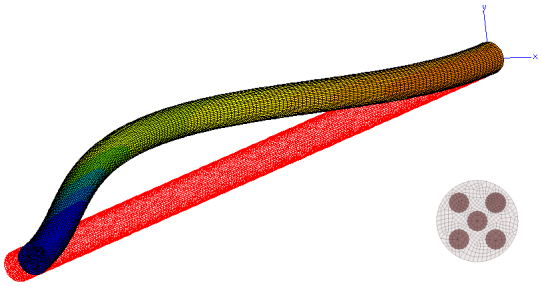


Figure 18: The beam in the reference and deformed configurations, with a view of the cross section

The following observations can be made:

- The performance of TRKS was really impressive because the systems converged in 10 times fewer iterations than with CG, but the size of the associated augmentation space was large (up to 208 vectors) and never ceased to increase;

method	avg. # it.	avg. n_c	CPU	WCT
CG	97.2	–	392	504
TRKS	7.8	131.7	203	217
SRKS 14	**33.8**	75.1	**278**	**298**

Table 7: Recycling performance for the buckling problem

- SRKS appeared to be efficient: the number of iterations was divided by 3 with a space whose size increased slowly, then reached a plateau;
- With the hardware configuration used, the best results in terms of computation time (a 48% CPU improvement) were achieved with TRKS, but the gain normalized by the number of augmentation vectors was better with SRKS (a 29% CPU improvement). SRKS was truly successful in controlling the dimension of the augmentation space.

In this example, the augmentation proved to be very efficient in terms of the iteration gain per augmentation vector, especially for SRKS (0.85). This was probably because of a specificity of the spectrum of the preconditioned operator due to the use of domain decomposition in large displacements. Indeed, the tangent matrix of a floating subdomain with prescribed Neumann conditions may become non-positive, contrary to the Dirichlet operator which remains positive definite. It is known that a slight lack of positivity of the operator does not prevent reorthogonalized conjugate gradients from converging [37], but convergence is slower than when all the eigenvalues are positive. The positivity of the preconditioner makes the selection procedure still possible. Moreover, the negative eigenvalues are systematically selected by the procedure, so using augmentation causes the solver to iterate in the subspace in which the operator is positive, leading to a much better convergence rate.

6 Conclusion

This paper dealt with the resolution of sequences of large linear systems with varying matrices and right-hand sides using conjugate gradients. We proposed several algorithms based on the augmentation of the current Krylov subspace by a selection of previously generated subspaces. The advantage of these methods is that some of the iterations are replaced by the preprocessing of a coarse problem associated with optimized operations.

When low accuracy is sufficient, total reuse of the previous subspaces (the TRKS algorithm) appears to lead to satisfactory results. When high accuracy is required, the subspaces are too unstable, which causes the dimension of the TRKS augmentation space to soar. Therefore, we proposed to retain only the part of the subspace generated by the Ritz vectors associated with converged Ritz values of the preconditioned operator (the SRKS algorithm). These vectors can be built very inexpensively. Such an augmentation was found to remain stable throughout the linear systems and to lead to a reduction in the number of iterations which is consistent with the theory. In terms of computation time, the proposed method leads to a variable, but always positive, gain compared to non-augmented systems. We observed CPU time improvements of 20% to 50%, and wall clock time improvements of 40% to 70%.

Up until now, our attempts to improve the selection algorithm by eliminating the converged values in the central part of the spectrum have not led to impressive results. This probably means that the Ritz vectors associated with converged Ritz values contain meaningful information which cannot be removed from the analysis of the current system. A continuation of this work could consist in a better analysis of the accumulation of the augmentation vectors. This can be done by studying the coarse matrix $\boldsymbol{C}^T\boldsymbol{A}\boldsymbol{C}$ whose distance to the identity matrix characterizes the variation of the Krylov subspaces. Another objective would be to port some of the ideas presented in this paper to nonsymmetric solvers.

References

[1] J. I. Aliaga, D. L. Boley, R. W. Freund, and V. Hernández. A Lanczos-type method for multiple starting vectors. *Mathematics of Computation*, 69:1577–1601, 2000.

[2] O. Axelsson and G. Lindskog. On the rate of convergence of the preconditioned conjugate gradient method. *Numerische Mathematik*, 48:499–523, 1986.

[3] A. H. Baker, J. M. Dennis, and E. R. Jessup. On improving linear solver performance: A block variant of GMRes. *SIAM Journal on Scientific Computing*, 27(5):1608–1626, 2006.

[4] J.F. Carpraux, S. Godunov, and S. Kuznetsov. Stability of the Krylov bases and subspaces. Technical Report 2296, INRIA, 1994.

[5] A. Chapman and Y. Saad. Deflated and augmented Krylov subspace techniques. *Numerical Linear Algebra with Applications*, 4(1):43–66, 1997.

[6] Z. Dostal. Conjugate gradient method with preconditioning by projector. *International Journal of Computer Mathematics*, 23:315–323, 1988.

[7] J. Erhel, K. Burrage, and B. Pohl. Restarted GMRes preconditioned by deflation. *Journal of Computational and Applied Mathematics*, 69:303–318, 1996.

[8] J. Erhel and F. Guyomarc'h. An augmented conjugate gradient method for solving consecutive symmetric positive definite linear systems. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1279–1299, 2000.

[9] C. Farhat, P.-S. Chen, and F.-X. Roux. The two-level FETI method - part II: Extension to shell problems. parallel implementation and performance results. *Computer Methods in Applied Mechanics and Engineering*, 155:153–180, 1998.

[10] C. Farhat, L. Crivelli, and F. X. Roux. Extending substructure based iterative solvers to multiple load and repeated analyses. *Computer Methods in Applied Mechanics and Engineering*, 117:195–209, 1994.

[11] C. Farhat and J. Mandel. The two-level FETI method for static and dynamic plate problems - part I: An optimal iterative solver for biharmonic systems. *Computer Methods in Applied Mechanics and Engineering*, 155:129–152, 1998.

[12] C. Farhat, K. Pierson, and M. Lesoinne. The second generation FETI methods and their application to the parallel solution of large-scale linear and geometrically non-linear structural analysis problems. *Computer Methods in Applied Mechanics and Engineering*, 184(2-4):333–374, 2000.

[13] C. Farhat and F. X. Roux. Implicit parallel processing in structural mechanics. *Computational Mechanics Advances*, 2(1):1–124, 1994. North-Holland.

[14] R. Freund. Model reduction methods based on Krylov subspaces. *Acta Numerica*, 12:267–319, 2003.

[15] Roland Freund. Krylov-subspace methods for reduced-order modeling in circuit simulation. *Journal of Computational and Applied Mathematics*, 123(1-2):395 – 421, 2000.

[16] P. Gosselet and C. Rey. On a selective reuse of Krylov subspaces in Newton-Krylov approaches for nonlinear elasticity. In *Proceedings of the* 14^{th} *conference on domain decomposition methods*, pages 419–426, Cocoyoc, Mexico, 2002.

[17] P. Gosselet and C. Rey. Non-overlapping domain decomposition methods in structural mechanics. *Archives of computational methods in engineering*, 13(4):515–572, 2007.

[18] P. Gosselet, C. Rey, P. Dasset, and F. Léné. A domain decomposition method for quasi incompressible formulations with discontinuous pressure field. *Revue européenne des élements finis*, 11:363–377, 2002.

[19] P. Gosselet, C. Rey, and D. Rixen. On the initial estimate of interface forces in FETI methods. *Computer Methods in Applied Mechanics and Engineering*, 192:2749–2764, 2003.

[20] P. Gosselet, D. Rixen, and C. Rey. A domain decomposition strategy to efficiently solve structures containing repeated patterns. *International Journal for Numerical Methods in Engineering*, 78(7):828–842, 2009.

[21] P.J. Heres, D. Deschrijver, W.H.A. Schilders, and T. Dhaene. Combining Krylov subspace methods and identification-based methods for model order reduction. *International Journal of Numerical Modelling : Electronic Networks, Devices and Fields*, 20(6):271–282, 2007.

[22] Z. Jia. The convergence of harmonic Ritz values, harmonic Ritz vectors and refined harmonic Ritz vectors. *Mathematics of computation.*, 74(251):1441–1456, 2004.

[23] G. Karypis and V. Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM Journal on Scientific Computing*, 20(1):359–392, 1998.

[24] M. E. Kilmer and E. de Sturler. Recycling subspace information for diffuse optical tomography. *SIAM Journal on Scientific Computing*, 27(6):2140–2166, 2006.

[25] A. Klawonn and O.B. Widlund. FETI and Neumann-Neumann iterative substructuring methods: connections and new results. *Communications on Pure and Applied Mathematics*, LIV:0057–0090, 2001.

[26] P. Le Tallec, J. Mandel, and M. Vidrascu. A Neumann-Neumann domain decomposition algorithm for solving plate and shell problems. *SIAM Journal on Numerical Analysis*, 35(2):836–867, April 1998.

[27] F. Léné and C. Rey. Some strategies to compute elastomeric lamified composite structures. *Composite structures*, 54:231–241, 2001.

[28] F. J. Lingen. Efficient Gram-Schmidt orthonormalisation on parallel computers. *Communication in Numerical Methods in Engineering*, 16:57–66, 2000.

[29] J. Mandel. Balancing domain decomposition. *Communication in Numerical Methods in Engineering*, 9:233–241, 1993.

[30] J. Mandel and M. Brezina. Balancing domain decomposition for problems with large jumps in coefficients. *Mathematics of Computation*, 65(216):1387–1401, 1996.

[31] M. Meyer and H. G. Matthies. Efficient model reduction in non-linear dynamics using the karhunen-loeve expansion and dual-weighted-residual methods. *Computational Mechanics*, 31(1-2):179–191, 2003.

[32] N. Molinari, C. Bonaldi, and J.P. Daurès. Multiple temporal cluster detection. *Biometrics*, 57:577–583, 2001.

[33] R. B. Morgan. A restarted GMRes method augmented with eigenvectors. *SIAM Journal on Matrix Analysis and Applications*, 16:1154–1171, 1995.

[34] Northwest Numerics. *Z-set user manual*, 2001.

[35] Y. Notay. On the convergence rate of the conjugate gradients in presence of rounding errors. *Numerische Mathematik*, 65:301–317, 1993.

[36] A. Nouy. A generalized spectral decomposition technique to solve a class of linear stochastic partial differential equations. *Computer Methods in Applied Mechanics and Engineering*, 196(45-48):4521–4537, 2007.

[37] C. Paige. Approximate solutions and eigenvalue bounds from Krylov subspaces. *Numerical Linear Algebra with Applications*, 2(2):115–133, 1995.

[38] M. Parks, E. De Sturler, G. Mackey, D. Johnson, and S. Maiti. Recycling Krylov subspaces for sequences of linear systems. *SIAM Journal on Scientific Computing*, 28(5):1651–1674, 2006.

[39] C. Rey and F. Risler. A Rayleigh-Ritz preconditioner for the iterative solution to large scale nonlinear problems. *Numerical Algorithms*, 17:279–311, 1998.

[40] Christian Rey. An acceleration technique for the solution of non-linear elasticity problems by domain decomposition. *Comptes Rendus de l'Académie des Sciences, Paris, série IIb*, 322(8):601–606, 1996.

[41] Franck Risler and Christian Rey. Iterative accelerating algorithms with Krylov subspaces for the solution to large-scale nonlinear problems. *Numerical Algorithms*, 23:1–30, 2000.

[42] D. Ryckelynck, F. Chinesta, E. Cueto, and A. Ammar. On the "a priori" model reduction: Overview and recent developments. *Archives of Computational Methods in Engineering*, 13:91–128, 2006.

[43] Y. Saad. Analysis of augmented Krylov subspace methods. *SIAM Journal on Matrix Analysis and Applications*, 18(2):435–449, April 1997.

[44] Y. Saad. *Iterative methods for sparse linear systems*. SIAM, Philadelphia, USA, 2nd edition, 2003.

[45] Y. Saad. *Numerical Methods for Large Eigenvalue Problems*, volume 66 of *Classics in Applied Mathematics*. SIAM, Philadelphia, USA, revised edition, 2011.

[46] Y. Saad, M. Yeung, J. Erhel, and F. Guyomarc'h. A deflated version of the conjugate gradient algorithm. *SIAM Journal on Scientific Computing*, 21(5):1909–1926, 2000.

[47] V. Simoncini and D. Szyld. Recent computational developments in Krylov subspace methods for linear systems. *Numerical Linear Algebra with Applications*, 14(1):1–59, 2007.

[48] J.M. Tang, R. Nabben, C. Vuik, and Y.A. Erlangga. Theoretical and numerical comparison of various projection methods derived from deflation, domain decomposition and multigrid methods. Reports of the Department of Applied Mathematical Analysis 07-04, Delft university of technology, 2007.

[49] A. van der Sluis and H. van der Vorst. The rate of convergence of conjugate gradients. *Numerische Mathematik*, 48:543–560, 1986.

[50] S. Wang, E. De Sturler, and G. Paulino. Large-scale topology optimization using preconditioner Krylov subspace methods with recycling. *International Journal for Numerical Methods in Engineering*, 69(12):2441–2468, 2007.

[51] Jia Z. and G.W. Stewart. On the convergence of the Ritz values, Ritz vectors and refined Ritz vectors. Technical Report 3896, Institute of Advanced Computer Studies, Department of Computer Science, University of Maryland at College Park, 1999.

E Article [Gosselet *et al.*, 2015], FETI-simultané et FETI-bloc

Cet article présente la méthode des variantes robustes de la méthode FETI. L'objectif est d'être capable de traiter les difficultés rencontrées en pratique sur les simulations industrielles, telles que de fortes hétérogénéités, des sous-domaines élancés, des interfaces irrégulières.

Les méthodes reposent sur l'introduction de blocs de directions de recherche. La méthode la plus prometteuse, S-FETI, s'interprète comme un multipréconditionnement, un article consacré à une mise en œuvre ambitieuse est en cours de préparation. L'objectif est de montrer que, modulo quelques techniques d'adaptation [Spillane, 2016], la méthode peut être efficace sur des problèmes industriels de taille raisonnable (quelques dizaines de millions d'inconnues, quelques centaines de sous-domaines) sur des clusters courants (quelques milliers de cœurs).

Une adaptation de la technique à certains solveurs non-symétriques est proposée dans [Bovet *et al.*, 2016].

INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2015; **104**:905–927
Published online 29 May 2015 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/nme.4946

Simultaneous FETI and block FETI: Robust domain decomposition with multiple search directions

Pierre Gosselet[1,*,†], Daniel Rixen[2], François-Xavier Roux[3] and Nicole Spillane[4]

[1]*LMT-Cachan/ENS-Cachan, CNRS, Université Paris-Saclay, 61 avenue du président Wilson, Cachan 94235, France*
[2]*Faculty of Mechanics, Institute of Applied Mechanics, Technische Universität München, Boltzmannstr. 15, Garching 85748, Germany*
[3]*Laboratoire Jacques-Louis Lions, CNRS UMR 7598, Université Pierre et Marie Curie, Paris 75005, France*
[4]*Center for Mathematical Modeling, Facultad de Ciencias Físicas y Matemáticas, Universidad de Chile, Beauchef 851, Santiago, Financial support by CONICYT (project Fondecyt 3150090) Chile*

SUMMARY

Domain decomposition methods often exhibit very poor performance when applied to engineering problems with large heterogeneities. In particular, for heterogeneities along domain interfaces, the iterative techniques to solve the interface problem are lacking an efficient preconditioner. Recently, a robust approach, named finite element tearing and interconnection (FETI)–generalized eigenvalues in the overlaps (Geneo), was proposed where troublesome modes are precomputed and deflated from the interface problem. The cost of the FETI–Geneo is, however, high. We propose in this paper techniques that share similar ideas with FETI–Geneo but where no preprocessing is needed and that can be easily and efficiently implemented as an alternative to standard domain decomposition methods. In the block iterative approaches presented in this paper, the search space at every iteration on the interface problem contains as many directions as there are domains in the decomposition. Those search directions originate either from the domain-wise preconditioner (in the simultaneous FETI method) or from the block structure of the right-hand side of the interface problem (block FETI). We show on two-dimensional structural examples that both methods are robust and provide good convergence in the presence of high heterogeneities, even when the interface is jagged or when the domains have a bad aspect ratio. The simultaneous FETI was also efficiently implemented in an optimized parallel code and exhibited excellent performance compared with the regular FETI method.

Received 20 August 2014; Revised 8 April 2015; Accepted 2 May 2015

KEY WORDS: domain decomposition; FETI; BDD; block Krylov methods; multiple preconditioner; heterogeneity

1. INTRODUCTION

Domain decomposition methods are mature solution techniques to enable computing the solution of large systems (typically arising from finite element models) on parallel computers. In particular, the nonoverlapping techniques such as the finite element tearing and interconnection (FETI) [1] and its primal counterpart (the balanced domain decomposition (BDD) [2]) have been successfully applied to solve several challenging mechanical problems (e.g., [3, 4]). The fundamental idea behind these efficient parallel solvers consists in solving local problems related to each domain with techniques that perform well sequentially on one processor and applying iterative techniques to find the interface unknowns connecting domains together, namely the interface forces in the dual Schur complement approaches (such as FETI) or the interface displacements in the primal Schur

*Correspondence to: Pierre Gosselet, LMT-Cachan/ENS-Cachan, CNRS, Université Paris-Saclay, 61 avenue du président Wilson, Cachan 94235, France.
†E-mail: pierre.gosselet@lmt.ens-cachan.fr

complement methods (such as BDD). Several variants of the primal and dual strategies in domain decomposition have been developed over the years to improve their robustness and efficiency: an overview can be found for instance in [5].

Nevertheless, for engineering problems where the structure is composed of parts with intricate shapes and/or made of materials with very different properties, domain decomposition techniques usually perform very poorly. A typical example one could mention is tires where the bulk is composed of soft rubber material in which different very stiff and slender components are embedded (steel cables and thin sheets of fiber-reinforced composites). A remedy could be to decompose these hard problems in such a way that each domain is nearly homogeneous [3]. Indeed, a number of contributions provide ways to solve problems with heterogeneities across the interfaces [6–13] or inside the subdomains and not near their boundaries [14–16]. This approach often results in bad load balancing and in domains with bad aspect ratios, which also pose a challenge for solving the interface problem iteratively. In the case of the Darcy equation, the authors in [17–19] prove convergence results, even in some cases where the jumps in the coefficients are along the interfaces. Within a class of coefficient distributions, the theory, which is based on weighted Poincare inequalities [20], allows discriminating between jumps in the coefficients that will affect convergence and those that will not. A typical case that cannot be solved efficiently with the, by now classical, techniques is shown in our numerical section (Figure 1). A significant contribution in achieving robustness, with much ongoing research, is the deluxe scaling (see [21] and references therein), which proposes to use near-optimal nondiagonal scaling. Another strategy, which the present work builds on, is presented next.

As was described in recent publications ([22–26] for the overlapping additive Schwarz method and [27] followed by [28–31] for nonoverlapping methods), the bad convergence of domain decomposition strategies in many challenging engineering problems can be traced back to the fact that important characteristics of the global problem cannot be approximated by the local information typically used to precondition the iterations on the interface problem. It was shown that the part of the problem that jeopardizes convergence can be revealed by eigenvalue problems on the interface of each subdomain: the strategy is to generate these problematic modes and apply deflation strategies (or coarse space approaches) to guarantee that the iterations are performed only on the part of the space that can be properly preconditioned. Those methods, given the generic name generalized eigenvalues in the overlaps (Geneo) in [24, 29, 32], exhibit remarkable robustness, both in theory and practice, even for decompositions where the domains have bad aspect ratios and where large heterogeneities across and along the interface are present (see Section 2 for more details). Unfortunately, robustness comes with a significant computational overhead related to finding the coarse space of 'bad' modes through eigenvalue problems. In Section 2, we introduce the Geneo coarse space for the FETI algorithm (see also Remark 2 for a comparison with the previous work [27]).

The objective of the present contribution is to propose robust FETI-type solvers that do not require a preprocessing step. We present two algorithms: the simultaneous FETI (S-FETI) algorithm and the block FETI (B-FETI) algorithm. The S-FETI algorithm was originally proposed in [33] for two subdomains and generalized in [34] to more subdomains. S-FETI consists in generating several search directions originating from the preconditioner and using them to solve the interface problem iteratively. A similar paradigm was later used in [35] to solve problems that include repeated components.

The previous contributions [33, 34] had discussed the potential of the method on some geometries but showed bad computational efficiency due to a crude and naive implementation of the algorithm. One major contribution of this paper, in addition to placing S-FETI in the context of FETI with the Geneo coarse space, B-FETI, and multipreconditioned conjugate gradient (CG) algorithms [36], is to propose an efficient organization of the algorithm such that it also becomes computationally highly efficient. The B-FETI algorithm is a block CG method [37]. The initialization procedure is new and allows rewriting the FETI linear system in block form in such a way that we update the solution at the same time as we compute Krylov subspaces that are very closely related to the Lanczos spaces for the computation of the local eigenvectors responsible for slow convergence. For this reason, the fundamental advantage over the usual FETI algorithm is that we are now able to catch several isolated eigenvalues within one iteration. We propose a conjecture in Equation (14)

DOI: 10.1002/nme

that attempts to draw a connection between the convergence of B-FETI and the convergence of the (robust and scalable) FETI algorithm with the Geneo coarse space [29]. Numerical experiments will confirm that both the B-FETI and S-FETI algorithms have similar convergence behaviors that are also similar to the convergence of FETI with the Geneo coarse space.

The outline of the article is as follows. In Section 2, we give a short summary of the FETI method as well as the Geneo coarse space because they form the basis for the proposed strategy. Then, Sections 3 and 4 are dedicated to explaining S-FETI and B-FETI, respectively. Finally, the methods are tested and evaluated in Section 5 for some simple but representative problems including some comparisons in CPU time between S-FETI and the classical FETI.

Note that, although only the FETI approaches are discussed in this paper, the ideas presented in this contribution can be extended in a straightforward manner to the other variants of FETI (such as the Dual-Primal FETI method (FETI-DP) [38, 39]) or to primal Schur complement methods such as BDD [2] and the Balancing Domain Decomposition by Constraints (BDDC) [40].

2. FETI IN A NUTSHELL

To begin, we shortly summarize the basic FETI strategy that will be used throughout this paper (see, for instance, [1, 5] for further details). Let us consider the symmetric positive definite problem $\mathbf{Ku} = \mathbf{f}$ associated with the finite element approximation of a linear mechanical problem set on domain Ω. Assume a partitioning into N subdomains $\Omega^{(s)}$ conforming to the mesh such that the partitioned problem writes

$$\begin{aligned} \mathbf{K}^{(s)}\mathbf{u}^{(s)} &= \mathbf{f}^{(s)} + \mathbf{t}^{(s)^T}\mathbf{B}^{(s)^T}\boldsymbol{\lambda} \\ \sum_s \mathbf{B}^{(s)}\mathbf{t}^{(s)}\mathbf{u}^{(s)} &= 0 \end{aligned} \tag{1}$$

where $\mathbf{t}^{(s)}$ are trace operators, $\mathbf{B}^{(s)}$ are signed Boolean assembly operators, and $\boldsymbol{\lambda}$ is the set of Lagrange multipliers that connect subdomains.

We use the following classical notations:

$$\mathbf{S}^{(s)} = \mathbf{K}^{(s)}_{bb} - \mathbf{K}^{(s)}_{bi}\mathbf{K}^{(s)^{-1}}_{ii}\mathbf{K}^{(s)}_{ib}; \quad \mathbf{F}^{(s)} = \mathbf{t}^{(s)}\mathbf{K}^{(s)^+}\mathbf{t}^{(s)^T}; \quad \mathbf{R}^{(s)} = \ker(\mathbf{K}^{(s)})$$

where $\mathbf{S}^{(s)}$ is the local Schur complement (i stands for internal degrees of freedom and b for boundary degrees of freedom), $\mathbf{F}^{(s)} = (\mathbf{S}^{(s)})^+$ is the local dual Schur complement, and $\mathbf{R}^{(s)}$ is a basis of rigid body modes. We also write

$$\begin{aligned} \mathbf{e} &= -\left(\ldots, \mathbf{f}^{(s)^T}\mathbf{R}^{(s)}, \ldots\right)^T & \mathbf{G} &= \left(\ldots, \mathbf{B}^{(s)}\mathbf{t}^{(s)}\mathbf{R}^{(s)}, \ldots\right) \\ \mathbf{F} &= \sum \mathbf{B}^{(s)}\mathbf{F}^{(s)}\mathbf{B}^{(s)^T} & \mathbf{d} &= -\sum_s \mathbf{B}^{(s)}\mathbf{t}^{(s)}\mathbf{K}^{(s)^+}\mathbf{f}^{(s)}, \end{aligned} \tag{2}$$

which leads to the classical FETI system:

$$\begin{pmatrix} \mathbf{F} & \mathbf{G} \\ \mathbf{G}^T & \mathbf{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{\lambda} \\ \ast \end{pmatrix} = \begin{pmatrix} \mathbf{d} \\ \mathbf{e} \end{pmatrix}. \tag{3}$$

The constraint $\mathbf{G}^T\boldsymbol{\lambda}$ is handled by the introduction of the initial estimate and the projector

$$\begin{aligned} \boldsymbol{\lambda}_0 &= \mathbf{AG}\left(\mathbf{G}^T\mathbf{AG}\right)^{-1}\mathbf{e} \\ \mathbf{P} &= \mathbf{I} - \mathbf{AG}\left(\mathbf{G}^T\mathbf{AG}\right)^{-1}\mathbf{G}^T \end{aligned} \tag{4}$$

so that $\mathbf{G}^T\boldsymbol{\lambda}_0 = \mathbf{e}$ and $\mathbf{G}^T\mathbf{P} = \mathbf{0}$. Matrix $\mathbf{A}$ is a symmetric positive definite matrix and can be taken as being the preconditioner $\tilde{\mathbf{S}}$ (discussed later in the text), identity, or a scaling matrix [41].

The unknown $\boldsymbol{\lambda}$ is sought as $\boldsymbol{\lambda} = \boldsymbol{\lambda}_0 + \mathbf{P}\tilde{\boldsymbol{\lambda}}$ where $\tilde{\boldsymbol{\lambda}}$ is a solution of

$$\mathbf{P}^T\mathbf{FP}\tilde{\boldsymbol{\lambda}} = \mathbf{P}^T\left(\mathbf{d} - \mathbf{F}\boldsymbol{\lambda}_0\right) = -\mathbf{P}^T\left(\sum_s \mathbf{B}^{(s)}\mathbf{t}^{(s)}\mathbf{K}^{(s)^+}\left(\mathbf{f}^{(s)} + \mathbf{t}^{(s)^T}\mathbf{B}^{(s)^T}\boldsymbol{\lambda}_0\right)\right). \tag{5}$$

DOI: 10.1002/nme

This system is solved by an iterative solver, the preconditioner $\tilde{\mathbf{S}}$ being

$$\tilde{\mathbf{S}} = \sum_s \tilde{\mathbf{B}}^{(s)} \tilde{\mathbf{S}}^{(s)} \tilde{\mathbf{B}}^{(s)^T}$$

where $\tilde{\mathbf{B}}^{(s)}$ are scaled assembling operators such that $\sum_s \mathbf{B}^{(s)} \tilde{\mathbf{B}}^{(s)^T} = \mathbf{I}$ and $\tilde{\mathbf{S}}^{(s)}$ are the Schur complements $\mathbf{S}^{(s)}$ or an approximation thereof. The scaling used in $\tilde{\mathbf{B}}^{(s)}$ is typically chosen based on the diagonal coefficients of the local stiffness matrices on the interface, namely a so-called k-scaling or super-lumped scaling. As explained in [7], this scaling can be seen as choosing a mechanically consistent combination of the interface reaction forces arising from the Dirichlet problem in each subdomain.

It is by now quite standard to augment the resolution by an additional constraint of the form $\mathbf{C}^T\mathbf{r} = 0$ where matrix $\mathbf{C}$ is a basis of a well-chosen subspace of range($\mathbf{P}$) and $\mathbf{r}$ is the residual vector. This means that at every iteration, the solution is required to solve the problem exactly in the subspace spanned by $\mathbf{C}$, or coarse space. When the constraint is implemented with a second level of initialization and projection, the resulting algorithm is referred to as FETI2 [42, 43]. It is summarized in Algorithm 1, where we introduced

$$\begin{aligned} \tilde{\boldsymbol{\lambda}}_0 &= \mathbf{C}\left(\mathbf{C}^T\mathbf{F}\mathbf{C}\right)^{-1}\mathbf{C}^T(\mathbf{d} - \mathbf{F}\boldsymbol{\lambda}_0) \\ \mathbf{P}_\mathbf{C} &= \mathbf{I} - \mathbf{C}\left(\mathbf{C}^T\mathbf{F}\mathbf{C}\right)^{-1}\mathbf{C}^T\mathbf{F}. \end{aligned} \tag{6}$$

Remark 1
In Algorithm 1, a full orthogonalization is employed as it is almost required when solving real engineering problems. A classical CG algorithm would correspond to $j = i$ (instead of $0 \leqslant j \leqslant i$) in the second-last line of the loop.

Algorithm 1: FETI2 with full orthogonalization

$\mathbf{r}_0 = \mathbf{P}^T {\mathbf{P}_\mathbf{C}}^T(\mathbf{d} - \mathbf{F}\boldsymbol{\lambda}_0)$
$\mathbf{z}_0 = \tilde{\mathbf{S}}\mathbf{r}_0$, $\mathbf{w}_0 = \mathbf{P}\mathbf{z}_0$, $\hat{\boldsymbol{\lambda}}_0 = 0$, $i = 0$
while $\sqrt{\mathbf{r}_i^T\mathbf{z}_i} > \epsilon$ **do**
  $\mathbf{q}_i = {\mathbf{P}_\mathbf{C}}^T\mathbf{F}\mathbf{w}_i$
  $\delta_i = \mathbf{q}_i^T\mathbf{w}_i$
  $\gamma_i = \mathbf{r}_i^T\mathbf{z}_i$
  $\hat{\boldsymbol{\lambda}}_{i+1} = \hat{\boldsymbol{\lambda}}_i + (\gamma_i/\delta_i)\mathbf{w}_i$
  $\mathbf{r}_{i+1} = \mathbf{r}_i - (\gamma_i/\delta_i)\mathbf{P}^T\mathbf{q}_i$
  $\mathbf{z}_{i+1} = \tilde{\mathbf{S}}\mathbf{r}_{i+1}$
  $\mathbf{w}_{i+1} = \mathbf{P}\mathbf{z}_{i+1}$ then for $0 \leqslant j \leqslant i$ $\begin{cases} \phi_{i,j} = \mathbf{q}_j^T\mathbf{w}_{i+1} \\ \mathbf{w}_{i+1} \leftarrow \mathbf{w}_{i+1} - (\phi_{i,j}/\delta_j)\mathbf{w}_j \end{cases}$
  $i \leftarrow i + 1$
end
$\boldsymbol{\lambda} = \boldsymbol{\lambda}_0 + \tilde{\boldsymbol{\lambda}}_0 + \mathbf{P}_\mathbf{C}\hat{\boldsymbol{\lambda}}_i$

Because the spectrum of the preconditioned FETI operator is bounded from below by 1, it is well known that matrix $\mathbf{C}$ should contain the eigenvectors $\mathbf{v}$ associated with the largest eigenvalues μ of the following generalized eigenvalue problem:

DOI: 10.1002/nme

$$\mathbf{P}^T\left(\mathbf{F}\mathbf{v}-\mu\tilde{\mathbf{S}}^{-1}\mathbf{v}\right)=0,\quad \mathbf{v}\in\text{range}(\mathbf{P}). \tag{7}$$

Various techniques can be employed to approximate these eigenvectors, like recycling of nearby Krylov subspaces [44]. One important result is provided by [29] where it is proved that the space spanned by the high-frequency eigenvectors can be approximated by solving (in parallel) a family of generalized eigenvalue problems. More precisely, in [29], it is proposed to compute the eigenpairs $(\mu_k^{(s)},\mathbf{v}_k^{(s)})$ indexed by k of

$$\mathbf{S}^{(s)}\mathbf{v}_k^{(s)}-\mu_k^{(s)}\mathbf{B}^{(s)^T}\tilde{\mathbf{S}}\mathbf{B}^{(s)}\mathbf{v}_k^{(s)}=0 \text{ for each } s=1,\ldots,N \tag{8}$$

and then define the columns of the constraint matrix $\mathbf{C}$ as $\left\{\mathbf{P}\tilde{\mathbf{S}}\mathbf{B}^{(s)^T}\mathbf{v}_k^{(s)};\ 0<\mu_k^{(s)}<\tau;\ s=1,\ldots,N\right\}$ for some chosen threshold $\tau\geqslant 0$. This is the FETI–Geneo algorithm, and it is guaranteed theoretically that the largest eigenvalue of the projected preconditioned operator is bounded by $\mathcal{N}\tau$ where $\mathcal{N}$ is the number of neighbors of a subdomain. In particular, this estimate does not depend on the number of subdomains (scalability) or the difficulty of the problem (robustness). The only additional assumption is that the matrix $\mathbf{A}$ in (4) correspond to the choice $\mathbf{A}=\tilde{\mathbf{S}}$. Unfortunately, the solution to these eigenproblems followed by an augmented resolution incurs a significant computational overhead, and the present article proposes a cheap, yet effective, alternative.

Remark 2
As already mentioned in the introduction, there is in fact a variety of coarse spaces that are constructed by solving generalized eigenvalues problems [22–28, 30, 31]. More specifically, the FETI–Geneo eigenvalue problem (8) is very strongly connected to the one that was previously proposed in [27, 28] for FETI-DP and BDDC. The difference is that, there, the eigenvalue problems are defined per interface between two subdomains whereas the Geneo problem involves a subdomain and all its neighbors (second term in (8)). Moreover the choice of eigenvalue problem relies on some heuristic assumptions. Also related to this family of methods, in [32], an algorithm (named frugal FETI) was sketched in order to capture the local contributions that penalize convergence within the CG algorithm.

The strategy proposed next, generalizing the multi-direction approach discussed in [33], can be considered as probably the simplest block procedure to build a multi-direction iteration and is thought to capture 'on the fly' the bad modes typically constructed a priori in Geneo approaches.

3. SIMULTANEOUS FETI

The S-FETI algorithm was introduced in [33] on a simple example with two subdomains. It enhances robustness by exploiting the additive structure of the preconditioner in order to generate as many search directions as there are subdomains at each step of the CG algorithm. We first present the algorithm in a general case (Algorithm 2) as well as the ideas that led to it. Then we discuss its connections with two existing algorithms, multipreconditioned CG and FETI–Geneo, and why it is expected to be robust. Finally, we comment on the cost of S-FETI and introduce an important trick (Equation (11)) to localize the application of the FETI operator, which makes our algorithm much more competitive.

3.1. The S-FETI algorithm

In classical FETI, the preconditioned residual writes $\mathbf{z}=\tilde{\mathbf{S}}\mathbf{r}=\sum_s\tilde{\mathbf{B}}^{(s)}\tilde{\mathbf{S}}^{(s)}\tilde{\mathbf{B}}^{(s)^T}\mathbf{r}$, and it is orthogonalized with respect to previous search directions to generate the new search direction. The idea underlying the S-FETI approach consists in letting the minimization process of the CG algorithm choose the best combination of local terms $\tilde{\mathbf{B}}^{(s)}\tilde{\mathbf{S}}^{(s)}\tilde{\mathbf{B}}^{(s)^T}\mathbf{r}$ (instead of simply adding them together to obtain $\mathbf{z}$), hence leading to an optimal, although more costly, choice. In other words, S-FETI, at a given iteration, uses each local term $\tilde{\mathbf{B}}^{(s)}\tilde{\mathbf{S}}^{(s)}\tilde{\mathbf{B}}^{(s)^T}\mathbf{r}$ as a search direction: the residual is minimized

DOI: 10.1002/nme

with respect to the subspace spanned by $\mathbf{Z} = \left(\ldots, \tilde{\mathbf{B}}^{(s)}\tilde{\mathbf{S}}^{(s)}\tilde{\mathbf{B}}^{(s)^T}\mathbf{r}, \ldots\right)$. The connection between $\mathbf{Z}$ and the usual $\mathbf{z}$ is of course that $\mathbf{z} = \mathbf{Z1}$ where $\mathbf{1} = (1, \ldots, 1)^T \in \mathbb{R}^N$, which explains why we monitor convergence in a classical way ($\sqrt{\mathbf{r}^\top \mathbf{z}}$ gives a measure of the residual comparable to the discretization error [45]).

The iteration scheme is given in Algorithm 2. For the sake of clarity, we give the size of the different operators

$$\mathbf{r}_i,\ \tilde{\boldsymbol{\lambda}}_i,\ \boldsymbol{\lambda},\ \boldsymbol{\lambda}_0 \in \mathbb{R}^n; \quad \mathbf{Z}_i,\ \mathbf{W}_i,\ \mathbf{Q}_i \in \mathbb{R}^{n\times N}; \quad \boldsymbol{\Delta}_i,\ \boldsymbol{\Phi}_{i,j} \in \mathbb{R}^{N\times N}; \quad \boldsymbol{\gamma}_i \in \mathbb{R}^N$$

where n is the number of unknowns and N is again the number of subdomains.

Algorithm 2: Simultaneous FETI

$\mathbf{r}_0 = \mathbf{P}^T(\mathbf{d} - \mathbf{F}\boldsymbol{\lambda}_0)$
$\mathbf{Z}_0 = \left(\ldots, \tilde{\mathbf{B}}^{(s)}\tilde{\mathbf{S}}^{(s)}\tilde{\mathbf{B}}^{(s)^T}\mathbf{r}_0, \ldots\right)$, $\mathbf{W}_0 = \mathbf{P}\mathbf{Z}_0$, $\tilde{\boldsymbol{\lambda}}_0 = 0$, $i = 0$
while $\sqrt{\mathbf{r}^T\mathbf{Z1}} > \epsilon$ **do**
  $\mathbf{Q}_i = \mathbf{F}\mathbf{W}_i$
  $\boldsymbol{\Delta}_i = \mathbf{Q}_i^T\mathbf{W}_i$
  $\boldsymbol{\gamma}_i = \mathbf{Z}_i^T\mathbf{r}_i$
  $\tilde{\boldsymbol{\lambda}}_{i+1} = \tilde{\boldsymbol{\lambda}}_i + \mathbf{W}_i\boldsymbol{\Delta}_i^+\boldsymbol{\gamma}_i$
  $\mathbf{r}_{i+1} = \mathbf{r}_i - \mathbf{P}^T\mathbf{Q}_i\boldsymbol{\Delta}_i^+\boldsymbol{\gamma}_i$
  $\mathbf{Z}_{i+1} = \left(\ldots, \tilde{\mathbf{B}}^{(s)}\tilde{\mathbf{S}}^{(s)}\tilde{\mathbf{B}}^{(s)^T}\mathbf{r}_{i+1}, \ldots\right)$
  $\mathbf{W}_{i+1} = \mathbf{P}\mathbf{Z}_{i+1}$ then for $0 \leqslant j \leqslant i$ $\begin{cases} \boldsymbol{\Phi}_{i,j} = \mathbf{Q}_j^T\mathbf{W}_{i+1} \\ \mathbf{W}_{i+1} \leftarrow \mathbf{W}_{i+1} - \mathbf{W}_j\boldsymbol{\Delta}_j^+\boldsymbol{\Phi}_{i,j} \end{cases}$
  $i \leftarrow i + 1$
end
$\boldsymbol{\lambda} = \boldsymbol{\lambda}_0 + \tilde{\boldsymbol{\lambda}}_i$

We point out that each iteration requires the inversion of the $N \times N$ matrix $\boldsymbol{\Delta}_i = \mathbf{W}_i^T\mathbf{F}\mathbf{W}_i$. Because $\mathbf{W}_i$ is the concatenation of localized contributions, it is reasonable to expect that $\boldsymbol{\Delta}_i$ be full-ranked. If it is not, then $\boldsymbol{\Delta}_i$ is only positive-semidefinite, and pseudo-inversion (denoted by $\boldsymbol{\Delta}_i^+$) is necessary. Another equivalent option would be to eliminate some directions in order to recover a full-rank family of vectors in $\mathbf{W}_i$. Then the next approximate solution would not change, but fewer vectors would need to be saved for future orthogonalization. In any case, the right-hand-side $\boldsymbol{\gamma}_i$ and range($\boldsymbol{\Phi}_{i,j}$) are both in range($\mathbf{W}_i^T$) = range ($\boldsymbol{\Delta}_i$) so that the iteration is always well defined.

Remark 3
More precisely, we propose the following procedure to compute the (pseudo)-inverse of $\boldsymbol{\Delta}_i$ and simplify the subsequent orthogonalization steps. A rank-revealing Cholesky factorization (symmetric pivoting) is employed just after $\boldsymbol{\Delta}_i$ is computed:

$$\mathbf{N}\boldsymbol{\Delta}_i\mathbf{N}^T = \mathbf{L}\mathbf{L}^T, \qquad \text{with } \mathbf{L} = \begin{pmatrix} \tilde{\mathbf{L}} & 0 \\ \times & 0 \end{pmatrix} \qquad \begin{matrix} \mathbf{N}: \text{ permutation matrix} \\ \tilde{\mathbf{L}}: \text{ lower triangular matrix.} \end{matrix} \tag{9}$$

The search directions can then be $\mathbf{F}$-orthonormalized and redundant directions suppressed by setting

$$\mathbf{W}_i \leftarrow \mathbf{W}_i\mathbf{N}^T\begin{pmatrix} \tilde{\mathbf{L}}^{-T} \\ 0 \end{pmatrix} \qquad \mathbf{Q}_i \leftarrow \mathbf{Q}_i\mathbf{N}^T\begin{pmatrix} \tilde{\mathbf{L}}^{-T} \\ 0 \end{pmatrix} \qquad \boldsymbol{\Delta}_i \leftarrow \mathbf{I}.$$

In that case, $\boldsymbol{\gamma}_i$ must be evaluated by the formula $\boldsymbol{\gamma}_i = \mathbf{W}_i^T\mathbf{r}_i$.

DOI: 10.1002/nme

3.2. S-FETI as a multipreconditioned CG algorithm: minimization property

The S-FETI algorithm is a multipreconditioned CG algorithm [46]. Indeed, at each iteration, N preconditioners are applied to generate the N columns in $\mathbf{Z}_i$. Then, each of these vectors is projected and orthogonalized to give a search direction (these are the N columns in $\mathbf{W}_i$). Finally, $\tilde{\boldsymbol{\lambda}}_i$ is updated to $\tilde{\boldsymbol{\lambda}}_{i+1}$ by adding the linear combination $\mathbf{W}_i \boldsymbol{\Delta}_i^+ \boldsymbol{\gamma}_i$ of these search directions that minimizes the error in the operator norm. Because the classical search direction $\mathbf{w} = \mathbf{W1} \in \text{range}(\mathbf{W})$, the new approximation is obviously always better than what classical CG would have given at that iteration.

A negative point of the method is that the CG short recurrence is broken, and full orthogonalization is required to fully benefit from the method. This is, however, only a theoretical drawback because, in practice, full orthogonalization is often used in classical FETI. The conjugacy ($\mathbf{F}$ orthogonality between search directions) is crucial because it ensures the following minimization property as proved in [46].

Theorem 1
The approximate solution computed by the i-th iteration of S-FETI minimizes the error $\tilde{\boldsymbol{\lambda}}_i - P\tilde{\boldsymbol{\lambda}}$ in the $\mathbf{F}$-norm (induced by the FETI operator) over all possible

$$\tilde{\boldsymbol{\lambda}}_i \in \bigoplus_{j=0}^{i-1} \text{span}\left(\mathbf{W}_j\right), \tag{10}$$

where $\oplus$ indicates a direct sum and $\mathbf{W}_j$ is defined in Algorithm 2.

The proof can be written in a similar way to the usual proofs for CG [47]. The two properties of multipreconditioned CG, which condition the choice of $\boldsymbol{\Delta}_i$, $\boldsymbol{\gamma}_i$, and $\boldsymbol{\Phi}_{i,j}$, are this minimization result and the $\mathbf{F}$ conjugacy of search directions $\left(\mathbf{W}_i^\top \mathbf{F} \mathbf{W}_j = 0, \quad \forall\, i \neq j\right)$.

A particularity of multipreconditioned CG algorithms is that the minimization space is not a Krylov subspace. The reason is that at each iteration, the approximate solution is updated in the direction given by the optimal linear combination of all preconditioners, but the coefficients (given by $\boldsymbol{\Delta}_i^+ \boldsymbol{\gamma}_i \in \mathbb{R}^N$) in the linear combination change from one iteration to the next. For this reason, we cannot justify a heuristic bound for the number of iterations as we will in (14) for B-FETI. We will, however, observe in Section 5 that on all test cases, both solvers behave in a similar way. This can be intuited by the fact that the search spaces are the same size ($N \times i$) and constructed with a combination of local and global components coming from the operator and the preconditioner. A strong connection between S-FETI and FETI–Geneo can also be drawn, which justifies why we expect such good robustness.

3.3. Connection to FETI–Geneo: why robustness is expected

As will become apparent in Section 5, the S-FETI algorithm is very efficient on hard problems for which the classical FETI typically requires many iterations. The convergence behavior is comparable to that of the FETI–Geneo algorithm [29] where a coarse space is constructed by solving in each subdomain a generalized eigenvalue problem (8) that isolates the part of the solution on which the preconditioner is not sufficiently efficient for the iterative solver to perform well. More precisely, the matrices in the pencil of the Geneo eigenproblems are on one hand $\mathbf{B}^{(s)^T} \tilde{\mathbf{S}} \mathbf{B}^{(s)}$ and on the other $\mathbf{S}^{(s)}$. With words, the vectors that are detected are the ones for which the local restriction of the (assembled) preconditioner $\mathbf{B}^{(s)^T} \tilde{\mathbf{S}} \mathbf{B}^{(s)}$ is not a good approximation for the nonassembled, local component $\mathbf{S}^{(s)}$ of the FETI operator. In S-FETI, the solution space results from successive applications of the local, nonassembled components $\tilde{\mathbf{B}}^{(s)} \mathbf{S}^{(s)} \tilde{\mathbf{B}}^{(s)^T}$ and the assembled $\mathbf{F}$, so the block of search directions spans a space where local effects are not gummed out. It thus bears similarities with the deflated space in which the Geneo iterations take place, and for this reason, convergence is expected to be very quick.

Remark 4
Although our numerical results (Section 5) point to the fact that S-FETI performs perfectly well, we mention the more recent multipreconditioned Generalized Minimal Residual (GMRES) algorithm

DOI: 10.1002/nme

[36] where, at the cost of saving more directions at each iteration, the error can be minimized over the larger subspace $\sum_{s=1}^{N} \mathcal{K}_i^N \left(\mathbf{F}, \mathbf{P}\tilde{\mathbf{B}}^{(s)}\mathbf{S}^{(s)}\mathbf{r}_0\right)$, with the multi-Krylov subspace defined by

$$\mathcal{K}_i^N(\mathbf{F}, \mathbf{x}) := \left\{ p\left(..., \mathbf{P}\tilde{\mathbf{B}}^{(s)}\mathbf{S}^{(s)}\tilde{\mathbf{B}}^{(s)^T}\mathbf{F}, ...\right)\mathbf{x};\ \begin{array}{c} p \text{ is a polynomial in} N \text{ variables} \\ \text{of degree at most } i-1 \end{array} \right\},$$

N being the number of subdomains. In [48], multiple preconditioned GMRES is applied to the Additive Schwarz domain decomposition technique.

3.4. Cost of S-FETI

An iteration of S-FETI does not require too much extra computational cost compared with classical FETI (for the discussion regarding parallelism, we assume that there is a bijection between the N subdomains and processors), for the following reasons:

- Exchanges are as frequent. Neighbor communications are identical although global reduction operations (due to scalar products) involve more data ($N \times N$ matrices $\mathbf{\Delta}$ and $\mathbf{\Phi}$, N vector $\boldsymbol{\gamma}$).
- Dense, but usually small, $N \times N$ symmetric positive-semidefinite matrices $\mathbf{\Delta}$ need to be (pseudo)-inverted.
- Sequences of N-blocks of vectors $\mathbf{W}_i$ and $\mathbf{Q}_i$ need to be stored instead of sequences of vectors.
- The most costly parts of the FETI algorithm are the local Neumann and Dirichlet solves in each subdomain. Compared with a classical FETI iteration, an S-FETI iteration does not require more work in the preconditioning step, but the operator $\mathbf{F}$ must now be applied to each of the N columns in $\mathbf{W}_i$. Nevertheless, the computation of $\mathbf{FW}_i$ can be performed efficiently. First, one has to remember that block operations are often proportionally much less expensive than single-vector operations because the computation time is driven by the memory access. Moreover, it is possible to cleverly use the locality of data by noting that $\mathbf{Z}_{i+1}$ is a sparse matrix (it only obtains values from its neighbors and itself) whereas $\mathbf{W}_{i+1}$ is not because of projection and orthogonalization. Furthermore, one observes that

$$\begin{aligned} \mathbf{Q}_{i+1} = \mathbf{FW}_{i+1} &= \mathbf{FPZ}_{i+1} - \sum_{j=0}^{i} \mathbf{Q}_j \mathbf{\Delta}_j^+ \mathbf{\Phi}_{i,j} \\ &= \left(\mathbf{FZ}_{i+1} - \mathbf{FAG}\left(\mathbf{G}^T\mathbf{AG}\right)^{-1}\mathbf{G}^T\mathbf{Z}_{i+1}\right) - \sum_{j=0}^{i} \mathbf{Q}_j \mathbf{\Delta}_j^+ \mathbf{\Phi}_{i,j} \end{aligned} \tag{11}$$

  where the product $\mathbf{FAG}$ is sparse (only neighbors of neighbors contribute to it) and can be computed once and for all during the initialization. This way, only localized Neumann problems $\mathbf{FZ}_{i+1}$ need to be solved, and the computational efficiency of S-FETI is significantly improved.

In the end, the extra cost per iteration is expected to remain very limited for a not-too-large number of subdomains N.[‡] It should also be noted that part of the additional cost is alleviated by the fact that the multiple Neumann problems to be solved by each domain at an iteration can be solved simultaneously as a block.

The extra costs have to be put in balance with our expectancy to divide the number of iterations to solve critical problems by a term of the order of N. More details on the practical implementation of S-FETI are given in Section 5.8.

4. BLOCK FETI

We present here the B-FETI technique. It is closely related to S-FETI with the advantage of preserving the short recurrence property. Indeed, it is a block CG algorithm [37] where an initial block of right-hand sides is generated in order to activate the local effects. This particular choice

[‡]Note that in the case of two subdomains, it was shown in [33] that the cost of one S-FETI iteration is nearly equal to an iteration of the classical FETI.

DOI: 10.1002/nme

of initialization is new, and we will justify it later in connection with the theory of the Geneo coarse space.

We first introduce the B-FETI algorithm. Then we give the minimization property that is satisfied at each iteration and discuss the choice of initialization and why it is expected that it leads to fast convergence (in a way comparable to FETI–Geneo). Finally, we comment on the cost of B-FETI.

4.1. The block FETI algorithm

The original block CG algorithm was designed to simultaneously solve the same problem for a number of different right-hand sides. Here, we solve for a unique right-hand side, but we reformulate the dual interface problem (3) in block form by considering separately the contribution of each subdomain to the right-hand side $\mathbf{d}$:

$$\mathbf{d} = \left(\ldots, \mathbf{B}^{(s)}\mathbf{t}^{(s)}\mathbf{K}^{(s)^+}\mathbf{f}^{(s)}, \ldots\right)\mathbf{1} = \left(\ldots, \mathbf{B}^{(s)}\mathbf{d}^{(s)}, \ldots\right)\mathbf{1}; \qquad \mathbf{1} = (1, \ldots, 1)^T \in \mathbb{R}^N.$$

Based on this fact, and for a given initial guess[§] $\boldsymbol{\lambda}_{00} \in \mathbb{R}^n$, we choose the initial block residual as

$$\mathbf{R}_0 := \mathbf{P}^T \left(\ldots, \mathbf{B}^{(s)}(\mathbf{d}^{(s)} - \mathbf{F}^{(s)}\mathbf{B}^{(s)^T}(\boldsymbol{\lambda}_0 + \mathbf{P}\boldsymbol{\lambda}_{00})), \ldots\right). \tag{12}$$

The B-FETI algorithm presented in Algorithm 3 is the block CG algorithm applied to the multiple right-hand side problem $\mathbf{P}^T\mathbf{F}\mathbf{P}\tilde{\boldsymbol{\Lambda}} = \mathbf{R}_0$. For the sake of clarity, we give the size of the different operators

$$\boldsymbol{\lambda},\ \boldsymbol{\lambda}_0,\ \boldsymbol{\lambda}_{00} \in \mathbb{R}^n; \quad \tilde{\boldsymbol{\Lambda}}_i, \mathbf{Z}_i,\ \mathbf{W}_i,\ \mathbf{Q}_i \in \mathbb{R}^{n\times N}; \quad \Gamma_i,\ \boldsymbol{\Delta}_i,\ \boldsymbol{\Phi}_{i,j} \in \mathbb{R}^{N\times N}$$

where again n is the number of unknowns and N is the number of subdomains. The block system is connected to the original system by the relation $\mathbf{r}_0 = \mathbf{R}_0\mathbf{1} = \mathbf{P}^T\left(\sum_s \mathbf{B}^{(s)}\left(\mathbf{d}^{(s)} - \mathbf{F}^{(s)}\mathbf{B}^{(s)^T}(\boldsymbol{\lambda}_0 + \mathbf{P}\boldsymbol{\lambda}_{00})\right)\right)$, so we can monitor the convergence of the original system within B-FETI, and the final solution is computed as $\boldsymbol{\lambda}_0 + \mathbf{P}\boldsymbol{\lambda}_{00} + \tilde{\boldsymbol{\Lambda}}_i\mathbf{1}$.

Algorithm 3: Block FETI with full orthogonalization (written as Block CG)

$\mathbf{R}_0 = \mathbf{P}^T\left(\ldots, \mathbf{B}^{(s)}(\mathbf{d}^{(s)} - \mathbf{F}^{(s)}\mathbf{B}^{(s)^T}(\boldsymbol{\lambda}_0 + \mathbf{P}\boldsymbol{\lambda}_{00})), \ldots\right)$
$\mathbf{Z}_0 = \tilde{\mathbf{S}}\mathbf{R}_0$, $\mathbf{W}_0 = \mathbf{P}\mathbf{Z}_0$, $\tilde{\boldsymbol{\Lambda}}_0 = 0$, $i = 0$
while $\sqrt{\mathbf{1}^T\mathbf{R}_i^T\mathbf{Z}_i\mathbf{1}} > \epsilon$ **do**
  $\mathbf{Q}_i = \mathbf{F}\mathbf{W}_i$
  $\boldsymbol{\Delta}_i = \mathbf{Q}_i^T\mathbf{W}_i$
  $\Gamma_i = \mathbf{R}_i^T\mathbf{Z}_i$
  $\tilde{\boldsymbol{\Lambda}}_{i+1} = \tilde{\boldsymbol{\Lambda}}_i + \mathbf{W}_i\boldsymbol{\Delta}_i^+\Gamma_i$
  $\mathbf{R}_{i+1} = \mathbf{R}_i - \mathbf{P}^T\mathbf{Q}_i\boldsymbol{\Delta}_i^+\Gamma_i$
  $\mathbf{Z}_{i+1} = \tilde{\mathbf{S}}\mathbf{R}_{i+1}$
  $\mathbf{W}_{i+1} = \mathbf{P}\mathbf{Z}_{i+1}$ then for $0 \leqslant j \leqslant i$ $\begin{cases} \boldsymbol{\Phi}_{i,j} = \mathbf{Q}_j^T\mathbf{W}_{i+1} \\ \mathbf{W}_{i+1} \leftarrow \mathbf{W}_{i+1} - \mathbf{W}_j\boldsymbol{\Delta}_j^+\boldsymbol{\Phi}_{i,j} \end{cases}$
  $i \leftarrow i + 1$
end
$\boldsymbol{\lambda} = \boldsymbol{\lambda}_0 + \mathbf{P}\boldsymbol{\lambda}_{00} + \tilde{\boldsymbol{\Lambda}}_i\mathbf{1}$

As for S-FETI, in each iteration we invert an $N \times N$ matrix $\boldsymbol{\Delta}_i$. In the algorithm, we use the notation $^+$ to refer to the pseudo inverse of $\boldsymbol{\Delta}_i$. Once more, these pseudo-inversions are well defined because range(Γ_i) and range($\boldsymbol{\Phi}_{i,j}$) are subsets of range($\boldsymbol{\Delta}_i$). The possibility that $\boldsymbol{\Delta}_i$ may be singular, or equivalently, that there be some linear dependence between the residuals, is a well-identified

[§] In Section 4.3, we propose to use a random initialization and explain why.

DOI: 10.1002/nme

problem in block CG. It corresponds to the case where the problem has been solved on a linear combination of the initial residuals. The solution is to deflate the block residual as is proposed in [49]. This deflation step is crucial to the efficiency of block CG because, the presence of a linear dependence means that all the work performed on one direction is not contributing to convergence anymore. Finally, we mention that the handling of $\boldsymbol{\Delta}_i$ can be performed according to Remark 3.

4.2. Minimization property and connection with Geneo

Thanks to the (new) choice of initialization in B-FETI, the algorithm satisfies a minimization property that justifies why we expect a convergence behavior similar to FETI–Geneo. First, we rewrite Algorithm 3 as Algorithm 4 where the approximate solution $\tilde{\boldsymbol{\lambda}}_i$ and residual $\mathbf{r}_i$ are updated directly and the block structure is only used to generate the search directions.

Algorithm 4: Block FETI with full orthogonalization (equivalent to Algorithm 3)

$\tilde{\boldsymbol{\lambda}}_0 = \boldsymbol{\lambda}_0 + \mathbf{P}\boldsymbol{\lambda}_{00}, \mathbf{r}_0 = \mathbf{P}^T\left(\mathbf{d} - \mathbf{F}\tilde{\boldsymbol{\lambda}}_0\right), i = 0$
$\mathbf{Z}_0 = \tilde{\mathbf{S}}\left(\ldots, \mathbf{B}^{(s)}(\mathbf{d}^{(s)} - \mathbf{F}^{(s)}\mathbf{B}^{(s)^T}\tilde{\boldsymbol{\lambda}}_0, \ldots\right)$
$\mathbf{W}_0 = \mathbf{P}\mathbf{Z}_0$
while $\sqrt{\mathbf{r}_i^T\mathbf{Z}_i\mathbf{1}} > \epsilon$ **do**
  $\mathbf{Q}_i = \mathbf{F}\mathbf{W}_i$
  $\boldsymbol{\Delta}_i = \mathbf{Q}_i^T\mathbf{W}_i$
  $\gamma_i = \mathbf{r}_i^T\mathbf{Z}_i$
  $\tilde{\boldsymbol{\lambda}}_{i+1} = \tilde{\boldsymbol{\lambda}}_i + \mathbf{W}_i\boldsymbol{\Delta}_i^+\gamma_i$
  $\mathbf{r}_{i+1} = \mathbf{r}_i - \mathbf{P}^T\mathbf{Q}_i\boldsymbol{\Delta}_i^+\gamma_i$
  $\mathbf{Z}_{i+1} = \tilde{\mathbf{S}}\mathbf{P}^T\mathbf{Q}_i$
  $\mathbf{W}_{i+1} = \mathbf{P}\mathbf{Z}_{i+1}$ then for $0 \leqslant j \leqslant i$ $\begin{cases} \boldsymbol{\Phi}_{i,j} = \mathbf{Q}_j^T\mathbf{W}_{i+1} \\ \mathbf{W}_{i+1} \leftarrow \mathbf{W}_{i+1} - \mathbf{W}_j\boldsymbol{\Delta}_j^+\boldsymbol{\Phi}_{i,j} \end{cases}$
  $i \leftarrow i + 1$
end
$\boldsymbol{\lambda} = \tilde{\boldsymbol{\lambda}}_i$

With this notation, the following minimization property holds.

Theorem 2
The approximate solution $\tilde{\boldsymbol{\lambda}}_i$ computed by the i-th iteration of B-FETI minimizes the error $\tilde{\boldsymbol{\lambda}}_i - P\tilde{\boldsymbol{\lambda}}$ in the $\mathbf{F}$-norm (induced by the FETI operator) over all possible

$$\tilde{\boldsymbol{\lambda}}_i \in \bigoplus_{s=1}^{N} \mathcal{K}_i\left(\tilde{\mathbf{S}}, \mathbf{F}, \mathbf{R}_0^{(s)}\right), \tag{13}$$

where $\oplus$ indicates a direct sum, $\mathbf{R}_0^{(s)} = \mathbf{B}^{(s)}(\mathbf{d}^{(s)} - \mathbf{F}^{(s)}\mathbf{B}^{(s)^T}(\boldsymbol{\lambda}_0 + \mathbf{P}\boldsymbol{\lambda}_{00}))$, and $\mathcal{K}_i\left(\tilde{\mathbf{S}}, \mathbf{F}, \mathbf{R}_0^{(s)}\right)$ is the associated Krylov subspace at iteration i:

$$\mathcal{K}_i\left(\tilde{\mathbf{S}}, \mathbf{F}, \mathbf{R}_0^{(s)}\right) = \text{span}\left\{\tilde{\mathbf{S}}\mathbf{R}_0^{(s)}, \ldots, (\tilde{\mathbf{S}}\mathbf{F})^{i-1}\tilde{\mathbf{S}}\mathbf{R}_0^{(s)}\right\}.$$

The proof can be written in a similar way to the usual proof of convergence for block CG [37], itself similar to the proof of convergence for CG [47]. The two main properties of B-FETI (which condition the choice of $\boldsymbol{\Delta}_i$, $\boldsymbol{\Phi}_{i,j}$, and $\boldsymbol{\Gamma}_i$ or γ_i as in any block CG algorithm) are this minimization result and $\mathbf{F}$ conjugacy of search directions $\left(\mathbf{W}_i^\top\mathbf{F}\mathbf{W}_j = 0, \quad \forall\, i \neq j\right)$. Moreover, in exact arithmetic, $\boldsymbol{\Phi}_{i,j} = 0$ for all $i < j$, which is the short recurrence property.

DOI: 10.1002/nme

Remark 5
The usual block CG optimality property [37] is that the error is minimized columnwise: each of the errors corresponding to each of the right-hand sides is minimized over the Krylov subspace generated by all N right-hand sides. The variable block CG method [49] was already an application of block CG to linear systems with a unique right-hand side with the objective of taking advantage of the enlarged Krylov subspace. There, in the initialization step, $N-1$ arbitrary vectors are added to the right-hand side $\mathbf{b}$ and to the initial guess $\mathbf{x_0}$ in order to construct a block initial residual, and so, part of the block CG minimization property is to ensure that the usual error $\mathbf{x}-\mathbf{x}_i$ is minimized over a subspace N times larger than the usual Krylov subspace for CG. This is also the case in Theorem 2 with the added advantage that the minimization spaces are designed to enhance convergence as explained further on.

An important difference with the S-FETI minimization result (Theorem 1) is that this time, the space over which we minimize can be written as a sum of Krylov subspaces. The fact that each of these Krylov subspaces is originated by a local vector is the reason why we make the following conjecture for the number n_{block} of iterations required to achieve convergence with B-FETI:

$$n_{block} \leqslant n_{geneo} + \max(n_{lanczos}), \tag{14}$$

where n_{geneo} and $n_{lanczos}$ are, respectively, the number of iterations needed to achieve convergence with the Geneo algorithm and the worse number of iterations of the Arnoldi-type eigensolver applied inside each subdomain to set up the Geneo coarse space. In turn, this conjecture would guarantee that B-FETI is scalable (i.e., that the convergence does not depend on the number of subdomains N): n_{geneo} does not depend on N [29] and $n_{lanczos}$ either because it refers to a computation inside one subdomain. This stems from the fact that we are simultaneously computing information comparable to the one that constitutes the Geneo coarse space and updating the approximate solution. The arguments to support this statement are the following:

(1) From the Geneo theory [29], we know that the eigenvalues of the preconditioned operator that are responsible for the slow convergence of FETI can be identified locally (per subdomain).
(2) The theory of block CG [37] states that the main advantage of this algorithm is that, instead of catching one bad eigenvalue per iteration, it can catch as many as there are blocks.
(3) The Krylov subspaces that are computed throughout the B-FETI iterations are exactly the spaces that are computed by an Arnoldi-type algorithm initialized by $\mathbf{R}_0^{(s)}$ to find the largest eigenvalues of the preconditioned operator.

In fact, we expect (14) to be a rather pessimistic estimate because we are in fact using more information than by solving first the generalized eigenvalue problems and then keeping only the eigenvectors: here, instead, we minimize over the full Lanczos subspaces. Another advantage is that we no longer need to specify a threshold for selecting eigenvectors because the process stops when convergence is achieved. We do point out that the subspaces do not correspond exactly to the Lanczos subspaces for computing the Geneo eigenvectors because there, the operators are also local. An exact diagnostic of which of these two methods requires more operations will be the subject of future work. Nevertheless, we give some insight on this question for a particular test case in Section 5.1.

4.3. Random initialization

A particular initialization $\boldsymbol{\lambda}_{00}$ is required in cases where the N local right-hand sides $\mathbf{B}^{(s)}\mathbf{K}^{(s)+}\left(\mathbf{f}^{(s)} - \mathbf{t}^{(s)^T}\mathbf{B}^{(s)^T}\boldsymbol{\lambda}_0\right)$ do not excite all subdomains. As can be seen from definitions (4) and (2), this can happen when not all domains are loaded and/or when not enough subdomains have rigid body modes.

Therefore, at initialization, a random starting vector $\boldsymbol{\lambda}_{00}$ is generated for $\tilde{\boldsymbol{\lambda}}$, and, in our applications, it was scaled to represent 1% of the forces $\left(\mathbf{f}^{(s)}\right)$. The reason to choose a random initialization is for all the columns in the initial residual to be linearly independent and also for all the vectors in the solution space to be represented. This is somewhat similar to the random initialization of the bootstrap adaptive multigrid algorithm [50].

DOI: 10.1002/nme

4.4. Cost of block FETI

One iteration of B-FETI is more expensive than S-FETI. The principal extra cost is due to the fact that both $\mathbf{F}$ and $\tilde{\mathbf{S}}$ need to be applied to N-blocks of vectors. The advantage is that the full reorthogonalization is no more an obligation, although necessary for most practical applications.

5. ASSESSMENTS

We first compare the different techniques for various academic problems known to trigger convergence difficulties with an `octave` implementation, and then we present first results for S-FETI on a realistic `fortran-mpi` implementation, which allows for time measurements.

5.1. High heterogeneity

These test cases are inspired by [24, 29]. Structures where heterogeneities are not aligned with the interface are known to cause convergence difficulties because classical scaling strategies are inefficient and only dedicated (Geneo) coarse problems can restore fast convergence.

Figure 1 presents one typical case of the two-dimensional (2D) representation of a horizontal beam clamped on its left side and submitted to given shear and traction on its right side; the ratio between the length and the thickness is 9. The beam is constituted by the stacking of seven layers of linear elastic materials with the same Poisson coefficient and Young moduli alternating between two values E_{stiff} and E_{soft}. This design aims at representing a soft material reinforced by stiff fibers. A band domain decomposition of nine square subdomains is employed, so that each subdomain has at most two neighbors and the material is identical on either side of each interface. Each subdomain is meshed by 434 first-order triangular elements, leading to a complete problem of 4200 degrees of freedom, of which 240 belong to the interface.

In Table I, we present the number of CG iterations needed for the classical FETI, S-FETI, and B-FETI; each of them is equipped either with the cheaper projector ($\mathbf{P}(\mathbf{A} = \mathbf{I})$) or with the optimal one ($\mathbf{P}(\mathbf{A} = \tilde{\mathbf{S}})$), for material contrasts ranging from 1 to 10^6. Because the initial residual depends on the choice for $\boldsymbol{\lambda}_0$ and hence on the choice for $\mathbf{A}$ in the rigid body mode projector (4), all iteration counts in Table I have been obtained using convergence criteria of 10^{-6} times the initial residual

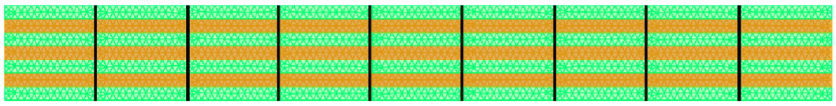


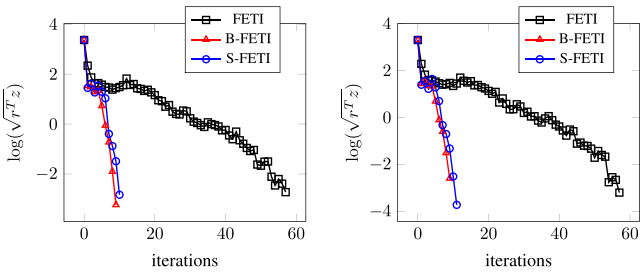
Figure 1. Heterogeneous beam, three stiff (orange or dark) fibers are embedded in a soft (green) material, nine-subdomain band decomposition.

Table I. Number of FETI iterations to decrease the initial residual by a 10^6 factor depending on the level of heterogeneity (problem in Figure 1).

$\frac{E_{stiff}}{E_{soft}}$	1	10	100	10^3	10^4	10^5	10^6
# Iterations FETI $\mathbf{P_I}$	6	9	18	34	51	63	67
# Iterations FETI $\mathbf{P}_{\tilde{\mathbf{S}}}$	5	6	9	18	31	40	43
# Iterations S-FETI $\mathbf{P_I}$	5	7	10	12	12	12	11
# Iterations S-FETI $\mathbf{P}_{\tilde{\mathbf{S}}}$	5	6	8	9	10	9	9
# Iterations B-FETI $\mathbf{P_I}$	5	7	9	10	11	11	11
# Iterations B-FETI $\mathbf{P}_{\tilde{\mathbf{S}}}$	5	6	8	11	11	11	11
# Iterations FETI Geneo $\mathbf{P}_{\tilde{\mathbf{S}}}$	4	5	6	5	5	5	5
(fixed coarse space size)	(54)	(54)	(54)	(54)	(54)	(54)	(54)
# Iterations FETI Geneo $\mathbf{P}_{\tilde{\mathbf{S}}}$	6	6	9	14	12	6	5
($0 < \mu^{(s)} < 0.15 \Rightarrow$ variable coarse space size)	(0)	(0)	(0)	(13)	(32)	(44)	(47)

FETI, finite element tearing and interconnection; S-FETI, simultaneous FETI; B-FETI, block FETI; Geneo, generalized eigenvalues in the overlaps.

DOI: 10.1002/nme



(a) Coarse Mesh (b) Fine Mesh

Figure 2. Convergence history: heterogeneous beam in flexion, 10^5 heterogeneity, $\mathbf{P_I}$. (a) Course mesh and (b) fine mesh. FETI, finite element tearing and interconnection; B-FETI, block FETI; S-FETI, simultaneous FETI.

corresponding to $\mathbf{A} = \tilde{\mathbf{S}}$ in order to investigate only the convergence behavior independently of the initial estimate (the influence of the choice of $\mathbf{A}$ on the initial residual has been discussed in [51]).

We observe that even the optimized projector cannot prevent classical FETI from experiencing a degradation of its performance when heterogeneity increases (for the highest heterogeneity, FETI with $\mathbf{P}_{\tilde{\mathbf{S}}}$ needs nine times as many iterations than in the homogeneous case). On the other hand, S-FETI and B-FETI are much more robust, because at worse, they need twice as many iterations as in the homogeneous case. Figure 2(a) presents one classical evolution of the residual throughout of the CG iterations; Figure 2(b) was obtained with a refined mesh (seven times as many degrees of freedom), illustrating the independence of the performance with respect to the discretization.

At the bottom of Table I, we have also included the number of iterations needed for FETI–Geneo (Algorithm 1 with the coarse space described at the end of Section 2) to converge. In the first case, we have chosen the coarse space size to be 54: we select in each subdomain the six eigenvectors from (8) associated with the smallest nonzero eigenvalues. It is known by experience [29] that this choice catches all the bad eigenvectors, and indeed, we observe that the number of iterations needed to converge is not influenced by the material heterogeneity. In the second case, we have let the method select automatically the size of the coarse space by selecting all eigenvectors from (8) such that $0 < \mu^{(s)} < 0.15$, which guarantees [29] that the condition number of the preconditioned operator is below $3/0.15 = 20$. We observe that when the material is not very heterogeneous, the method does not construct a coarse space and the number of iterations increases slightly from six, when $E_{stiff} = E_{soft}$, to nine, when $E_{stiff} = 100\, E_{soft}$. After that, the size of the coarse space increases with the heterogeneity (never exceeding 47), and the number of iterations remains below 14.

Of course, the number of iterations is only one indication of how a method performs. Because this is the most costly part of the FETI algorithms, we now compare the number of local solves. We use notation N for the number of subdomains and $\mathcal{N}$ for the largest number of neighbors of a subdomain including itself:

- Within the classical FETI algorithm, two local solves (one Dirichlet and one Neumann) are performed per subdomain.
- Each iteration of S-FETI requires in each subdomain $\mathcal{N}$ Neumann solves (as explained in the last item in the discussion on the cost of S-FETI in Section 3.4) and 1 Dirichlet solve.
- Within B-FETI, once all blocks of vectors have complete fill-in, each iteration requires N Dirichlet solves and N Neumann solves per subdomain.
- Within FETI2 with the Geneo coarse space, no extra local solve is required in the iteration process, but we need to consider the overhead cost of solving the Geneo eigenproblem. We assume that it is solved approximately by an iterative Lanczos method and that the computation of each eigenvector requires three Lanczos iterations and hence three applications of the operator $\mathbf{S}^{(s)^{-1}}\mathbf{B}^{(s)^T}\tilde{\mathbf{S}}\mathbf{B}^{(s)}\mathbf{v}^{(s)}$, which means 3 Neumann solves and $3\mathcal{N}$ Dirichlet solves. In order to build the coarse space, we also need to apply the preconditioner to the eigenvector, meaning

DOI: 10.1002/nme

$\mathcal{N}$ extra Dirichlet solves. In conclusion, the cost of computing one vector for the coarse space is $3 + 4\mathcal{N}$.

In the test case at hand, we have $N = 9$ and $\mathcal{N} = 3$, so each S-FETI iteration requires twice as many local solves as a classical FETI iteration, each B-FETI iteration requires nine times as many local solves as a classical FETI iteration, and FETI–Geneo iterations are as costly as classical FETI iterations, but the computation of each eigenvectors requires approximately 15 local solves.

With this and the results from Table I, we conclude that B-FETI increases the number of local solves while S-FETI and FETI–Geneo increase it for the easier problems and reduce it for the harder problems. The best improvements are observed with S-FETI.

In defense of the B-FETI method, we recall that it does not require full reorthogonalization and that applying the same operator to N columns of a vector is much less expensive than applying the same operator N times. We also recall that FETI–Geneo is at this time the only algorithm for which convergence in a few iterations is guaranteed theoretically.

Remark 6

In this discussion, we have left out two important parameters: the cost of orthogonalization (including for the coarse space) and the number of applications of the projector. In fact, as usual, the only way to truly compare the performance of our solver is to compare CPU times, which we do in Section 5.8 for FETI and S-FETI.

5.2. *Bad aspect ratio*

Another cause for bad convergence is the bad aspect ratio of subdomains. To activate this problem, we dilate the previous problem in the transverse directions: thickness now ranges between $1/5$ and 10, while the length of subdomains remains 1. The aspect ratio is defined as the thickness divided by the length. The connectivity of the mesh (and thus the number of nodes) is unchanged; elements are distorted when aspect ratio is far from 1.

Table II presents the number of CG iterations required to decrease the residual by a 10^6 factor. We observe that classical FETI converges more slowly when the interfaces are proportionally closer (aspect ratio > 1): three times as many iterations for an aspect ratio of 5 and five times as many iterations when aspect ratio is 10. S-FETI and B-FETI are in the worst case twice as slow.

5.3. *Irregular interfaces*

The shape of interfaces is known to have a strong influence on the convergence of the solver [9]: roughly, the straighter the better. The irregular decomposition of the beam shown in Figure 3 was obtained by an automatic graph partitioner (Metis [52]). Performances are presented in Table III.

Table II. Number of FETI iterations to decrease the initial residual by a 10^6 factor depending on the aspect ratio (problem in Figure 1).

Aspect ratio	$1/5$	1	5	10
# Iterations FETI $\mathbf{P_I}$	5	6	17	29
# Iterations S-FETI $\mathbf{P_I}$	5	5	9	11
# Iterations B-FETI $\mathbf{P_I}$	5	5	8	10

FETI, finite element tearing and interconnection; S-FETI, simultaneous FETI; B-FETI, block FETI.

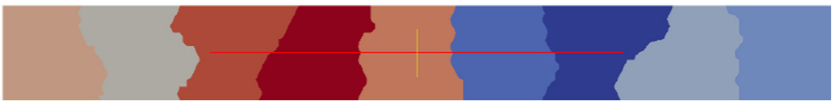


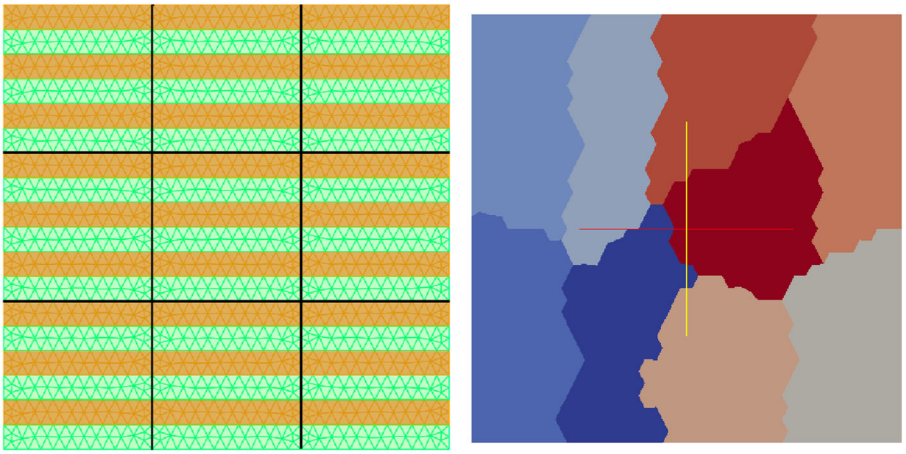
Figure 3. Homogeneous beam with irregular interfaces.

DOI: 10.1002/nme

Table III. Number of FETI iterations to decrease the initial residual by a 10^6 factor depending on the decomposition (Figures 1 and 3).

Decomposition	Straight	Irregular
# Iterations FETI $\mathbf{P_I}$	6	18
# Iterations FETI $\mathbf{P_{\tilde{S}}}$	5	17
# Iterations S-FETI $\mathbf{P_I}$	5	9
# Iterations S-FETI $\mathbf{P_{\tilde{S}}}$	5	9
# Iterations B-FETI $\mathbf{P_I}$	5	10
# Iterations B-FETI $\mathbf{P_{\tilde{S}}}$	5	10

FETI, finite element tearing and interconnection; S-FETI, simultaneous FETI; B-FETI, block FETI.



(a) Heterogeneous square (green = soft material, orange or dark = stiff material), regular decomposition (black lines)

(b) Automatic decomposition

Figure 4. Heterogeneous square. (a) Heterogeneous square (green = soft material, orange or dark = stiff material, black lines, regular decomposition). (b) Automatic decomposition.

Table IV. Number of FETI iterations to decrease the initial residual by a 10^6 factor for the square problem (problem in Figure 4).

Heterogeneity ratio Decomposition	1 Regular	1 Automatic	10^5 Regular	10^5 Automatic
# Iterations FETI $\mathbf{P_I}$	12	19	42	94
# Iterations FETI $\mathbf{P_{\tilde{S}}}$	12	21	48	93
# Iterations S-FETI $\mathbf{P_I}$	8	11	10	19
# Iterations S-FETI $\mathbf{P_{\tilde{S}}}$	8	12	11	19
# Iterations B-FETI $\mathbf{P_I}$	7	9	8	13
# Iterations B-FETI $\mathbf{P_{\tilde{S}}}$	7	9	9	13

FETI, finite element tearing and interconnection; S-FETI, simultaneous FETI; B-FETI, block FETI.

We observe that the S-FETI and B-FETI are less impacted by the irregularity of the interfaces than classical FETI. This will also be illustrated in the next example.

DOI: 10.1002/nme

5.4. Decomposition with cross-points

Cross-points, namely interface nodes that belong to more than two domains, often appear in decomposed problems and are known to sometimes be the cause of bad convergence in the case of heterogeneous structures [7].

Figure 4 presents the heterogeneous domain with two decompositions (regular and automatic). The square is clamped on its bottom side and submitted to traction and shear on its top side. The discretization leads to about 2800 degrees of freedom; in the regular case, the dimension of the interface is 300, and in the automatic case, it is 350. Table IV summarizes the performance in the case of homogeneous or heterogeneous square (10^5 heterogeneity) with regular or automatic decompositions into nine subdomains. As previously, we observe that the S-FETI and B-FETI are much less influenced than the classical FETI by the irregular decomposition and the high heterogeneity in the presence of cross-points.

5.5. Incompressibility

Incompressibility is also a known factor for convergence difficulties. A classical remedy is to add a coarse problem related to the conservation of the volume of the subdomains [12, 13, 16, 53, 54].

We consider the geometry of the beam (Figure 1) with homogeneous linear elastic material in plane strain. The bottom and top faces are clamped, and a pressure is imposed on the left side, whereas the right side is free. Table V gives the number of iterations to converge for various Poisson coefficients close to the incompressible limit $\nu \simeq 0.5$.

We observe how block strategies enable to limit the degradation of the convergence rate: when $(0.5 - \nu)$ goes from 10^{-5} to 10^{-6}, classical FETI needs twice as many iterations, whereas block strategies only need 25% more iterations. Although this is interesting, it is clear that when the incompressibility limit is approached, an additional coarse grid as proposed in [53] is needed.

5.6. Scalability results

We present scalability results for 2D problems. Beam-like structures similar to the ones in Figure 1 and in Figure 4(a) are again considered. We make the number of domains vary considering a decomposition only in the horizontal direction for the problem of Figure 1 and a regular decomposition in squares for the problem of Figure 4. While changing the number of domains, we keep the discretization per domain constant; that is, the size of the local problems in a domain are constant, whereas the size of the overall problem is changing with the number of domains. The results are given in Tables VI and VII, respectively. In the homogeneous case, the convergence of classical FETI is weakly impacted by the increase of the number of subdomains, whereas in the heterogeneous case, the number of iterations explodes when increasing N. S-FETI and B-FETI are much less sensitive with respect to the number of subdomains, which can be partially explained by the fact that for these methods, adding subdomains also means adding search directions.

To investigate the convergence when the size of the local problems is changing, we consider the problem of Figure 4(a) with a decomposition in 3×3 domains and make the element size vary (hence also the size of the local problems). From Table VIII, it is observed that S-FETI and B-FETI are barely affected by the discretization size similarly to what is expected for the classical FETI.

Table V. Number of FETI iterations to decrease the initial residual by a 10^6 factor for the quasi-incompressible problem (problem in Figure 1).

	$1/2 - \nu = 10^{-1}$	$1/2 - \nu = 10^{-5}$	$1/2 - \nu = 10^{-6}$
# Iterations FETI	5	31	63
# Iterations S-FETI	5	18	23
# Iterations B-FETI	5	18	22

DOI: 10.1002/nme

5.7. *Inclusion problem*

We consider the problem of inclusions near the boundary of the subdomains as represented in Figure 5. Let L/H be the thickness of the layer between the boundary and the inclusion relative to the size of the subdomain. Table IX gives the number of iterations for FETI, S-FETI, and B-FETI in various geometric and material configurations. As explained in [16] (where a dedicated coarse space is proposed in the context of FETI-DP and BDDC to control the condition number), the soft inclusion case is problematic for classical domain decomposition methods, in particular when the inclusions are near the boundary.

As in previous experiments, S-FETI and B-FETI algorithms behave more robustly than classical FETI: for the thinnest layer case, the worst heterogeneous case compared with the homogeneous

Table VI. Number of FETI iterations for various strip decompositions (problem in Figure 1, constant subdomain size).

Nbr. of domains N		2	4	8	16	32
Homogeneous beam	FETI $\mathbf{P_I}$	5	6	6	6	6
	S-FETI $\mathbf{P_I}$	5	5	5	5	5
	B-FETI $\mathbf{P_I}$	5	5	5	5	5
Heterogeneous (10^5) beam	FETI $\mathbf{P_{\tilde{S}}}$	7	18	37	68	112
	S-FETI $\mathbf{P_{\tilde{S}}}$	5	8	9	10	10
	B-FETI $\mathbf{P_{\tilde{S}}}$	7	11	13	13	14
	S-FETI $\mathbf{P_I}$	7	10	12	13	13
	B-FETI $\mathbf{P_I}$	7	9	10	11	12

FETI, finite element tearing and interconnection; S-FETI, simultaneous FETI; B-FETI, block FETI.

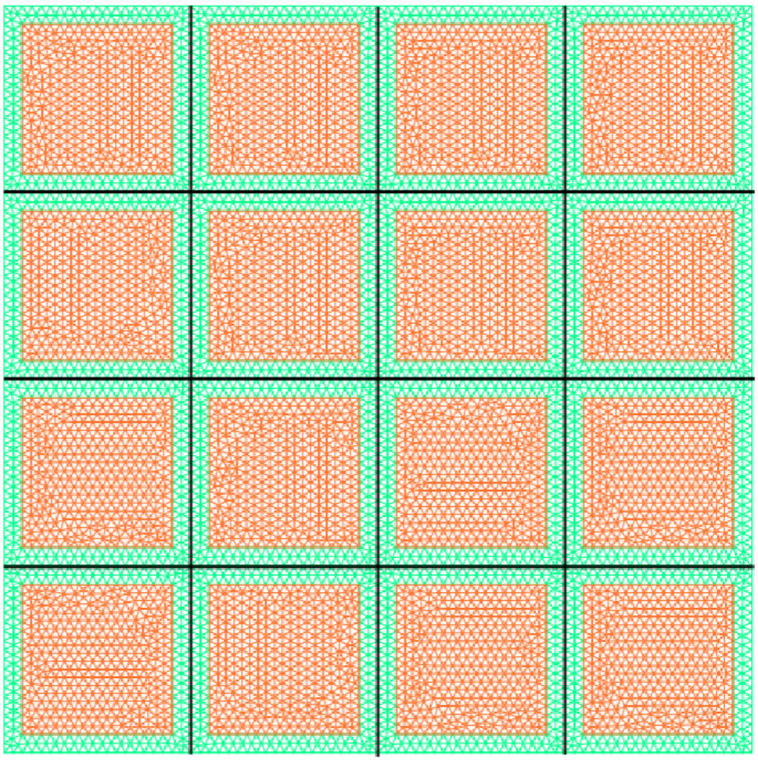


Figure 5. 4×4 square domain with inclusions (10^5 stiffness ratio).

DOI: 10.1002/nme

Table VII. Number of FETI iterations for various ruled decompositions of a square (problem in Figure 4(a), constant subdomain size).

		2 × 2 sd	3 × 3 sd	4 × 4 sd	5 × 5 sd
Homogeneous square	FETI $\mathbf{P_I}$	9	12	14	15
	S-FETI $\mathbf{P_I}$	8	8	9	9
	B-FETI $\mathbf{P_I}$	6	7	7	7
Heterogeneous (10^5) square	FETI $\mathbf{P_{\tilde{S}}}$	12	48	89	143
	S-FETI $\mathbf{P_{\tilde{S}}}$	7	11	13	14
	B-FETI $\mathbf{P_{\tilde{S}}}$	6	9	9	10
	S-FETI $\mathbf{P_I}$	7	10	12	13
	B-FETI $\mathbf{P_I}$	5	8	9	9

FETI, finite element tearing and interconnection; S-FETI, simultaneous FETI; B-FETI, block FETI.

Table VIII. Number of FETI iterations for 3 × 3 decomposition of a square with varying discretization (problem in Figure 4(a)).

	# Nodes	1531	4051	6919	10092	14273
Homogeneous square	FETI $\mathbf{P_I}$	12	13	14	15	15
	S-FETI $\mathbf{P_I}$	8	9	9	9	10
	B-FETI $\mathbf{P_I}$	7	7	8	8	8
Heterogeneous (10^5) square	FETI $\mathbf{P_{\tilde{S}}}$	48	42	46	46	50
	S-FETI $\mathbf{P_I}$	10	11	13	11	13
	B-FETI $\mathbf{P_I}$	8	8	9	9	9

FETI, finite element tearing and interconnection; S-FETI, simultaneous FETI; B-FETI, block FETI.

case requires 2.5 times more iterations for classical FETI and only 1.3 for S-FETI and 1.37 for B-FETI.

Table IX. Number of FETI iterations for 4×4 squares with thin layers (Figure 5).

L/H	1/8 (9 500 dofs)			1/12 (21 000 dofs)		
E_{inclu}/E_{bulk}	1	10^5	10^{-5}	1	10^5	10^{-5}
FETI	15	13	33	16	13	41
S-FETI	9	9	12	10	9	13
B-FETI	8	7	10	8	7	11

FETI, finite element tearing and interconnection; S-FETI, simultaneous FETI; B-FETI, block FETI.

5.8. Optimized implementation and time measurement for S-FETI

The implementation of the S-FETI method can be optimized in several ways (refer to Algorithm 2):

Simultaneous forward-backward substitutions By definition, the s-th column of $\mathbf{Z}_i$ is nonzero only on the interface of subdomain $\Omega^{(s)}$, and so, given a column in $\mathbf{Z}_i$, its product by $\mathbf{F}$ requires solving a Neumann problem only in the subdomain itself and its neighbors. Conversely, given one subdomain $\Omega^{(s)}$, the work load associated with the computation of $\mathbf{F}\mathbf{Z}_i$ is $\mathcal{N}$ Neumann solves $\mathbf{F}^{(s)}$ (once more, $\mathcal{N}$ is the number of neighbors of a subdomain including itself). In particular, this is much fewer than the rank of $\mathbf{Z}_i$, which is equal to the number of subdomains. These multiple local solutions can be computed much more efficiently on a multi-core machine as a single

DOI: 10.1002/nme

solution. Indeed, in both cases, the number of memory accesses is almost equal to the number of nonzero entries in the factorized matrix, whereas the arithmetic complexity is multiplied by the number of simultaneous right-hand sides with the result that multiple forward–backward substitutions do not present the same memory bottleneck issue than a single one. For instance, for a finite element sparse matrix of dimension 200 000 on a 12-core Intel Nehalem processor, Santa Clara, California, US, the time for a single forward–backward substitution on a single core is 0.7 s with the Intel Pardiso [55] solver, whereas the time for 12 simultaneous forward–backward substitutions on 12 cores is only 1 s. For a single right-hand side, multi-core parallelization only decreases the time by 30% at best. Finally, if several single forward substitutions for different matrices are performed at the same time, the memory bottleneck causes the time for each one to increase dramatically.

Parallel implementation of $\mathbf{P}$ with low-rank corrections Within S-FETI, the columns of $\mathbf{W}_i$ are built from the projected vectors $\mathbf{PZ}_i$, and these are not local. Fortunately, the FETI projection $\mathbf{P}$ only performs a low-rank correction, so $\mathbf{PZ}_i$ can be computed at only a small extra cost. This cost is even further reduced with the simple $\mathbf{P_I}$ projector (i.e., when $\mathbf{A} = \mathbf{I}$; Section 2), which we have chosen to use in our tests. It operates as follows:

$$\mathbf{P_I Z}_i = \mathbf{Z}_i - \mathbf{G}\boldsymbol{\beta}_i \text{ where } \boldsymbol{\beta}_i \text{ solves } \mathbf{G}^T\mathbf{G}\boldsymbol{\beta}_i = \mathbf{G}^T\mathbf{Z}_i.$$

As usual, this guarantees that $\mathbf{G}^T\mathbf{P_I Z}_i = 0$. Note that by construction, $\mathbf{G}$ has the same sparse pattern as $\mathbf{Z}_i$, meaning that, given a column $\mathbf{Z}_i^{(s)}$ of $\mathbf{Z}_i$, $\mathbf{G}^T\mathbf{Z}_i^{(s)}$ is computed by applying only dot products by the columns of $\mathbf{G}$ corresponding to $\Omega^{(s)}$ and its neighbors. For this reason and because $(\mathbf{G}^T\mathbf{G})$ was factorized during the initialization phase of FETI, $\boldsymbol{\beta}_i$ can be computed in parallel. Each subdomain is in charge of computing one column $\boldsymbol{\beta}_i^{(s)}$ of $\boldsymbol{\beta}_i$ by solving, via a forward–backward substitution, one system $(\mathbf{G}^T\mathbf{G})\boldsymbol{\beta}_i^{(s)} = \mathbf{G}^T\mathbf{Z}_i^{(s)}$. Of course, $\boldsymbol{\beta}_i$ is a dense matrix, whose number of rows is equal to the rank of $\mathbf{G}$ and number of columns is equal to the number of subdomains (also the rank of $\mathbf{Z}_i$), but nevertheless, once $\boldsymbol{\beta}_i$ has been computed, computing $\mathbf{PZ}_i = \mathbf{Z}_i - \mathbf{G}\boldsymbol{\beta}_i$ requires just a low-rank correction of $\mathbf{Z}_i$ in each subdomain because only a few columns of $\mathbf{G}$ are nonzero in each subdomain.

Preservation of locality in computing $\mathbf{FW}_i$ As already explained in the last item of the discussion in Section 3.4, the costly part in computing $\mathbf{FW}_i$ comes down to the computation of $\mathbf{FPZ}_i = \mathbf{FZ}_i - \mathbf{FG}\boldsymbol{\beta}_i$, and this can again be obtained with local low-rank corrections of $\mathbf{FZ}_i$ using $(\mathbf{FG})$ that has been computed at the initialization phase of S-FETI.

Optimization of the orthogonalization procedure Once a set of vectors $\mathbf{PZ}_i$ has been computed, it must be $\mathbf{F}$-orthogonalized to compute the new set of search directions $\mathbf{W}_i$. Instead of using a modified Gram–Schmidt procedure that requires many Message Passing Interface (MPI) reductions of dimension 1, $(\mathbf{PZ}_i)^T\mathbf{FPZ}_i$ can be computed by computing the local contribution of each subdomain, using Basic Linear Algebra Subprograms of level 3 (BLAS3) kernels, and only one MPI reduction to compute all the entries at once. Then we use a Cholesky factorization of $\mathbf{PZ}_i^T\mathbf{FPZ}_i$ to compute the new set of $\mathbf{F}$-orthonormal search directions $\mathbf{W}_i$. The complete $\mathbf{F}$ orthogonalization with the previous search direction vectors can be performed by block as well, with the same kind of optimization using local BLAS3 kernels and reducing the number of reduction operations compared with the standard modified Gram–Schmidt method.

In order to evaluate the method, we consider several test cases of 2D linear elasticity on a square domain clamped on one side and with imposed displacements on the opposite side. They are presented in Figure 6. We use either a regular checkerboard decomposition into 10×10 regular square subdomains each containing 180 000 degrees of freedom (the global problem is then more than 17 M degrees of freedom large once duplicated degrees of freedom have been removed) or a decomposition into slices (for two different orientations of the heterogeneity bands, see Figure 6) with 100 subdomains for the same global problems. The heterogeneous stiff stripes are either quasi-

DOI: 10.1002/nme

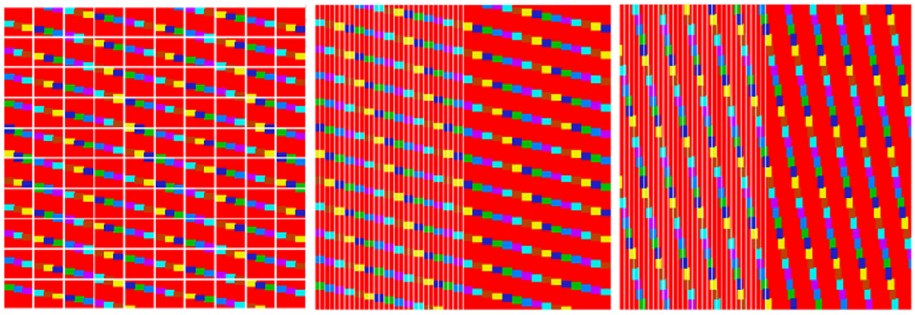


Figure 6. Three decompositions with heterogeneous materials.

Table X. CPU performance of FETI and S-FETI for 2D elasticity problems (Figure 6).

Decomp.	Poisson	Solver	# Iterations	# Search directions	Max # local solutions	Time (s)
Checker	0.4999	FETI	233	233	466	991
		S-FETI	46	4600	276	320
Slices 1	0.4999	FETI	> 800	> 800	> 1600	> 7300
		S-FETI	152	15200	608	4653
Slices 2	0.4999	FETI	> 800	> 800	> 1600	> 7300
		S-FETI	144	14400	576	4455
Slices 1	0.45	FETI	> 800	> 800	> 1600	> 7300
		S-FETI	48	4800	192	493
Slices 2	0.45	FETI	409	409	818	1979
		S-FETI	36	3600	144	363

FETI, finite element tearing and interconnection; S-FETI, simultaneous FETI; B-FETI, block FETI.

orthogonal to the interfaces, as in the figure at the center, or quasi-parallel to the interfaces as in the right. So, all the features that lead to poor convergence of the standard FETI method are present.

All subdomains are heterogeneous. The model features a composite material with a Young's modulus of 1 for the matrix, which may be nearly incompressible (Poisson ratio 0.45 or 0.4999), containing rigid stripes made of various blocks randomly affected with a Young's modulus of 10, 10^2, 10^3 10^4, or 10^5. Because the stripes are not parallel to the borders of the square, each interface features simultaneously various degrees of heterogeneity along and across it. In the two cases of decomposition in slices, the aspect ratio of the subdomains is bad. The decomposition is the same in both cases, so all subdomains except the first are floating.

The tests are conducted on a cluster of 2.6-GHz eight-core Xeon processors connected by a gigabit Ethernet network. Each subdomain is allocated to four cores (two subdomains per processor). Intel fortran compiler and MKL Pardiso solver are used.

The FETI iterations are performed until both the relative error $\|\mathbf{u}_i - \mathbf{u}_{i-1}\|/\|\mathbf{u}_i\|$ and the relative residual $\|(\mathbf{K}\mathbf{u}_i - \mathbf{f})\|/\|\mathbf{f}\|$ for the global problem are smaller than 10^{-6}. For both FETI and S-FETI, the super-lumped scaling and rigid body projections are used.

Table X presents the results for the different test cases. For FETI and S-FETI, we compare the number of iterations needed to achieve convergence, the number of generated search directions (which is equal to the number of iterations for FETI and to the number of iterations multiplied by the number of subdomains for S-FETI), the maximal number of local (Dirichlet or Neumann) resolutions on the subdomains, and the wallclock time. In the cases where the number of FETI iterations is more than 800, the iterations were stopped before convergence was reached (the order of magnitude of the residual was still greater than 10^{-2}).

All these test cases show that the S-FETI method is more robust than the standard FETI method for this kind of highly heterogeneous problems. Even in the case of the checkerboard decomposition where the aspect ratio of the subdomains is good, the S-FETI method is much faster than the FETI method. This case also clearly shows the benefit related to improved performance due to multiple forward–backward substitutions. The ratio between the numbers of local solutions for FETI and S-

DOI: 10.1002/nme

FETI is smaller than 2, whereas the ratio between the wallclock times is greater than 3, although the reconjugation procedure is more arithmetically intensive for S-FETI, because the total number of search directions is much larger.

6. CONCLUSION

A well-known problem when applying domain decomposition techniques to real engineering problems is the presence of heterogeneities along the interfaces. Classical preconditioning techniques perform very poorly in such cases. The FETI–Geneo was proposed lately to ensure robustness in those cases, by precomputing troublesome interface modes and including them in an auxiliary coarse grid. That method, however, requires costly pre-processing.

In this paper, we proposed two block strategies with the purpose of enriching the search space at every iteration on the interface problem. The methods were discussed and tested for the FETI method. Nevertheless, applying the same concepts to other nonoverlapping methods (e.g., BDD, FETI-DP, or BDDC) is straightforward.

The first method, S-FETI, exploits the additive structure of the preconditioner to generate a family of search directions and let the solver choose the best way to combine them instead of just computing their sum. The extra cost of the method is limited because sparsity can be exploited when searching for the solution in the multiple directions. One drawback of the S-FETI method is that it is no longer a genuine CG method and full orthogonalization of the search directions is necessary. This drawback is, however, a nonissue in practice because full orthogonalization is required even for CG.

The second method, B-FETI, exploits the additive structure of the problem to generate a family of right-hand sides to be solved by a block CG. Full reorthogonalization is no more mandatory, but this method requires many more local Dirichlet and Neumann solves than the classical FETI.

Both methods have very good computational properties because one iteration involves as many exchanges as classical FETI (but with larger amount of data) and they rely on block resolutions.

The assessments showed that both methods are much more robust than classical FETI for highly heterogeneous problems, for decompositions with jagged interfaces and for subdomains with bad aspect ratio. CPU time assessments of the S-FETI method showed that it can lead to very significant gains compared with standard FETI methods.

The methods need to be assessed on larger class of problems in particular in the cases where the decomposition involves a very large number of subdomains. It is expected that, because they imply highly optimized operations conducted on blocks of vectors, the proposed methods should be interesting default strategies for people concerned by the robustness of their domain decomposition solvers.

ACKNOWLEDGEMENTS

This work was financially supported by CONICYT through project Fondecyt 3150090.

REFERENCES

1. Farhat C, Roux FX. A method of finite element tearing and interconnecting and its parallel solution algorithm. *International Journal for Numerical Methods in Engineering* 1991; **32**(6):1205.
2. Mandel J. Balancing domain decomposition. *Communications in Numerical Methods in Engineering* 1993; **9**(3): 233–241.
3. Bhardwaj M, Day D, Farhat C, Lesoinne M, Pierson K, Rixen D. Application of the FETI method to ASCI problems: scalability results on a thousand-processor and discussion of highly heterogeneous problems. *International Journal for Numerical Methods in Engineering* 2000; **47**(1-3):513–536.
4. Brands D, Klawonn A, Rheinbach O, Schröder J. Modelling and convergence in arterial wall simulations using a parallel FETI solution strategy. *Computer Methods in Applied Mechanics and Engineering* 2008; **11**(5):569–583.
5. Gosselet P, Rey C. Non-overlapping domain decomposition methods in structural mechanics. *Archives of Computational Methods in Engineering* 2006; **13**(4):515–572.
6. Farhat C, Rixen D. A new coarsening operator for the optimal preconditioning of the dual and primal domain decomposition methods: application to problems with severe coefficient jumps. In *Proceedings of the Seventh Copper*

DOI: 10.1002/nme

Mountain Conference on Multigrid Methods, Duane Melson N, McCormick SF, Manteuffel TA, Douglas MCC (eds).: NASA, Langley Virginia US, 1995; 301–316.
7. Rixen D, Farhat C. A simple and efficient extension of a class of substructure based preconditioners to heterogeneous structural mechanics problems. *International Journal for Numerical Methods in Engineering* 1999; **44**(4):489–516.
8. Klawonn A, Rheinbach O. Robust FETI-DP methods for heterogeneous three dimensional elasticity problems. *Computer Methods in Applied Mechanics and Engineering* 2007; **196**(8):1400–1414.
9. Klawonn A, Rheinbach O, Widlund OB. An analysis of a FETI–DP algorithm on irregular subdomains in the plane. *SIAM Journal on Numerical Analysis* 2008; **46**(5):2484–2504.
10. Dryja M, Sarkis MV, Widlund OB. Multilevel Schwarz methods for elliptic problems with discontinuous coefficients in three dimensions. *Numerische Mathematik* 1996; **72**(3):313–348.
11. Mandel J, Brezina M. Balancing domain decomposition for problems with large jumps in coefficients. *Mathematics of Computation* 1996; **65**(216):1387–1401.
12. Dohrmann CR, Widlund OB. An overlapping Schwarz algorithm for almost incompressible elasticity. *SIAM Journal on Numerical Analysis* 2009; **47**(4):2897–2923.
13. Dohrmann CR, Widlund OB. Hybrid domain decomposition algorithms for compressible and almost incompressible elasticity. *International Journal for Numerical Methods in Engineering* 2010; **82**(2):157–183.
14. Pechstein C, Scheichl R. Scaling up through domain decomposition. *Applicable Analysis* 2009; **88**(10-11): 1589–1608.
15. Pechstein C, Scheichl R. Analysis of FETI methods for multiscale PDEs. *Numerische Mathematik* 2008; **111**(2): 293–333.
16. Gippert S, Klawonn A, Rheinbach O. Analysis of FETI-DP and BDDC for linear elasticity in 3D with almost incompressible components and varying coefficients inside subdomains. *SIAM Journal on Numerical Analysis* 2012; **50**(5):2208–2236.
17. Pechstein C, Scheichl R. Analysis of FETI methods for multiscale PDEs. Part II: interface variation. *Numerische Mathematik* 2011; **118**(3):485–529.
18. Pechstein C, Sarkis M, Scheichl R. New theoretical coefficient robustness results for FETI-DP. In *Domain Decomposition Methods in Science and Engineering XX*, Lecture Notes in Computational Science and Engineering. Springer-Verlag: Heidelberg, 2011; 327–335.
19. Pechstein C. *Finite and Boundary Element Tearing and Interconnecting Solvers for Multiscale Problems*, Lecture Notes in Computational Science and Engineering, vol. 90. Springer-Verlag: Berlin Heidelberg, 2013.
20. Pechstein C, Scheichl R. Weighted Poincaré inequalities. *IMA Journal on Numerical Analysis* 2013; **33**(2):652–686.
21. Beirão da Veiga L, Pavarino LF, Scacchi S, Widlund OB, Zampini S. Isogeometric BDDC preconditioners with deluxe scaling. *SIAM Journal of Scientific Computing* 2014; **36**(3):A1118–A1139.
22. Nataf F, Xiang H, Dolean V, Spillane N. A coarse space construction based on local Dirichlet-to-Neumann maps. *SIAM Journal of Scientific Computing* 2011; **33**(4):1623–1642.
23. Dolean V, Nataf F, Scheichl R, Spillane N. Analysis of a two-level Schwarz method with coarse spaces based on local Dirichlet-to-Neumann maps. *Computational Methods in Applied Mathematics* 2012; **12**(4):391–414.
24. Spillane N, Dolean V, Hauret P, Nataf F, Pechstein C, Scheichl R. Abstract robust coarse spaces for systems of PDEs via generalized eigenproblems in the overlaps. *Numerische Mathematik* 2014; **126**(4):741–770.
25. Galvis J, Efendiev Y. Domain decomposition preconditioners for multiscale flows in high contrast media: reduced dimension coarse spaces. *Multiscale Modeling and Simulation* 2010; **8**(5):1621–1644.
26. Efendiev Y, Galvis J, Lazarov R, Willems J. Robust domain decomposition preconditioners for abstract symmetric positive definite bilinear forms. *ESAIM Mathematical Modelling and Numerical Analysis* 2012; **46**(5):1175–1199.
27. Mandel J, Sousedík B. Adaptive selection of face coarse degrees of freedom in the BDDC and the FETI-DP iterative substructuring methods. *Computer Methods in Applied Mechanics and Engineering* 2007; **196**(8):1389–1399.
28. Sousedík B, Šístek J, Mandel J. Adaptive-multilevel BDDC and its parallel implementation. *Computing* 2013; **95**(12):1087–1119.
29. Spillane N, Rixen DJ. Automatic spectral coarse spaces for robust FETI and BDD algorithms. *International Journal for Numerical Methods in Engineering* 2013; **95**(11):953–990.
30. Klawonn A, Radtke P, Rheinbach O. Adaptive coarse spaces for BDDC with a transformation of basis. *Twenty Second International Conference on Domain Decomposition Methods*, Università della Svizzera Italiana, Lugano, Switzerland, 2014.
31. Klawonn A, Radtke P, Rheinbach O. FETI-DP methods with an adaptive coarse space. *SIAM Journal on Numerical Analysis* 2015; **53**(1):297–320.
32. Spillane N. Robust domain decomposition methods for symmetric positive definite problems. *Ph.D. Thesis*, Thèse de l'Ecole doctorale de Mathmatiques de Paris centre, Laboratoire Jacques Louis Lions, Université Pierre et Marie Curie, Paris, 2014.
33. Rixen D. Substructuring and dual methods in structural analysis. *Ph.D. Thesis*, Université de Liège, Belgium, Collection des Publications de la Faculté des Sciences appliquées, n.175, 1997.
34. Rixen DJ. A domain decomposition interface solver with multiple direction of descent for heterogeneous problems. *84th Annual Meeting of the International Association of Applied Mathematics and Mechanics, GAMM, Novi-Sad, Serbia (oral presentation only)*, University of Novi Sad, Serbia, 2013.
35. Gosselet P, Rixen DJ, Rey C. A domain decomposition strategy to efficiently solve structures containing repeated patterns. *International Journal for Numerical Methods in Engineering* 2009; **78**(7):828–842.

DOI: 10.1002/nme

36. Greif C, Rees T, Szyld DB. MPGMRES: a generalized minimum residual method with multiple preconditioners. *Technical Report 11-12-23*, Department of Mathematics, Temple University, 2011. Revised September 2012 and January 2014. Also available as Technical Report TR-2011-12, Department of Computer Science, University of British Columbia.
37. O'Leary DP. The block conjugate gradient algorithm and related methods. *Linear Algebra and its Applications* 1980; **29**:293–322.
38. Farhat C, Pierson K, Lesoine M. The second generation FETI methods and their application to the parallel solution of large-scale linear and geometrically non-linear structural analysis problems. *Computer Methods in Applied Mechanics and Engineering* 2000; **184**(2-4):333–374.
39. Farhat C, Lesoinne M, LeTallec P, Pierson K, Rixen D. FETI-DP: a dual-primal unified FETI method - part I: a faster alternative to the two-level FETI method. *International Journal for Numerical Methods in Engineering* 2001; **50**(7):1523–1544.
40. Dohrmann CR. A preconditioner for substructuring based on constrained energy minimization. *SIAM Journal of Scientific Computing* 2003; **25**(1):246–258 (electronic).
41. Rixen DJ, Farhat C, Tezaur R, Mandel J. Theoretical comparison of the FETI and algebraically partitioned FETI methods, and performance comparisons with a direct sparse solver. *International Journal for Numerical Methods in Engineering* 1999; **46**(4):501–533.
42. Farhat C, Mandel J. The two-level FETI method for static and dynamic plate problems. I. An optimal iterative solver for biharmonic systems. *Computer Methods in Applied Mechanics and Engineering* 1998; **155**(1-2):129–151.
43. Farhat C, Chen PS, Mandel J, Roux FX. The two-level FETI method. II. Extension to shell problems, parallel implementation and performance results. *Computer Methods in Applied Mechanics and Engineering* 1998; **155**(1-2): 153–179.
44. Gosselet P, Rey C, Pebrel J. Total and selective reuse of Krylov subspaces for the solution to a sequence of nonlinear structural problems. *International Journal for Numerical Methods in Engineering* 2013; **94**(1):60–83.
45. Rey V, Rey C, Gosselet P. A strict error bound with separated contributions of the discretization and of the iterative solver in non-overlapping domain decomposition methods. *Computer Methods in Applied Mechanics and Engineering* 2013; **270**(1):293–303.
46. Bridson R, Greif C. A multipreconditioned conjugate gradient algorithm. *SIAM Journal on Matrix Analysis and Applications* 2006; **27**(4):1056–1068 (electronic).
47. Saad Y. *Iterative Methods for Sparse Linear Systems* (2nd edn). Society for Industrial and Applied Mathematics (SIAM): Philadelphia, PA, 2003.
48. Greif C, Rees T, Szyld DB. Additive Schwarz with variable weights. In *Domain Decomposition Methods in Science and Engineering XXI*, vol. 98, Erhel J, Gander MJ, Halpern L, Pichot G, Sassi T, Widlund O (eds)., Lecture Notes in Computational Science and Engineering. Springer International Publishing: Cham, Switzerland, 2014; 779–787.
49. Nikishin AA, Yeremin AY. Variable block CG algorithms for solving large sparse symmetric positive definite linear systems on parallel computers. I. General iterative scheme. *SIAM Journal on Matrix Analysis and Applications* 1995; **16**(4):1135–1153.
50. Brandt A, Brannick J, Kahl K, Livshits I. Bootstrap AMG. *SIAM Journal of Scientific Computing* 2011; **33**(2): 612–632.
51. Gosselet P, Rey C, Rixen D. On the initial estimate of interface forces in FETI methods. *Computer Methods in Applied Mechanics and Engineering* 2003; **192**(25):2749–2764.
52. Karypis G, Kumar V. METIS: a software package for partitioning unstructured graphs, partitioning meshes, and computing fill-reducing orderings of sparse matrices, Department of Computer Science, University of Minnesota, 1998. http://glaros.dtc.umn.edu/gkhome/views/metis.
53. Vereecke B, Bavestrello H, Dureisseix D. An extension of the FETI domain decomposition method for incompressible and nearly incompressible problems. *Computer Methods in Applied Mechanics and Engineering* 2003; **192**(31):3409–3429.
54. Zsaki A, Rixen DJ, Paraschivoiu M. A substructured based iterative inner solver coupled with Uzawa's algorithm for the Stokes problem. *International Journal for Numerical Methods in Fluids* 2003; **43**:215–230.
55. Schenk O, Gartner K. Solving unsymmetric sparse systems of linear equations with PARDISO. *Journal of Future Generation Computer Systems* 2004; **20**(3):475–487.

DOI: 10.1002/nme

F Article [Parret-Fréaud *et al.*, 2016], construction parallèle de champs admissibles

Cet article présente la version la plus aboutie de la reconstruction parallèle de champs admissibles en vue du calcul parallèle de l'erreur de discrétisation. Il précise le papier original [Parret-Fréaud *et al.*, 2010]. On insiste sur le soin particulier à apporter aux points multiples.

Par ailleurs on investigue le cas des structures hétérogènes et on montre comment améliorer l'estimateur séquentiel pour qu'il donne des performances équivalentes à l'estimateur parallèle dans les cas où ce dernier fonctionnait mieux. Enfin cette étude permet d'étendre les résultats sur la séparation des sources d'erreur (discrétisation et décomposition de domaine) aux méthodes FETI-DP et BDDC.

Concernant la reconstruction des champs admissibles par sous-domaine, une généralisation de la méthode EET est présentée dans [Rey *et al.*, 2014b].

INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 0000; **00**:1–20
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/nme

Improved recovery of admissible stress in domain decomposition methods — application to heterogeneous structures and new error bounds for FETI-DP

A. Parret-Fréaud[2], V. Rey[1*], P. Gosselet[1], C. Rey[2]

[1] *LMT-Cachan / Ecole Normale Supérieure de Cachan, CNRS, Université Paris Saclay 61, avenue du président Wilson, 94235 Cachan, France,*

[2] *Safran Tech, rue des Jeunes Bois Châteaufort CS 80112,78772 Magny les Hameaux, France*

SUMMARY

This paper investigates the question of the building of admissible stress field in a substructured context. More precisely we analyze the special role played by multiple points. This study leads to (1) an improved recovery of the stress field, (2) an opportunity to minimize the estimator in the case of heterogeneous structures (in the parallel and sequential case), (3) a procedure to build admissible fields for FETI-DP and BDDC methods leading to an error bound which separates the contributions of the solver and of the discretization. Copyright © 0000 John Wiley & Sons, Ltd.

Received ...

KEY WORDS: Verification; Domain decomposition methods; Heterogeneity; Multiple points; FETI-DP

1. INTRODUCTION

The strong mathematical properties of the finite element method [1] for the approximation of the solution of mechanical problems are unfortunately not sufficient to precisely guarantee a priori the quality of the computed fields. A posteriori verification aims at providing a numerical estimate of the distance between the unknown exact solution and the calculation. Several types of error estimators exist.

Estimators based on the lack of regularity [2] of the stress field are often efficient and easy to implement but they may underestimate the error. Strict estimators can be obtained at the price of the evaluation of constants that depend on the shape of the domain, which is not very practical. Estimators based on the error in constitutive equation [3, 4], or on the equilibrated residuals [5, 6] (which in fact are equivalent approaches) give strict bounds without constant but they require the computation of a statically admissible (equilibrated) stress field.

Statically admissible stress field can be obtained by separate dual analysis [7], or by a post-processing of the finite element displacement. Among these post-processors, we can cite the Element Equilibration Technique (EET) [8, 9] and its recent variant [10, 11, 12], and flux-free techniques based on partition of unity [13, 14, 15, 16, 17, 18]. Note that whatever the approach, the numerical cost is never negligible.

*Correspondence to: LMT-Cachan / Ecole Normale Supérieure de Cachan, CNRS, Université Paris Saclay 61, avenue du président Wilson, 94235 Cachan, France, valentine.rey@lmt.ens-cachan.fr

In [19], it was shown that the methods to post-process balanced stress fields could be embedded within the framework of non-overlapping domain decomposition [20]. In particular, when using the balancing domain decomposition (BDD [21]) or the finite element tearing and interconnecting (FETI [22]), specific displacement and traction fields can be generated and as inputs of a parallel procedure for the computation of statically admissible fields. Those results are recalled in Section 2.

The first aim of this article, which is a subject of Section 3, is to investigate more deeply the proposed procedure and in particular to emphasize the role played by nodes shared by several subdomains. These nodes, often referred to as multiple points or cross-points, require specific attention for the computation of statically admissible stress fields. In particular, in case of strong heterogeneities, ignoring the importance of multiple points may lead to a defective error estimation. To tackle this difficulty, we analyze the role of the multiple points in error estimation and take advantage of the optimization problem it triggers to better the reconstruction of the admissible stress field. In Section 4, we show how the classical procedure EET to build statically admissible fields can be improved in order to take into account strong heterogeneity. The result is a more accurate sequential error estimator. We highlight the strong similarity between the optimization at the multiple points and the optimization in this sequential estimator. Then in Section 5, we give numerical results of the optimized procedures applied to a two-dimensional mechanical problem with strong heterogeneities. Finally, we extend the parallel error estimation procedure to the FETI-DP algorithm [23, 24] and BDDC algorithm by a specific treatment of multiple points for the construction of balanced tractions. The bound separates the contributions of the solver and of the discretization.

2. PRINCIPLE OF ERROR ESTIMATION IN SUBSTRUCTURED PROBLEMS

This section recalls the main principles of our approach for a posteriori error estimation in the framework of non-overlapping domain decomposition, with application to FETI [22] and BDD [21] as presented in [19].

2.1. Reference mechanical problem and admissibility spaces

Let us consider the static equilibrium of a structure which occupies the open polyhedral domain $\Omega \subset \mathbb{R}^d$ and which is submitted to given body force f, traction force g on $\partial_f\Omega$ and displacement u_d on the complementary part $\partial_u\Omega$ such that $\text{meas}(\partial_u\Omega) \neq 0$. We assume the structure undergoes small perturbations and that the material is linear elastic, characterized by the Hooke's tensor $\mathbb{H}$. In the following, u is the unknown displacement field, $\varepsilon(u)$ is the symmetric part of the gradient and σ is the Cauchy stress tensor.

For an open subset $\omega \subset \Omega$, we note $\partial_u\omega = \partial_u\Omega \cap \partial\omega$ and $\partial_f\omega = \partial_f\Omega \cap \partial\omega$, and we introduce the following subspaces of *kinematically admissible* ($\mathcal{K}a$) and *statically admissible* ($\mathcal{S}a$) fields:

$$\mathcal{K}a(\omega) = \left\{ v \in \left[H^1(\omega)\right]^d \text{ such that } \text{tr}(v)_{|\partial_u\omega} = u_d \right\}, \tag{1a}$$

$$\mathcal{K}a^0(\omega) = \left\{ v \in \left[H^1(\omega)\right]^d \text{ such that } \text{tr}(v)_{|\partial_u\omega} = 0 \right\}, \tag{1b}$$

$$\mathcal{K}a^{00}(\omega) = \left\{ v \in \left[H^1(\omega)\right]^d \text{ such that } \text{tr}(v)_{|\partial\omega\setminus\partial_f\omega} = 0 \right\}, \tag{1c}$$

$$\mathcal{S}a(\omega) = \left\{ \begin{array}{c} \tau \in \left[L^2(\omega)\right]^{d\times d}_{sym} \text{ such that } \forall\ v \in \mathcal{K}a^{00}(\omega) \\ \displaystyle\int_\omega \tau : \varepsilon(v)\, dx = \int_\omega f.v\, dx + \int_{\partial_f\omega} g.v\, dS \end{array} \right\} \tag{1d}$$

where tr is the trace operator. We introduce the following functional called *error in constitutive relation* for a pair of displacement and stress fields $(v,\tau) \in \left[H^1(\omega)\right]^d \times$

DOI: 10.1002/nme

$\left[L^2(\omega)\right]^{d^2}_{sym}$:

$$\boldsymbol{e}_{\mathrm{CR},\omega}(v,\tau) = \|\tau - \mathbb{H} : \varepsilon(v)\|_{\mathbb{H}^{-1},\omega}, \quad \text{with } \| \bullet \|_{\mathbb{H}^{-1},\omega} = \left[\int_\omega \bullet : \mathbb{H}^{-1} : \bullet \, dx\right]^{1/2} \tag{2}$$

The mechanical problem set on Ω may be formulated as:

$$\text{Find } (u,\sigma) \in \mathcal{K}a(\Omega) \times \mathcal{S}a(\Omega) \text{ verifying } \boldsymbol{e}_{\mathrm{CR},\Omega}(u,\sigma) = 0, \tag{3}$$

Under the given assumptions, the solution to this problem exists and is unique, so that σ matches the singleton defined by $\mathcal{S}a(\Omega) \cap \mathbb{H} : \varepsilon\left(\mathcal{K}a(\Omega)\right)$.

Let us now consider a decomposition of domain Ω into open subsets $(\Omega^{(s)})_{1\leqslant s\leqslant \mathcal{N}_{\mathrm{sd}}}$ ($\mathcal{N}_{\mathrm{sd}}$ is the number of subdomains) so that $\Omega^{(s)} \cap \Omega^{(s')} = \varnothing$ for $s \neq s'$ and $\bar{\Omega} = \cup_s \bar{\Omega}^{(s)}$, and mark with superscript (s) the restriction on domain $\Omega^{(s)}$. Under this framework, kinematic and static admissibility on the whole structure may be restricted to each sub-structure $\Omega^{(s)}$ providing the verification of interface conditions, namely displacements continuity (4a) and balance of tractions (or action-reaction principle) (4b). Therefore, we have, for any globally admissible pair (v,τ):

$$v \in \mathcal{K}a(\Omega) \Leftrightarrow \left\{ \begin{array}{l} v^{(s)} \in \mathcal{K}a(\Omega^{(s)}),\ \forall s \\ \mathrm{tr}(v^{(s)}) = \mathrm{tr}(v^{(s')}) \text{ on } \Upsilon^{(s,s')},\ \forall (s,s') \end{array} \right. , \tag{4a}$$

$$\tau \in \mathcal{S}a(\Omega) \Leftrightarrow \left\{ \begin{array}{l} \tau^{(s)} \in \mathcal{S}a(\Omega^{(s)}),\ \forall s \\ \tau^{(s)} \cdot n^{(s)} + \tau^{(s')} \cdot n^{(s')} = 0 \text{ on } \Upsilon^{(s,s')},\ \forall (s,s') \end{array} \right. , \tag{4b}$$

where $\Upsilon^{(s,s')} = \partial\Omega^{(s)} \cap \partial\Omega^{(s')}$ is the interface between $\Omega^{(s)}$ and $\Omega^{(s')}$.

Of course, the error in constitutive relation can be written as a sum of local contributions:

$$\forall (v,\tau) \in \left[H^1(\omega)\right]^d \times \left[L^2(\omega)\right]^{d^2}_{sym}, \boldsymbol{e}_{\mathrm{CR},\Omega}(v,\tau)^2 = \sum_{s=1}^{\mathcal{N}_{\mathrm{sd}}} \left(\boldsymbol{e}_{\mathrm{CR},\Omega^{(s)}}(v^{(s)},\tau^{(s)})\right)^2.$$

2.2. Basics on error estimation

Our error estimation technique is based on the following property, called the Pragger-Synge theorem [3]:

$$\forall (\hat{u},\hat{\sigma}) \in \mathcal{K}a(\Omega) \times \mathcal{S}a(\Omega),\ \|\varepsilon(u-\hat{u})\|^2_{\mathbb{H},\Omega} + \|\sigma - \hat{\sigma}\|^2_{\mathbb{H}^{-1},\Omega} = \boldsymbol{e}^2_{\mathrm{CR},\Omega}(\hat{u},\hat{\sigma}) \tag{5}$$

where (u,σ) solves the reference problem (3). We choose to measure the error in displacement $e = u - \hat{u}$, and introduce the following norm: $|||e|||_\Omega := \|\varepsilon(e)\|_{\mathbb{H},\Omega} \leqslant \boldsymbol{e}_{\mathrm{CR},\Omega}(\hat{u},\hat{\sigma})$. Thus for any admissible approximation of the solution $(\hat{u},\hat{\sigma}) \in \mathcal{K}a(\Omega) \times \mathcal{S}a(\Omega)$, the error in constitutive relation is a computable upper bound of the error. Then the problem of the error estimation can be addressed through the ability to build kinematically and statically admissible approximations.

In the case of the finite element approach (of the monolithic problem), the displacement approximation u_h solves the following problem:

$$\begin{array}{l} \text{Find } u_h \in \mathcal{K}a_h(\Omega) \text{ such that } \sigma_h = \mathbb{H} : \varepsilon(u_h) \text{ satisfies} \\ \displaystyle\int_\Omega \sigma_h : \varepsilon(v_h)\, dx = \int_\Omega f.v_h\, dx + \int_{\partial_f\Omega} g.v_h\, dS,\ \forall\ v_h \in \mathcal{K}a^0_h(\Omega). \end{array} \tag{6}$$

where $\mathcal{K}a_h(\Omega)$ is a finite dimension subspace of $\mathcal{K}a(\Omega)$ so that we set $\hat{u} = u_h$. Note that u_h takes the following form $u_h = \boldsymbol{\varphi}_h \mathbf{u}$ where $\boldsymbol{\varphi}_h$ is the matrix of shape functions and $\mathbf{u}$ the vector of nodal displacements.

Prepared using nmeauth.cls
DOI: 10.1002/nme

The solution $\sigma_h = \mathbb{H} : \varepsilon(u_h)$ unfortunately does not belong to $\mathcal{S}a(\Omega)$, except in trivial cases. The construction of the admissible stress field $\hat{\sigma}_h \in \mathcal{S}a(\Omega)$ requires to apply a whole procedure (often referred to as recovery of balanced residual). It is a crucial point since the sharpness of error estimation strongly depends on the quality of the $\hat{\sigma}_h$.

We note by $\mathcal{F}_{\text{eq}}$ the algorithm chosen to build an admissible stress field $\hat{\sigma}_h$ from the finite element one σ_h. $\mathcal{F}_{\text{eq}}$ also takes as inputs the continuous representation of the imposed body force f and of the traction force g.

$$\hat{\sigma}_h = \mathcal{F}_{\text{eq}}(\sigma_h, f, g) \in \mathcal{S}a(\Omega)$$

In our applications, the chosen algorithm is either the Element Equilibration Technique [8] or the Flux-free technique [14] with p-refinement for elements problems (three degrees higher polynomial basis [25]). The general principles of the EET technique are recalled in section 4 where an improvement is proposed in order to take into account the material heterogeneity.

2.3. Admissible field recovery for FETI and BDD domain decomposition

We assume that the mesh and the decomposition of Ω are conforming so that (i) each element only belongs to one subdomain and (ii) nodes are matching on the interfaces. Under this assumption, each degree of freedom is either located inside a subdomain or on its boundary $\Gamma^{(s)} = \cup_{s'} \Upsilon^{(s,s')}$ where it is shared with at least one neighboring subdomain.

In order to decouple the subdomains, we introduce $\boldsymbol{\lambda}_b^{(s)}$ the vector of unknown nodal reactions imposed on the boundary of $\Omega^{(s)}$ by its neighbors. The finite element equilibrium of subdomain s then writes:

$$\mathbf{K}^{(s)}\mathbf{u}^{(s)} = \mathbf{f}^{(s)} + \mathbf{t}^{(s)^T}\boldsymbol{\lambda}_b^{(s)} \tag{7}$$

where $\mathbf{K}^{(s)}$ is the stiffness matrix, $\mathbf{f}^{(s)}$ is the vector of generalized forces and $\mathbf{t}^{(s)}$ is the discrete trace operator (which extracts the boundary values from a vector defined on the whole subdomain) and thus $\mathbf{t}^{(s)^T}$ is the extension by zero operator.

To complete the system, we need the discrete counterparts of the interface equations: continuity of displacement (4a) and balance of forces (4b). This is done by the introduction of assembling operators $\mathbf{A}$ and $\mathbf{B}$ (they will play an important role in our analysis and will be thoroughly described in the next section):

$$\sum_s \mathbf{A}^{(s)}\boldsymbol{\lambda}_b^{(s)} = \mathbf{0} \tag{8a}$$

$$\sum_s \mathbf{B}^{(s)}\mathbf{t}^{(s)}\mathbf{u}^{(s)} = \mathbf{0} \tag{8b}$$

Note that when there are only two subdomains, and thus one simple interface, these equations can be written as: $\mathbf{u}_b^{(1)} - \mathbf{u}_b^{(2)} = \mathbf{0}$ and $\boldsymbol{\lambda}_b^{(1)} + \boldsymbol{\lambda}_b^{(2)} = \mathbf{0}$.

In [19], it was proved that when using the most classical iterative solvers for system (7,8a,8b), namely FETI [22] and BDD [21], it is possible to build at no extra cost the following fields, whatever the convergence state of the solver:

- $\left(\mathbf{u}_D^{(s)}\right)$: vectors of displacements continuous at the interface ($\sum_s \mathbf{B}^{(s)}\mathbf{t}^{(s)}\mathbf{u}_D^{(s)} = \mathbf{0}$) which define a globally admissible field: $\hat{u}_D = (\boldsymbol{\varphi}_h^{(s)}\mathbf{u}_D^{(s)})_s \in \mathcal{K}a(\Omega)$.
- $\left(\boldsymbol{\lambda}_N^{(s)}\right)$: vectors of nodal reaction which are balanced at the interface, so that $\sum_s \mathbf{A}^{(s)}\boldsymbol{\lambda}_N^{(s)} = \mathbf{0}$, and which make the local equilibrium (7) well posed Neumann problems.

DOI: 10.1002/nme

- $\left(\mathbf{u}_N^{(s)}\right)$: vectors of displacements associated with the resolution of (7) with imposed reactions $\left(\boldsymbol{\lambda}_N^{(s)}\right)$. Let $u_N = (\boldsymbol{\varphi}_h^{(s)}\mathbf{u}_N^{(s)})_s$, the field $\sigma_N = (\mathbb{H} : \varepsilon(u_N))$ is a global stress field which satisfies the finite element equilibrium.

As a consequence, it was proposed, still in [19], to use σ_N as a starting point for the computation of a global statically admissible stress field. In order to use classical recovery algorithms, one simply had to give a continuous representation $g_\Gamma^{(s)} \in L^2(\Gamma^{(s)})$ of the traction field on the boundary compatible with the nodal reactions $\left(\boldsymbol{\lambda}_N^{(s)}\right)$:

$$\boldsymbol{\lambda}_N^{(s)^T} = \int_{\Gamma^{(s)}} g_\Gamma^{(s)} \cdot \boldsymbol{\varphi}_{hb}^{(s)} dS \tag{9}$$

Then one simply used the recovery algorithm in parallel:

$$\hat{\sigma}_N = \left(\mathcal{F}_{\text{eq}}\left(\sigma_N^{(s)}, f^{(s)}, g^{(s)}, g_\Gamma^{(s)}(\boldsymbol{\lambda}_N^{(s)})\right)\right)_s$$

leading to the following inequality [19]:

$$|||u - \hat{u}_D|||^2 \leqslant \sum_{s=1}^{\mathcal{N}_{\text{sd}}} \left(\boldsymbol{e}_{\text{CR},\Omega^{(s)}}(\hat{u}_D^{(s)}, \hat{\sigma}_N^{(s)})\right)^2 \tag{10}$$

In fact, there are cases where the above approach is not satisfactory:

- There is a risk that in the vicinity of multiple points $g_\Gamma^{(s)}(\boldsymbol{\lambda}_N^{(s)})$ looses its balance ($g_{\Gamma|\Upsilon^{(s,s')}}^{(s)} + g_{\Gamma|\Upsilon^{(s,s')}}^{(s')} \neq 0$), because the contributions of the two (or more) neighbors are not correctly distinguished. This causes a theoretical loss (never met in practice) of the exactness of the error bounding.
- In the presence of heterogeneity, the parallel estimator may become unduly large compared to the sequential one.

The first problem is addressed in section 3, the second one in section 4 and assessments are presented in section 5. An important by-product of these developments, exposed in section 6, is the extension of all previous methods and results to FETI-DP and BDD-C algorithms which are very popular domain decomposition methods [24, 26].

Remark. In [27], it was proved that the following bounds with separated contributions could also be derived:

$$\begin{aligned} |||u - \hat{u}_D||| &\leqslant \|\mathbf{r}\|_M + \sqrt{\sum_{s=1}^{\mathcal{N}_{\text{sd}}} \left(\boldsymbol{e}_{\text{CR},\Omega^{(s)}}(u_N^{(s)}, \hat{\sigma}_N^{(s)})\right)^2} \\ |||u - u_N||| &\leqslant \|\mathbf{r}\|_M + \sqrt{\sum_{s=1}^{\mathcal{N}_{\text{sd}}} \left(\boldsymbol{e}_{\text{CR},\Omega^{(s)}}(u_N^{(s)}, \hat{\sigma}_N^{(s)})\right)^2} \end{aligned} \tag{11}$$

where $\|\mathbf{r}\|_M$ is a well-chosen norm of the solver's residual computed at each iteration, whereas the second term is governed by the discretization error.

3. RECOVERY OF ADMISSIBLE FIELDS IN THE PRESENCE OF MULTIPLE POINTS

In this section we present how multiple points need to be taken into account when recovering stress fields in parallel. To do so, we first show the impact of multiple points on assembling operators, then we present the improved recovery procedure and finally we show how an opportunity is left to optimize the estimator.

Prepared using nmeauth.cls
DOI: 10.1002/nme

3.1. Assembling operators

The assembling operators aim at connecting neighboring subdomains together. They rely on various descriptions of the interface between subdomains. An example is provided in figure 1 where operators are concatenated in a row $\mathbf{A} = \begin{pmatrix}\mathbf{A}^{(1)} & \dots & \mathbf{A}^{(\mathcal{N}_{sd})}\end{pmatrix}$. In this example, we use a crosspoint of multiplicity 4 (the node is shared by 4 subdomains) but all developments in this section remain valid whatever the multiplicity.

Since we assume conforming meshes and discretization at the interface, we can use a description of the substructures in terms of degrees of freedom. In the following expressions, the boundary operator ∂ gives the degrees of freedom associated with nodes on the boundary of the subdomain. We can define different sets of degrees of freedom on the interface. In all cases, degrees of freedom where Dirichlet conditions are imposed are excluded:

$$\begin{aligned}
&\Upsilon^{(i,j)} = (\partial\Omega^{(i)} \cap \partial\Omega^{(j)})\backslash\partial_u\Omega, \text{ interface between } \Omega^{(i)} \text{ and } \Omega^{(j)}\\
&\Gamma^{(i)} = \partial\Omega^{(i)}\backslash\partial\Omega = \bigcup_j \Upsilon^{(i,j)}, \text{ boundary of } \Omega^{(i)}\\
&\Upsilon_p = \bigcup_i \Upsilon^{(i)}, \text{ primal interface}
\end{aligned} \tag{12}$$

The first way to connect subdomains together is to locate their boundary $(\Gamma^{(i)})$ in the global interface Υ_p (also called primal interface). To do so we introduce the inclusion map of the local boundary into the global interface, which we call the primal assembly operator:

$$\mathbf{A}^{(s)} : \mathbb{R}^{\Gamma^{(s)}} \longrightarrow \mathbb{R}^{\Upsilon_p}. \tag{13}$$

$\mathbf{A}^{(s)}$ is a full rank matrix with $\#\Upsilon_p$ rows and $\#\Gamma^{(s)}$ columns, with one 1 per column and zeros elsewhere. Figure 1(a) shows an example for operators $\mathbf{A}^{(s)}$ in the simple case of four substructures with a primal interface constituted by five nodes.

The second type of connection is realized using any set of matrices $(\mathbf{B}^{(s)})$ such that:

$$\begin{aligned}
\sum_s \mathbf{B}^{(s)}{\mathbf{A}^{(s)}}^T &= 0,\\
\text{Rank}\left(\mathbf{B}^{(s)}\right) &= \#\Gamma^{(s)}.
\end{aligned} \tag{14}$$

$\mathbf{B}^{(s)}$ has $\#\Gamma^{(s)}$ columns and all $(\mathbf{B}^{(s)})_s$ have the same number of rows which shall be sufficiently large for each degree of freedom of $(\Gamma^{(s)})$ to be involved at least once. The rows of $(\mathbf{B}^{(s)})_s$ are called connecting relations and they are represented by the set Υ_d (also called dual interface).

Let $\mathbb{R}^\Gamma$ represent the space of vectors defined on the boundary of the subdomains; we have the following fundamental property:

$$\mathbb{R}^\Gamma = \text{Range}\left(\mathbf{A}^T\right) \overset{\perp}{\oplus} \text{Range}\left(\mathbf{B}^T\right). \tag{15}$$

In other words, a vector defined on the boundary of subdomains can be decomposed in a unique way as a the sum of a continuous (primal) contribution and a balanced (dual) contribution.

From a practical point of view, there exists many ways to construct the dual assembly operators $(\mathbf{B}^{(s)})$. We only present the most classical assembly operator used by solvers (simply written $\mathbf{B}^{(s)}$) and a second assembly operator more relevant for the recovery of admissible fields (written $\mathbf{B}_{\text{F}}^{(s)}$), we omit other possibilities [28].

The most classical dual assembly operator relies on the use of the following description of the dual interface:

$$\Upsilon_d = \Upsilon_d^C := \left(\Upsilon^{(i,j)}\right)_{i<j}$$

DOI: 10.1002/nme

and $\mathbf{B}^{(s)}$ is the signed inclusion map $\mathbb{R}^{\Gamma^{(s)}} \longrightarrow \mathbb{R}^{\Upsilon_d^C}$. The signs ensure that two neighbors have opposite contributions. With that construction, which is very easy to implement, even subdomains sharing one single node create a connection. This leads to many redundant rows in $\mathbf{B}$, namely $m(m-1)/2$ for a m-multiplicity node while only $(m-1)$ relationships are necessary to ensure the connectivity. However, it is well known that redundancies do not affect the resolution procedure [22]. The classical dual assembly operator is illustrated on figure 1(b).

A second way to define the dual assembly operator is to only keep, in the connectivity table interfaces with non-zero measure: these are faces in 3D (and edges in 2D).

$$\Upsilon_d = \Upsilon_d^F := \left\{ \Upsilon^{(i,j)},\ i<j,\ \text{with}\ \text{meas}(\Upsilon^{(i,j)}) \neq 0 \right\}$$

This dual assembly operator, written $\mathbf{B}_{\mathrm{F}}$, is illustrated on figure 1(c). Note that there is still one redundancy per multiple point. A specific basis of $\ker(\mathbf{B}_{\mathrm{F}}^T)$, associated with a cyclic stress, is illustrated in figure 2, we write it down $\mathbf{R}^{\circlearrowleft}$. Note that building $\mathbf{R}^{\circlearrowleft}$ is a rather simple operation which relies on an analysis of the mesh connectivity and of the decomposition.

3.2. Scaled assembling operators

The scaled assembling operators $(\tilde{\mathbf{A}}^{(s)})$ and $(\tilde{\mathbf{B}}^{(s)})$ play a role in the preconditioning of the domain decomposition methods. They are such that $\sum^{(s)} \tilde{\mathbf{A}}^{(s)} \mathbf{A}^{(s)^T} = \mathbf{I}$ and $\sum^{(s)} \mathbf{B}^{(s)} \tilde{\mathbf{B}}^{(s)^T} \mathbf{B}^{(j)} = \mathbf{B}^{(j)}$. In other words, scaled assembling operators are pseudo-inverses of assembling operators: $\tilde{\mathbf{A}} = \mathbf{A}^{T^+}$ and $\tilde{\mathbf{B}} = \mathbf{B}^{T^+}$. Many scaling being possible, in order to characterized them, they are associated with optimization problems:

$$\begin{aligned} &(\mathbf{u}^{(s)})_s \text{ given in } \mathbb{R}^{\Gamma},\ \mathbf{U} = \sum \tilde{\mathbf{A}}^{(s)} \mathbf{u}^{(s)} \Leftrightarrow \mathbf{U} = \arg\min_{\mathbb{R}^{\Upsilon_p}} \|\mathbf{u}^{(s)} - \mathbf{A}^{(s)^T} \mathbf{U}\|_{\mathcal{K}} \\ &(\boldsymbol{\lambda}^{(s)})_s \text{ given in } \mathbb{R}^{\Gamma},\ \boldsymbol{\Lambda} = \sum \tilde{\mathbf{B}}^{(s)} \boldsymbol{\lambda}^{(s)} \Leftrightarrow \boldsymbol{\Lambda} = \arg\min_{\mathbb{R}^{\Upsilon_d}} \|\boldsymbol{\lambda}^{(s)} - \mathbf{B}^{(s)^T} \boldsymbol{\Lambda}\|_{\mathcal{K}^{-1}} \end{aligned} \tag{16}$$

where $\mathcal{K}$ and $\mathcal{K}^{-1}$ stand for well-chosen norm of $\mathbb{R}^{\Gamma}$. Note that because of the redundancies at multiple points, $\tilde{\mathbf{B}}^{(s)}$ is not fully characterized by previous system, but anyhow the mechanically consistent piece of information $\mathbf{B}^{(s)^T} \boldsymbol{\Lambda}$ is unique (it only depends on $\mathcal{K}^{-1}$).

3.3. Building a continuous representation of balanced intereffort

In this subsection, we show how a nodal intereffort $\boldsymbol{\Lambda}_F \in \mathbb{R}^{\Upsilon_d^F}$ allows a more precise recovery of interface fluxes than (9). Indeed we can build the piecewise function $g_\Gamma^{(s)} = (g_F^{(s,s')})_{s'}$ solution to:

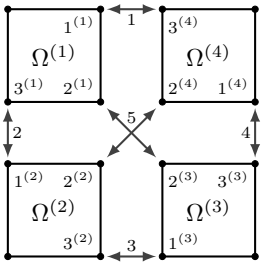
$$\left(\mathbf{B}_F^{(s)^T} \boldsymbol{\Lambda}_F \right)^T_{|\Upsilon^{(s,s')}} = \int_{\Upsilon^{(s,s')}} g_F^{(s,s')} \cdot \boldsymbol{\varphi}_{hb}^{(s)} dS \tag{17}$$

Typically one can develop $g_F^{(s,s')}$ on the finite element shape functions of $\Upsilon^{(s,s')}$ as was done in [19] on $\Gamma^{(s)}$. Then we automatically have $g_F^{(s,s')} = -g_F^{(s',s)} \in L^2(\Upsilon^{(s,s')})$, and the classical recovery can be used in parallel on the subdomains:

$$\hat{\sigma}_N = \left(\mathcal{F}_{\text{eq}} \left(\sigma_N^{(s)}, f^{(s)}, g^{(s)}, \left(g_F^{(s,s')} (\mathbf{B}_F^{(s)^T} \boldsymbol{\Lambda}_F) \right)_{s'} \right) \right)_s$$

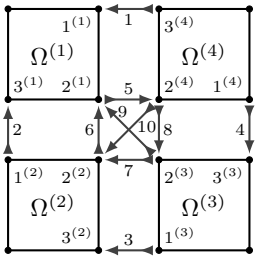
In the following subsection, we show how $\boldsymbol{\Lambda}_F \in \mathbb{R}^{\Upsilon_d^F}$ can be obtained, starting from very general data, which are easily obtained in BDD and FETI [19], but also in FETI-DP approach as explained in section 6.

DOI: 10.1002/nme



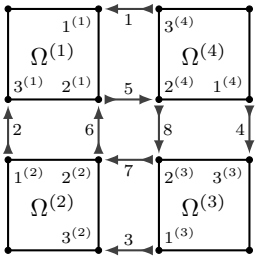
$$\mathbf{A} = \begin{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \end{pmatrix}$$

(a) Primal assembly operator $\mathbf{A}$



$$\mathbf{B} = \begin{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & -1 & 0 \end{pmatrix} \end{pmatrix}$$

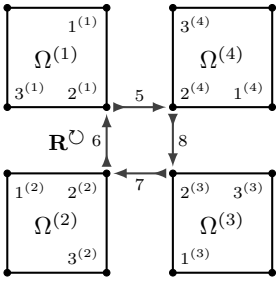
(b) Classic dual assembly operator $\mathbf{B}$



$$\mathbf{B}_{\mathrm{F}} = \begin{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & -1 & 0 \end{pmatrix} \end{pmatrix}$$

(c) Face-only dual assembly operator $\mathbf{B}_{\mathrm{F}}$

Figure 1. Examples of primal and dual assembly operators



$$\mathbf{R}^{\circlearrowleft} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

Figure 2. Illustration of $\mathbf{R}^{\circlearrowleft}$ in the framework of figure 1(c)

3.4. General methodology to compute nodal intereffort $\mathbf{\Lambda}_F$

Using nodal interefforts like $\mathbf{\Lambda}_F$ in Range($\mathbf{B}_F$) makes it simple to recover admissible stress fields. Unfortunately, it does not corresponds to the classical representation of the interface: in the primal approach, operator $\mathbf{A}$ is used; whereas in the dual approach operator $\mathbf{B}$ is used, which differs from $\mathbf{B}_F$ as soon as there exist nodes shared by more than 3 subdomains (in which case $\mathbf{B}$ features interactions between subdomains with a zero-measure interface).

DOI: 10.1002/nme

Our starting point for the construction of $\mathbf{\Lambda}_F$ is a vector of nodal tractions $\boldsymbol{\lambda}_N^{(s)}$ defined on the boundary of substructures such that:

$$\sum_s \mathbf{A}^{(s)} \boldsymbol{\lambda}_N^{(s)} = 0$$

which can always be obtained during FETI or BDD iterations [19]: in FETI, we directly have $\boldsymbol{\lambda}_N^{(s)} = \mathbf{B}^{(s)^T} \mathbf{\Lambda}$ where $\mathbf{\Lambda}$ is the main unknown of the problem; whereas in BDD, we have $\boldsymbol{\lambda}_N^{(s)} = \boldsymbol{\lambda}^{(s)} - \tilde{\mathbf{A}}^{(s)^T} \sum_j \mathbf{A}^{(j)} \boldsymbol{\lambda}^{(j)}$ where $(\boldsymbol{\lambda}^{(s)})_s$ are the nodal reactions resulting from local Dirichlet problems. The construction of $(\boldsymbol{\lambda}_N^{(s)})$ during FETI-DP iterations is discussed in section 6.

We thus wish to compute an interaction vector $\mathbf{\Lambda}_F$ which corresponds to that distribution of effort:

$$\boldsymbol{\lambda}_N^{(s)} = \mathbf{B}_F^{(s)^T} \mathbf{\Lambda}_F \tag{18}$$

Thanks to (15), this problem possesses solutions. Because of the redundancy of $\mathbf{B}_\mathrm{F}$ at multiple points, solutions are defined up to a member of the null space of $\mathbf{B}_\mathrm{F}$. The solution is made unique by imposing that $\|\mathbf{\Lambda}_F - \mathbf{\Lambda}_F^1\|_\mathcal{P}$ shall be minimal in $\mathbb{R}^{\Upsilon_d^F}$ for a chosen norm defined by a symmetric positive definite matrix $\mathbf{P}$ and a reference nodal interaction $\mathbf{\Lambda}_F^1$.

The $\mathcal{P}$-norm defines a weighted version of $(\mathbf{B}_\mathrm{F}^{(s)})$ which we write down $(\widehat{\mathbf{B}}_F^{(s)})$:

$$\mathbf{\Lambda}_F = \sum_s \widehat{\mathbf{B}}_F^{(s)} \boldsymbol{\lambda}_N^{(s)} + \left(\mathbf{I} - \sum_s \widehat{\mathbf{B}}_F^{(s)} \mathbf{B}_F^{(s)^T} \right) \mathbf{\Lambda}_F^1 \tag{19}$$

$$\text{with } \widehat{\mathbf{B}}_F^{(s)} = \mathbf{P}^{-1} \mathbf{B}_F^{(s)} \left(\sum_j \mathbf{B}_\mathrm{F}^{(j)^T} \mathbf{P}^{-1} \mathbf{B}_\mathrm{F}^{(j)} \right)^+ \tag{20}$$

$(\widehat{\mathbf{B}}_F^{(s)})$ is a sparse matrix with dense blocks for each multiple point. It is often computationally more interesting and numerically more stable to seek $\mathbf{\Lambda}_F$ as a corrected guess: let $\mathbf{\Lambda}_F^0$ be any vector satisfying (18) (typically obtained with $\mathbf{P} = \mathbf{I}$), we have:

$$\mathbf{\Lambda}_F = \mathbf{\Lambda}_F^0 - \mathbf{R}^\circlearrowleft (\mathbf{R}^{\circlearrowleft^T} \mathbf{P} \mathbf{R}^\circlearrowleft)^{-1} \mathbf{R}^{\circlearrowleft^T} \mathbf{P} \left(\mathbf{\Lambda}_F^0 - \mathbf{\Lambda}_F^1 \right) \tag{21}$$

The choices of the $\mathcal{P}$-norm and of the reference interaction $\mathbf{\Lambda}_F^1$ are crucial to obtain a good estimation of the error, in particular in the presence of heterogeneities (see Section 5 for assessments). From our experiments, we draw two conclusions: first the heterogeneity must be taken into account, second $\mathbf{\Lambda}_F^1$ must be chosen in agreement with the real mechanical state of the structure which can be estimated through the finite element Cauchy stress on the element edges $\sigma_h^{(s)} \cdot n^{(s,s')}$. It comes out that this issue has a direct equivalent in the sequential Element Equilibration Technique (EET) when optimizing the force fluxes around an internal node (see [27] for a discussion on the potential closed stress fluxes in finite element models). Note that, to the authors' knowledge, heterogeneities had never been considered in papers related to the EET.

In the next section, we present an improvement of the sequential EET in order to take into account heterogeneities. Regarding domain decomposition and multiple points, we propose to conduct the same computation. More precisely, we condense the star-patch problem built around multiple points on the interface in order to find the best $\mathbf{\Lambda}_F$. This strategy implies limited extra sparse communications: one small all-to-all exchange on local communicators associated with vertexes (in 2D) or edges (in 3D). The methodology to recover optimized stress is summed up in figure 3.

DOI: 10.1002/nme

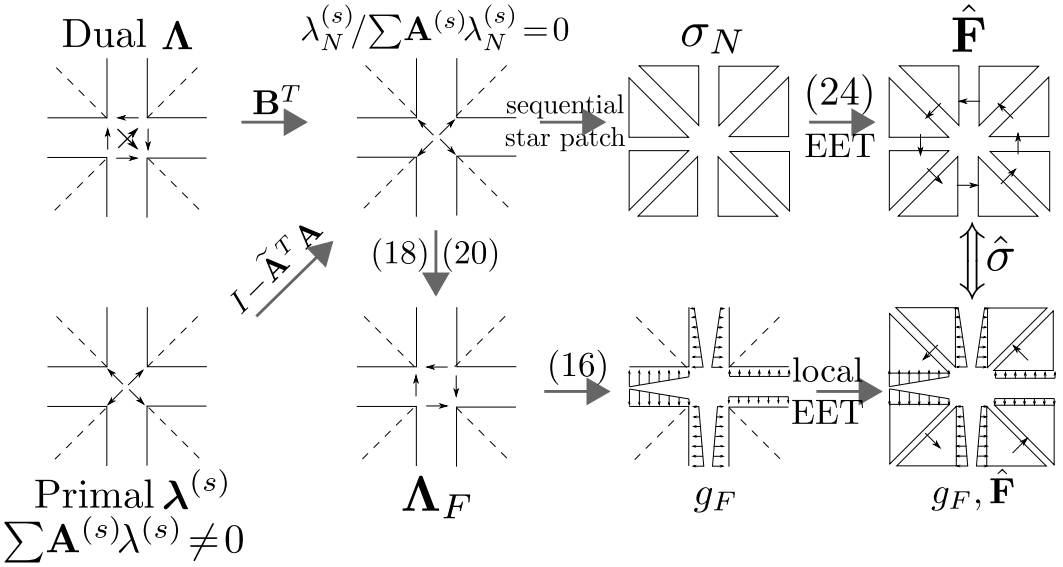


Figure 3. Methodology for parallel stress recovery / equivalent sequential star-patch

4. IMPROVEMENT OF SEQUENTIAL ESTIMATORS FOR HETEROGENEOUS PROBLEMS

This section presents how the Element Equilibration Technique (EET) can be improved to take into account heterogeneity. Moreover, as announced in previous section, these improvements have a direct counterpart in the parallel reconstruction of admissible stress field for domain decomposition methods.

We first quickly recall the main principles of equilibrated stresses recovery through the EET algorithm and invite the interested reader to refer to [8, 29] for more details. Besides, it should be noted that the improvement presented here may also be applied to newer EESPT algorithm [10], or the STARFLEET method [12], since the common basis is shared by these approaches.

4.1. Classical recovery of balanced stress

The element-wise recovery of equilibrated stress takes place in two steps:

1. A traction field $(\hat{F}_\gamma)_\gamma$ is built on the edges γ (faces in 3D) of the mesh so that it verifies the equilibrium with both external and internal loading;
2. $\hat{\sigma}_h$ is recovered from $(\hat{F}_\gamma)$ through the resolution of element-wise problems defined as:

$$\begin{aligned} \operatorname{div}(\hat{\sigma}_{h|E}) + f = 0 \qquad & \text{on } E, \\ \hat{\sigma}_{h|E}.n_E = \delta_E^\gamma \hat{F}_\gamma \qquad & \text{on } \gamma \in \partial E, \end{aligned}$$

where $\delta_E^\gamma = \pm 1$ is a signed boolean coefficient verifying $\delta_E^\gamma + \delta_{E'}^\gamma = 0$ on the edge $\gamma = \partial E \cap \partial E'$ in order to enable the balance of stress fluxes across element boundaries.

As will be shown in section 5, the jump of material coefficients need to be taken into account during the first step in order to build relevant values for equilibrated the traction fields $\hat{F}_\gamma$, leading to effective error estimators.

DOI: 10.1002/nme

The EET recovery method seeks the equilibrated traction field $(\hat{F}_\gamma)$ thanks to the following prolongation condition on each element E of the mesh:

$$\int_E (\hat{\sigma}_h - \sigma_h).\nabla\varphi_i \, dE = 0, \ \forall E \in \mathcal{E}_h, \ \forall i \in \mathcal{N}_h^E, \tag{22}$$

where $\mathcal{E}_h$ is the set of elements of Ω_h and $\mathcal{N}_h^E$ the set of vertexes of E. This condition, which links the admissible stress field $\hat{\sigma}_h$ to the finite element one σ_h, leads to the following equations for $(\hat{F}_\gamma)$:

$$\sum_{\gamma\in\partial E} \int_{\partial E} \delta_E^\gamma \hat{F}_\gamma \varphi_i \, d\Gamma = \int_E (\sigma_h : \varepsilon(\varphi_i) - f\varphi_i) \, dE, \qquad \forall E \in \mathcal{E}_h, \ \forall i \in \mathcal{N}_h^E,$$

In a practical way, the previous system enables to seek for generalized nodal values $\hat{\mathbf{F}}_{\gamma,i}^{\varphi}$ of $(\hat{F}_\gamma)$ defined on each edge γ adjacent to node i by:

$$\hat{\mathbf{F}}_{\gamma,i}^{\varphi} = \int_\gamma \delta_E^\gamma \hat{F}_\gamma \varphi_i d\Gamma.$$

We derive the resulting system in the case of one internal node as in figure 4. Note that similar optimization can be conducted in the case of nodes on the part of the boundary where a Dirichlet condition is prescribed (in other configurations the system is closed and leaves no opportunity for optimization). After expressing the condition (23) for each element containing the node i, one obtains the following problem on patch $\omega_{h,i}$ corresponding to the support of shape function φ_i (fig. 4):

$$\begin{pmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ \vdots & \vdots & \ddots & \vdots \\ -1 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \hat{\mathbf{F}}_{\gamma_1,i}^{\varphi} \\ \vdots \\ \hat{\mathbf{F}}_{\gamma_n,i}^{\varphi} \end{pmatrix} = \begin{pmatrix} \int_{E_1} (\sigma_h : \varepsilon(\varphi_i) - f\varphi_i) dE \\ \vdots \\ \int_{E_n} (\sigma_h : \varepsilon(\varphi_i) - f\varphi_i) dE \end{pmatrix} \tag{23}$$

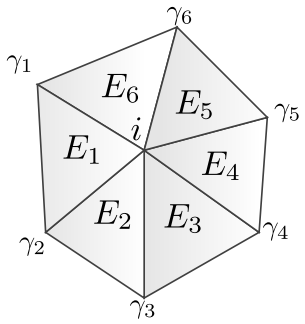


Figure 4. Star patch $\omega_{h,i}$ associated with the interior node i

The system (23) is not sufficient to find an unique set of generalized nodal values $(\hat{\mathbf{F}}_{\gamma,i}^{\varphi})_{\gamma\in\partial_i^-}$, where ∂_i^- is the set of edges radiating from the node i. Indeed the vector $\mathbf{v}^{\circlearrowleft} = \begin{pmatrix} 1 & \dots & 1 \end{pmatrix}^T$ is a basis of the null space. It is clearly the sequential counterpart of the basis of Ker $\left(\mathbf{B}_{\mathrm{F}}^T\right)$, as illustrated on figure 2. Classically, the solution is made unique by recasting the system in a constrained optimization problem:

$$\begin{aligned} &\text{minimize} \sum_{\gamma\in\partial_i^-} \left(\frac{\hat{\mathbf{F}}_{\gamma,i}^{\varphi} - \mathbf{F}_{\gamma,i}^{m}}{\mathrm{meas}(\gamma)} \right)^2 \text{ under condition (23)} \\ &\text{where } \mathbf{F}_{\gamma,i}^{m} = \int_\gamma \frac{1}{2} \left[\delta_E^\gamma \sigma_{h|E}.n_E + \delta_{E'}^\gamma \sigma_{h|E'}.n_{E'} \right] \varphi_i \, d\Gamma. \end{aligned} \tag{24}$$

DOI: 10.1002/nme

From a mechanical point of view, previous problem consists in computing the equilibrated traction field $(\hat{\mathbf{F}}_{\gamma,i})_\gamma$ nearest to the mean values of stress fluxes across elements' edges defined by $(\mathbf{F}^m_{\gamma,i})_\gamma$.

4.2. Improvement for heterogeneous problems

In the heterogeneous case, considering for instance problems with strong gaps of the Young modulus across elements, the previous choice of minimization does not seem to be relevant. Following the strategies set up when preconditioning heterogeneous problems solved by FETI or BDD algorithm, we propose to use weighted mean stress with regard to Young modulus Y in the minimization process:

$$\begin{aligned}
&\text{minimize} \sum_{\gamma\in\partial_i^-} p_\gamma \left(\frac{\hat{\mathbf{F}}^\varphi_{\gamma,i} - \mathbf{F}^m_{\gamma,i}}{\text{meas}(\gamma)}\right)^2 \text{ under condition (23)} \\
&\text{where } \tilde{\mathbf{F}}^m_{\gamma,i} = \int_\gamma \frac{1}{\mathrm{Y}_E^{-1} + \mathrm{Y}_{E'}^{-1}} \left[\mathrm{Y}_E^{-1}\delta^\gamma_E \sigma_{h|E}.n_E + \mathrm{Y}_{E'}^{-1}\delta^\gamma_{E'}\sigma_{h|E'}.n_{E'}\right]\varphi_i \, d\Gamma \\
&\text{and } p_\gamma = \left(\frac{1}{\mathrm{Y}_{E'}} + \frac{1}{\mathrm{Y}_E}\right) \text{ with } \gamma = \partial E \cap \partial E'.
\end{aligned} \tag{25}$$

Therefore, the equilibrated traction field is closer to the stress flux associated with the adjacent element with stronger flexibility.

Note that $(\hat{\mathbf{F}}^\varphi_{\gamma,i})\gamma$ can be searched as a sum of a particular solution $(\hat{\mathbf{F}}^{\varphi,P}_{\gamma,i})\gamma$ and of an element of the kernel $\beta\mathbf{v}^\circlearrowleft$ where β is a scalar computed in order to minimize the following distance :

$$\frac{1}{2}\left\| \left(\hat{\mathbf{F}}^{\varphi,P}_{\gamma,i} + \beta\mathbf{v}^\circlearrowleft - \tilde{\mathbf{F}}^m_{\gamma,i}\right)_{\gamma\in\partial_i^-} \right\|_{\mathcal{P}} \tag{26}$$

where $\mathcal{P}$ is the norm defined by $\text{diag}(p_\gamma/\sqrt{\text{meas}(\gamma)})$ which makes this minimization problem equivalent to (25). One can recognize the problem solved in the case of a multiple point (21) as detailed in the previous section.

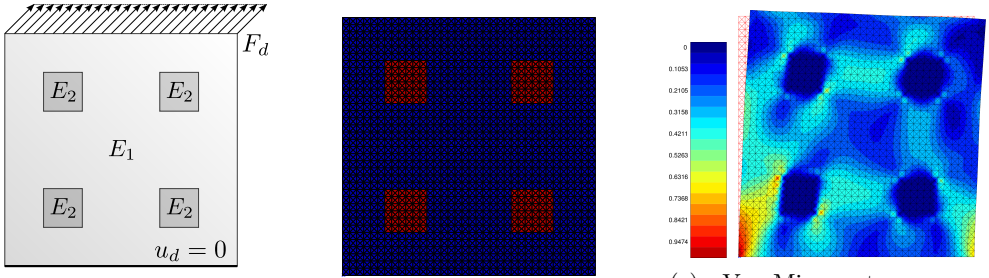
5. NUMERICAL ASSESSMENTS

In order to assess the performance of our parallel estimator in the presence of multiple points and heterogeneities, we consider the 2D problem of a square of side L clamped on its basis, submitted to traction and shear on its upper edge while both left and right edges are assumed to be traction-free. Four square are included in the matrix, as shown in figure 5. Materials are chosen to be isotropic linear elastic, with parameters $E_1 = 2.10^5$ Pa and $\nu_1 = \nu_2 = 0.3.E_2$ spans values making the heterogeneity ratio varying from 10^{-6} to 10^6.

A mesh constituted by P1 triangular elements of characteristic size $h = \frac{L}{36}$ is used. A sequential computation on the whole domain is conducted, followed by domain decomposition calculations obtained by splitting of the original domain in an increasing number $\mathcal{N}_{\text{sd}}$ of subdomains, with $\mathcal{N}_{\text{sd}} = 5, 9, 18, 36$.

Both BDD and FETI algorithms used to solve the substructured problems are respectively equipped with Neumann-Neumann and Dirichlet preconditioners with stiffness scaling operators. Beside, the convergence criterion of the solver is set to a value making the algebraic error negligible with respect to the discretization error. At convergence of the solver, we perform error estimation, as described in Section 2. The method used to build statically admissible stress field is either the EET technique [8] or the flux-free technique [14], written down SPET. Element problems are solved with p-refinement (three degrees higher polynomial basis [25]).

Prepared using nmeauth.cls
DOI: 10.1002/nme



(a) Description and loading

(b) Example of mesh generated($h = L/36$)

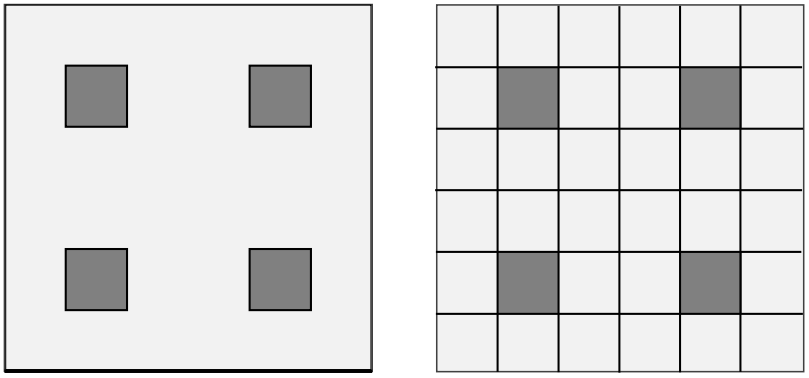
(c) Von-Mises stresses on mesh with $h = L/36$ ($E_2 = 200$Pa

Figure 5. 2D problem studied

5.1. Quality of error estimators with various partitioning

In this subsection, we consider four substructurings and two distributions of materials with inverse heterogeneity ratios (soft inclusions or stiff inclusions).

5.1.1. Domain partitioning and notation In Figure 6, we give two decompositions that involve homogeneous subdomains. The first decomposition is directly based on the position of the inclusions and is without multiple points. In Figure 7, we illustrate decompositions in 9 and 18 subdomains that lead to heterogeneities inside subdomains.



(a) Decomposition into 5 subdomains (matrix and four inclusions)

(b) Decomposition into 36 subdomains

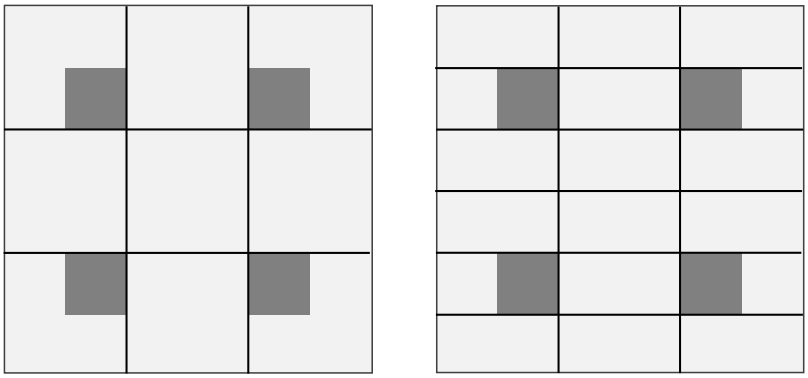
Figure 6. Substructurings with homogeneous domains

In the presentation of numerical results, subscript *optim* refers to optimization with respect to multiple points in substructured context and optimization inside the EET procedure in case of heterogeneities in the domain. For instance, since the substructuring into 9 subdomains leads to heterogeneous subdomains, EET_{optim} means that both optimizations for multiple points and for heterogeneity inside each subdomain have been done.

5.1.2. Case of soft inclusions In Table I, we present the error estimations normalized by the energy norm of the finite element solution in the case of $\frac{E_2}{E_1} = 10^{-5}$ for the four considered substructurings.

We make the following observations:

DOI: 10.1002/nme



(a) Decomposition into 9 subdomains

(b) Decomposition into 18 subdomains

Figure 7. Substructurings with heterogeneous domains

Number of subdomains	EET	EET_{optim}	$SPET$	$SPET_{optim}$
1	30.5	$3.64\ 10^{-1}$	$2.43\ 10^{-1}$	
5	$3.64\ 10^{-1}$		$2.43\ 10^{-1}$	
9	37.9	29.6	28.9	22.7
18	40.5	25.3	33.7	19.2
36	44.1	$3.99\ 10^{-1}$	38.9	$3.31\ 10^{-1}$

Table I. Dependence of the error estimators wrt the substructuring in presence of soft inclusions (10^5 ratio of Young's moduli, values normalized by the energy).

- in the sequential case, the optimization in the EET technique (presented in Section 4) enables to recover a good error estimation close to the one provided by the SPET which is always the better.
- With 5 subdomains, there is no multiple points and domains are homogeneous. Therefore no optimization was done.
- With 9 or 18 subdomains, the optimization at the multiple points enables to better the error estimation but does not lead to results as accurate as for the sequential optimization. This is due to the strong heterogeneity at the interface. As shown on the error maps in figure 8, the optimization at the multiple points reduces the parasite error in those multiples points but not along the interfaces crossing heterogeneities.

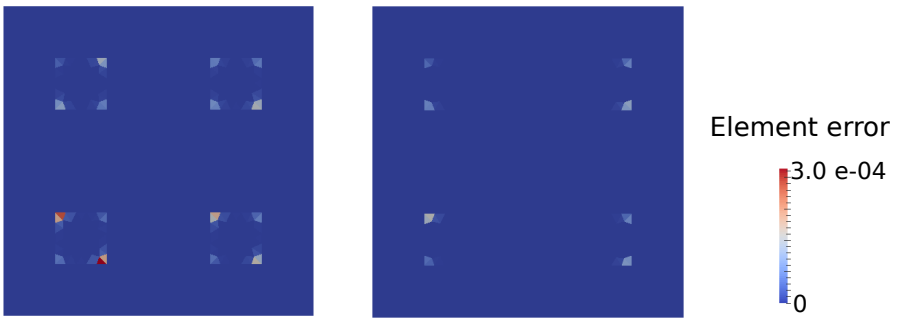


Figure 8. Error maps obtained with EET-based error estimation without optimization (left) and with optimization (right) with 18 subdomains

DOI: 10.1002/nme

5.1.3. Case of stiff inclusions In Table II, we present the relative error estimations for $\frac{E_2}{E_1} = 10^5$ for the four considered substructurings.

Number of subdomains	EET	EET_{optim}	$SPET$	$SPET_{optim}$
1	$2.36\ 10^{-1}$	$2.23\ 10^{-1}$	$1.64\ 10^{-1}$	
5	$2.34\ 10^{-1}$		$1.86\ 10^{-1}$	
9	$2.43\ 10^{-1}$	$2.35\ 10^{-1}$	$1.81\ 10^{-1}$	$1.80\ 10^{-1}$
18	$2.41\ 10^{-1}$	$2.37\ 10^{-1}$	$1.93\ 10^{-1}$	$1.92\ 10^{-1}$
36	$2.37\ 10^{-1}$	$2.36\ 10^{-1}$	$2.00\ 10^{-1}$	$1.99\ 10^{-1}$

Table II. Dependence of the error estimators wrt the substructuring in presence of stiff inclusions (10^5 ratio of Young's moduli, values normalized by the energy).

We observe that, in the sequential case, even without optimization, the error estimation provided by the EET is almost as accurate as the one provided by the SPET. We notice that the optimization in the EET does not worsen the error estimation nor improves it a lot. The error estimation in parallel resolutions is already as accurate as the error estimation in a sequential computations. As a consequence, it is not surprising that the specific treatment of multiple point does not lead to better results.

5.2. Quality of error estimators with increasing heterogeneity between subdomains

In this subsection, we decompose the structure into 36 identical square subdomains (see Figure 6(b)) and we keep the same substructuring and discretization. Note that with this decomposition, every subdomain is homogeneous. We study the behavior of the error estimators with increasing heterogeneity. In the presentation of the results, we adopt the following notations :

- EET : sequential resolution and use of the EET procedure for the construction of admissible stress fields
- SPET : sequential resolution and use of the SPET procedure for the construction of admissible stress fields
- EET optim : sequential resolution and use of the EET procedure with optimization for the construction of admissible stress fields
- DD EET : parallel resolution and use of the EET procedure
- DD SPET : parallel resolution and use of the SPET procedure
- DD optim EET : parallel resolution with optimization on the multiples points and use of the EET procedure
- DD optim SPET : parallel resolution with optimization on the multiples points and use of the SPET procedure

In Figure 9, E_2 is smaller than E_1 and in Figure 10, E_2 is larger than E_1.

We observe that in the cases where basic estimators behave poorly (EET, DD EET and DD SPET with soft inclusions), the optimization enables to recover correct order of magnitude whatever the heterogeneity ratio. In other cases all estimators give quite close results.

6. APPLICATION TO ERROR ESTIMATION IN FETI-DP ALGORITHM

The results of previous sections are very useful in order to derive a procedure to obtain admissible fields in the FETI-DP method [23, 24] and to prove that the bounds with separated contributions of [12, 30, 31] also apply to this algorithm.

Note that all results presented here are also valid for BDDC [26] which corresponds to the derivation of the same ideas as FETI-DP starting from the primal approach.

Prepared using nmeauth.cls
DOI: 10.1002/nme

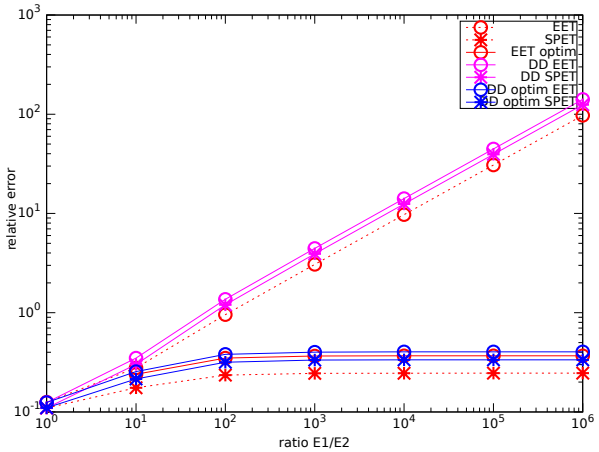


Figure 9. Quality of error estimators with increasing heterogeneity : soft inclusions

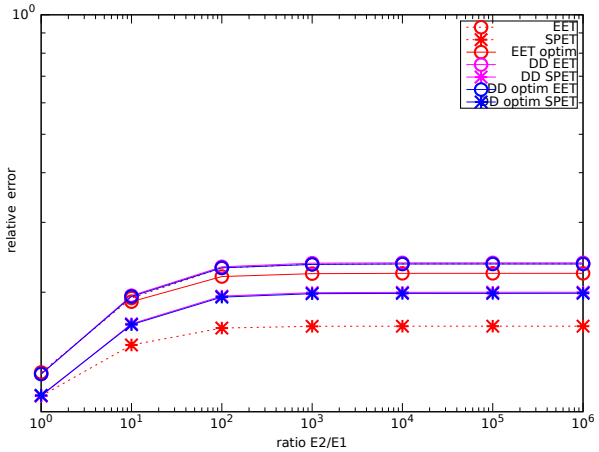


Figure 10. Quality of error estimators with increasing heterogeneity : stiff inclusions

6.1. The FETI-DP algorithm

In the FETI-DP algorithm, primal constraints are incorporated in the formulation so that subdomains always remain weakly connected. The chosen connections make all local problems well posed, and act like a coarse problem which warranties the scalability of the method even in difficult situations: [26, 32, 33]. The most basic constraint consists, in 2D, in ensuring the continuity of the displacement at corners (which are the boundary of edges, multiple points being a particular class of such nodes). This is the type of constraints we will focus on subsequently.

Note that more general constraints can be chosen, like weighted averages over faces (in 3D). Often these constraints are implemented using local change of basis (so that even complicated constraints end up to be applied on single modified degrees of freedom). Anyhow these changes of basis do not modify the methodology to compute admissible fields because continuity and balance are properties independent from the chosen basis.

Let the subscript c represent corners, and the subscript r represent the remaining degrees of freedom of which we distinguish the internal dofs i from the other interface dofs o (not corners). We define in a straightforward manner the restriction of trace and assembly operators on corners and other degrees of freedom. The FETI-DP problem can be written

DOI: 10.1002/nme

as:

$$\text{Find } \boldsymbol{\Lambda}_o \text{ such that } \begin{cases} \begin{cases} \mathbf{K}^{(s)}\mathbf{u}^{(s)} = \mathbf{f}^{(s)} + \mathbf{t}_o^{(s)^T}\mathbf{B}_o^{(s)^T}\boldsymbol{\Lambda}_o, \ \forall s \\ \text{under the constaint } \sum_s \mathbf{B}_c^{(s)}\mathbf{t}_c^{(s)}\mathbf{u}^{(s)} = 0 \end{cases} \\ \text{such that } \sum_s \mathbf{B}_o^{(s)}\mathbf{t}_o^{(s)}\mathbf{u}^{(s)} = 0 \end{cases} \tag{27}$$

Algorithm 1 presents the associated algorithm, with the following notations:

- $\mathbf{K}_{cc}^*$ matrix of the coarse problem:

$$\mathbf{K}_{cc}^* = \sum_s \mathbf{A}_c^{(s)} \left(\mathbf{K}_{cc}^{(s)} - \mathbf{K}_{rc}^{(s)^T} \mathbf{K}_{rr}^{(s)^{-1}} \mathbf{K}_{rc}^{(s)} \right) \mathbf{A}_c^{(s)^T} \tag{28}$$

- Scaled assembly matrix $\widetilde{\mathbf{B}}_o^{(s)}$, such that $\sum \mathbf{B}_o^{(s)} \widetilde{\mathbf{B}}_o^{(s)^T} = \mathbf{I}$
- Forward problem $(\mathbf{u}^{(s)}, \boldsymbol{\lambda}_c^{(s)}) = \text{Solve}_L(\boldsymbol{\lambda}_o^{(s)}, \mathbf{f}^{(s)})$:

$$\begin{cases} \mathbf{u}^{(s)} = \begin{pmatrix} \mathbf{u}_{1,r}^{(s)} + \mathbf{u}_{3,r}^{(s)} \\ \mathbf{u}_c^{(s)} \end{pmatrix}, \quad \boldsymbol{\lambda}_c^{(s)} = \mathbf{K}_{cr}^{(s)}(\mathbf{u}_{1,r}^{(s)} + \mathbf{u}_{3,r}^{(s)}) + \mathbf{K}_{cc}^{(s)}\mathbf{u}_c^{(s)} \\ \text{with } \mathbf{u}_{1,r}^{(s)} \text{ solution to } \mathbf{K}_{rr}^{(s)}\mathbf{u}_{1,r}^{(s)} = \mathbf{t}^{(s)^T}\boldsymbol{\lambda}_o^{(s)} + \mathbf{f}^{(s)} \\ \text{and } \mathbf{u}_c^{(s)} = -\mathbf{A}_c^{(s)^T}\mathbf{K}_{cc}^{*\,-1} \sum_j \mathbf{A}_c^{(j)}\mathbf{K}_{cr}^{(j)}\mathbf{u}_{1,r}^{(j)} \\ \text{and } \mathbf{u}_{3,r}^{(s)} \text{ solution to } \mathbf{K}_{rr}^{(s)}\mathbf{u}_{3,r}^{(s)} = -\mathbf{K}_{rc}^{(s)}\mathbf{u}_c^{(s)} \end{cases} \tag{29}$$

- Preconditioning $(\boldsymbol{\delta\lambda}^{(s)}, \delta\mathbf{u}^{(s)}) = \text{Solve}_S(\mathbf{u}_o^{(s)})$

$$\begin{cases} \mathbf{K}^{(s)}\delta\mathbf{u}^{(s)} = \mathbf{t}^{(s)^T}\boldsymbol{\delta\lambda}^{(s)} \\ \text{under the conditions } \mathbf{t}_o^{(s)}\delta\mathbf{u}^{(s)} = \mathbf{u}_o^{(s)}, \ \mathbf{t}_c^{(s)}\delta\mathbf{u}^{(s)} = 0 \end{cases}$$

6.2. Parallel reconstruction of admissible fields and error estimation

In algorithm 1, we showed how it was possible at no cost to build the fields $\mathbf{u}_N^{(s)}, \mathbf{u}_D^{(s)}, \boldsymbol{\lambda}_N^{(s)}$. As proved in the following paragraphs, the following properties hold:

- $\mathbf{u}_D^{(s)}$ defines a continuous field of displacement:

$$\sum_s \mathbf{B}_o^{(s)}\mathbf{t}_o^{(s)}\mathbf{u}_D^{(s)} = \sum_s \mathbf{B}_o^{(s)}\mathbf{t}_o^{(s)}\mathbf{u}_N^{(s)} - \sum_s \mathbf{B}_o^{(s)}\mathbf{t}_o^{(s)}\widetilde{\mathbf{B}}_o^{(s)^T}\mathbf{B}_o^{(s)}\mathbf{t}_o^{(s)}\mathbf{u}_N^{(s)} = 0 \tag{30}$$

  The continuity at corners is at the basis of the coarse problem and is not perturbed during the preconditioning (where null displacement is imposed on corners)
- $(\mathbf{u}_N^{(s)}, \boldsymbol{\lambda}_N^{(s)})$ satisfy the subdomains' equilibrium:

$$\mathbf{K}^{(s)}\mathbf{u}_N^{(s)} = \mathbf{f}^{(s)} + \mathbf{t}^{(s)^T}\boldsymbol{\lambda}_N^{(s)} \tag{31}$$

  and reactions are balanced at the interface:

$$\begin{aligned} &\sum_s \mathbf{A}_o^{(s)}\boldsymbol{\lambda}_{N,o}^{(s)} = \sum_s \mathbf{A}_o^{(s)}\mathbf{B}_o^{(s)}\boldsymbol{\Lambda}_o = 0 \\ &\sum_s \mathbf{A}_c^{(s)}\boldsymbol{\lambda}_{N,c}^{(s)} = 0 \text{ from definition (29)} \end{aligned} \tag{32}$$

DOI: 10.1002/nme

Algorithm 1: FETI-DP: main unknown $\mathbf{\Lambda}_o$

Initialization $\mathbf{\Lambda}_o$;
$(\mathbf{u}_N^{(s)}, \boldsymbol{\lambda}_c^{(s)}) = \text{Solve}_L(\mathbf{B}_o^{(s)^T}\mathbf{\Lambda}_o, \mathbf{f}^{(s)})$; // $\boldsymbol{\lambda}_N^{(s)} = \left[\mathbf{B}_o^{(s)^T}\mathbf{\Lambda}_o; \boldsymbol{\lambda}_c^{(s)}\right]$
Compute residual $\mathbf{r} = (\sum_s \mathbf{B}_o^{(s)}\mathbf{t}_o^{(s)}\mathbf{u}_N^{(s)})$;
Define local displacement $\delta\mathbf{u}_o^{(s)} = \widetilde{\mathbf{B}}_o^{(s)^T}\mathbf{r}$;
$(\boldsymbol{\delta\lambda}^{(s)}, \delta\mathbf{u}^{(s)}) = \text{Solve}_S(\delta\mathbf{u}_o^{(s)})$; // $\mathbf{u}_D^{(s)} = \mathbf{u}_N^{(s)} - \delta\mathbf{u}^{(s)}$, $\boldsymbol{\lambda}_D^{(s)} = \boldsymbol{\lambda}_N^{(s)} - \boldsymbol{\delta\lambda}^{(s)}$
Compute preconditioned residual $\mathbf{z} = \sum_s \widetilde{\mathbf{B}}_o^{(s)}\boldsymbol{\delta\lambda}_o^{(s)}$;
Compute search direction $\mathbf{w} = \mathbf{z}$;
while $\sqrt{\mathbf{r}^T\mathbf{z}} > \epsilon$ **do**
  $(\delta\mathbf{u}_N^{(s)}, \Delta\boldsymbol{\lambda}_c^{(s)}) = \text{Solve}_L(\mathbf{B}_o^{(s)^T}\mathbf{w}, 0)$;
  $\mathbf{q} = \sum_s \mathbf{B}_o^{(s)}\mathbf{t}_o^{(s)}\delta\mathbf{u}_N^{(s)}$;
  $\alpha = (\mathbf{r}^T\mathbf{z})/(\mathbf{q}^T\mathbf{w})$;
  $\mathbf{\Lambda}_o \leftarrow \mathbf{\Lambda}_o + \alpha\mathbf{w}$; // $\mathbf{u}_N^{(s)} \leftarrow \mathbf{u}_N^{(s)} + \alpha\delta\mathbf{u}_N^{(s)}$, $\boldsymbol{\lambda}_N^{(s)} = \left[\mathbf{B}_o^{(s)^T}\mathbf{\Lambda}_o; \boldsymbol{\lambda}_c^{(s)} + \alpha\Delta\boldsymbol{\lambda}_c^{(s)}\right]$
  $\mathbf{r} \leftarrow \mathbf{r} - \alpha\mathbf{q}$;
  $\delta\mathbf{u}_o^{(s)} = \widetilde{\mathbf{B}}_o^{(s)^T}\mathbf{r}$;
  $(\boldsymbol{\delta\lambda}^{(s)}, \delta\mathbf{u}^{(s)}) = \text{Solve}_S(\delta\mathbf{u}_o^{(s)})$; // $\mathbf{u}_D^{(s)} = \mathbf{u}_N^{(s)} - \delta\mathbf{u}^{(s)}$, $\boldsymbol{\lambda}_D^{(s)} = \boldsymbol{\lambda}_N^{(s)} - \boldsymbol{\delta\lambda}^{(s)}$
  $\mathbf{z} = \sum_s \widetilde{\mathbf{B}}_o^{(s)}\boldsymbol{\delta\lambda}_o^{(s)}$;
  $\mathbf{w} \leftarrow \mathbf{z} - (\mathbf{q}^T\mathbf{z})/(\mathbf{q}^T\mathbf{w})\mathbf{w}$
end

This means that $u_D = (\boldsymbol{\varphi}_h^{(s)}\mathbf{u}_D^{(s)})$ is a kinematically admissible displacement field, and that $\sigma_N^{(s)} = \mathbb{H} : \varepsilon(\boldsymbol{\varphi}_h^{(s)}\mathbf{u}_N^{(s)})$ can be used, together with $\boldsymbol{\lambda}_N^{(s)}$, to compute a globally statically admissible stress field $\hat{\sigma}_h$.

Once those fields are obtained, the error estimation consists in the evaluation of the quantity

$$\boldsymbol{e}_{\text{CR},\Omega}(\hat{u}_h, \hat{\sigma}_h) = \sqrt{\sum_{s=1}^{\mathcal{N}_{\text{sd}}} \left(\boldsymbol{e}_{\text{CR},\Omega^{(s)}}(\hat{u}_h^{(s)}, \hat{\sigma}_h^{(s)})\right)^2}.$$

which is a strict upper bound of the true error.

6.3. Separation of the sources of the error

We have the following property, using notations of algorithm 1:

$$\begin{aligned}
|||u_N - u_D|||_\Omega^2 &= \sum_s \int_{\Omega^{(s)}} \varepsilon(\underline{u}_D^{(s)} - \underline{u}_N^{(s)}) : \mathbb{H} : \varepsilon(\underline{u}_D^{(s)} - \underline{u}_N^{(s)}) d\Omega \\
&= \sum_s \left(\boldsymbol{\lambda}_D^{(s)} - \boldsymbol{\lambda}_N^{(s)}\right)^T \mathbf{t}^{(s)} \left(\mathbf{u}_D^{(s)} - \mathbf{u}_N^{(s)}\right) \\
&= \sum_s \boldsymbol{\delta\lambda}^{(s)^T}\mathbf{t}^{(s)}\delta\mathbf{u}^{(s)} = \sum_s \boldsymbol{\delta\lambda}_o^{(s)^T}\delta\mathbf{u}_o^{(s)} \\
&= \mathbf{r}^T\mathbf{z}
\end{aligned} \tag{33}$$

DOI: 10.1002/nme

where we have used the fact that $\delta \mathbf{u}_c^{(s)} = 0$

This means that the quantity $|||u_N - u_D|||_\Omega^2$ is naturally computed at each iteration. Thus all the results involving separation of sources of error (11) can be used in FETI-DP (see also [30, 31] for bounds on quantities of interest and lower bounds).

7. CONCLUSION

In this article, we emphasize the role of multiple points for the error estimation in the framework of domain decomposition methods. We propose an optimization to reconstruct nodal reactions at these specific points that is optional in absence of heterogeneity but is necessary to recover a efficient error estimator in case of strong heterogeneities. By the way, because of strong similarities between optimization at multiple points and reconstruction of fluxes in star-patches in the EET algorithm, we propose an enhancement of the EET in case of strong heterogeneities. The numerical assessments on a 2D mechanical structure show the quality of the proposed improvements. Finally, the specific attention towards the multiple points also allows us to extend the parallel error estimation procedure to FETI-DP (and BDDC) algorithms.

ACKNOWLEDGEMENT

The authors wish to thank professor Nicolas Moës for pointing out the particular role of multiple points in the recovery process.

REFERENCES

1. Brenner SC, Ridgway Scott L. *The mathematical theory of finite element methods, third edition*, *Texts in applied mathematics*, vol. 15. Springer, 2008.
2. Zienkiewicz OC, Zhu JZ. A simple error estimator and adaptive procedure for practical engineerng analysis. *International Journal for Numerical Methods in Engineering* 1987; **24**(2):337–357, doi: 10.1002/nme.1620240206.
3. Ladevèze P. Comparaison de modèles de milieux continus. PhD Thesis, Université P. et M. Curie 1975.
4. Ladevèze P, Pelle JP. *Mastering Calculations in Linear and Nonlinear Mechanics*. Springer: New York, 2004.
5. Kelly DW. The self-equilibration of residuals and complementary a posteriori error estimates in the finite element method. *International Journal for Numerical Methods in Engineering* 1984; **20**(8):1491–1506.
6. Kelly DW, Isles JD. Procedures for residual equilibration and local error estimation in the finite element method. *Communications in Applied Numerical Methods* 1989; **5**(8):497–505, doi: 10.1002/cnm.1630050803.
7. Kempeneers M, Debongnie JF, Beckers P. Pure equilibrium tetrahedral finite elements for global error estimation by dual analysis. *International Journal for Numerical Methods in Engineering* 2009; **81**(4):513–536, doi:10.1002/nme.2703.
8. Ladevèze P, Leguillon D. Error estimate procedure in the finite element method and application. *SIAM Journal of Numerical Analysis* 1983; **20**(3):485–509.
9. Ladevèze P, Rougeot P. New advances on a posteriori error on constitutive relation in finite element analysis. *Computer Methods in Applied Mechanics and Engineering* 1997; **150**(1-4):239–249, doi: 10.1016/S0045-7825(97)00089-3.
10. Ladevèze P, Chamoin L, Florentin E. A new non-intrusive technique for the construction of admissible stress fields in model verification. *Computer Methods in Applied Mechanics and Engineering* 2010; **199**:766–777, doi:10.1016/j.cma.2009.11.007.
11. Pled F, Chamoin L, Ladevèze P. On the techniques for constructing admissible stress fields in model verification: Performances on engineering examples. *International Journal for Numerical Methods in Engineering* 2011; **88**(5):409–441, doi:10.1002/nme.3180.
12. Rey V, Gosselet P, Rey C. Study of the strong prolongation equation for the construction of statically admissible stress fields: implementation and optimization. *Computer Methods in Applied Mechanics and Engineering* 2014; **268**(1):82–104, doi:10.1016/j.cma.2013.08.021.
13. Carstensen C, Funken SA. Fully reliable localized error control in the fem. *SIAM J. Sci. Comput.* 2000; **21**(4):1465–1484, doi:10.1137/S1064827597327486.

14. Parés N, Santos H, Díez P. Guaranteed energy error bounds for the poisson equation using a flux-free approach: Solving the local problems in subdomains. *International Journal for Numerical Methods in Engineering* 2009; **79**(10):1203–1244, doi:10.1002/nme.2593.
15. Parés N, Díez P, Huerta A. Exact bounds for linear outputs of the advection-diffusion-reaction equation using flux-free error estimates. *SIAM J. Sci. Comput.* 2009; **31**(4):3064–3089, doi:10.1137/080724356.
16. Gallimard L. A constitutive relation error estimator based on traction-free recovery of the equilibrated stress. *International Journal for Numerical Methods in Engineering* 2009; **78**(4):460–482, doi: 10.1002/nme.2496.
17. Maunder EAW, Moitinho de Almeida JP. The stability of stars of triangular equilibrium plate elements. *International Journal for Numerical Methods in Engineering* 2009; **77**(7):922–968, doi: 10.1002/nme.2441.
18. Moitinho de Almeida JP, Maunder EAW. Recovery of equilibrium on star patches using a partition of unity technique. *International Journal for Numerical Methods in Engineering* 2009; **79**(12):1493–1516, doi:10.1002/nme.2623.
19. Parret-Fréaud A, Rey C, Gosselet P, Feyel F. Fast estimation of discretization error for fe problems solved by domain decomposition. *Computer Methods in Applied Mechanics and Engineering* 2010; **199**(49-52):3315–3323, doi:10.1016/j.cma.2010.07.002.
20. Gosselet P, Rey C. Non-overlapping domain decomposition methods in structural mechanics. *Archives of Computational Methods in Engineering* 2006; **13**(4):515–572, doi:10.1007/BF02905857.
21. Mandel J. Balancing domain decomposition. *Communications in Numerical Methods in Engineering* 1993; **9**(3):233, doi:10.1002/cnm.1640090307.
22. Farhat C, Roux FX. Implicit parallel processing in structural mechanics. *Computational Mechanics Advances* 1994; **2**(1):1–124. North-Holland.
23. Farhat C, Lesoinne M, Pierson K. A scalable dual-primal domain decomposition method. *Numer. Linear Algebra Appl.* 2000; **7**(7-8):687–714.
24. Farhat C, Lesoinne M, LeTallec P, Pierson K, Rixen D. FETI-DP: a dual-primal unified FETI method - part i: a faster alternative to the two-level FETI method. *Int. J. Num. Meth . Eng.* 2001; **50**(7):1523–1544.
25. Babuška I, Strouboulis T, Upadhyay CS, Gangaraj SK, Copps K. Validation of a posteriori error estimators by numerical approach. *International journal for numerical methods in engineering* 1994; **37**(7):1073–1123, doi:10.1002/nme.1620370702.
26. Dohrmann CR. A preconditionner for substructuring based on constrained energy minimization. *SIAM Journal for Scientific Computing* 2003; **25**:246.
27. Rey V, Rey C, Gosselet P. A strict error bound with separated contributions of the discretization and of the iterative solver in non-overlapping domain decomposition methods. *Computer Methods in Applied Mechanics and Engineering* 2013; **270**(1):293–303.
28. Justino MR, Park KC, Felippa CA. An algebraically partitioned feti method for parallel structural analysis: performance evaluation. *International Journal for Numerical Methods in Engineering* 1997; **40**(15):2739–2758, doi:10.1002/(SICI)1097-0207(19970815)40:15<2739::AID-NME186>3.0.CO;2-0.
29. Ladevèze P, Pelle JP, Rougeot P. Error estimation and mesh optimization for classical finite elements. *Engineering Computations* 1993; **8**(1):69–80.
30. Rey V, Gosselet P, Rey C. Strict bounding of quantities of interest in computations based on domain decomposition. *Computer Methods in Applied Mechanics and Engineering* 2015; **287**(0):212 – 228, doi: http://dx.doi.org/10.1016/j.cma.2015.01.009.
31. Rey V, Gosselet P, Rey C. Strict lower bounds with separation of sources of error in non-overlapping domain decomposition methods. *International Journal for Numerical Methods in Engineering* 2016; :n/a–n/adoi:10.1002/nme.5244. Nme.5244.
32. A Klawonn A, O Rheinbach O. Robust feti-dp methods for heterogeneous three dimensional elasticity problems. *Computer Methods in Applied Mechanics and Engineering* 2007; **196**:1400–1414.
33. Klawonn A, Rheinbach O, Widlund OB. An analysis of a feti-dp algorithm on irregular subdomains in the plane. *SIAM J. Numer. Anal.* Jun 2008; **46**(5):2484–2504, doi:10.1137/070688675.

G Article [Rey *et al.*, 2016], bornes sur l'erreur dans les calculs par décomposition de domaine

Cet article présente les derniers résultats obtenus sur l'obtention de bornes supérieures et inférieures sur l'erreur (globale et sur des quantités d'intérêt) où la contribution de la discrétisation et celle de la décomposition de domaine apparaissent séparément. Ces bornes permettent d'arrêter les solveurs dès que l'erreur de discrétisation devient majoritaire. Une première séquence de calculs, avec adaptation de maillage et recyclage de l'information, est proposée.

Sur le même thème, le lecteur pourra consulter [Rey *et al.*, 2014a] où la première borne supérieure à contributions séparées est présentée et [Rey *et al.*, 2015] qui traite des quantités d'intérêt.

Strict lower bounds with separation of sources of error in non-overlapping domain decomposition methods

V. Rey[1], P. Gosselet[1], C. Rey[2]
[1] LMT-Cachan / Ecole Normale Supérieure de Cachan, CNRS, Université Paris Saclay
61, avenue du président Wilson, 94235 Cachan, France,
[2] Safran Tech, rue des Jeunes Bois Châteaufort CS 80112,78772 Magny les Hameaux, France

June 16, 2016

Abstract

This article deals with the computation of guaranteed lower bounds of the error in the framework of finite element (FE) and domain decomposition (DD) methods. In addition to a fully parallel computation, the proposed lower bounds separate the algebraic error (due to the use of a DD iterative solver) from the discretization error (due to the FE), which enables the steering of the iterative solver by the discretization error. These lower bounds are also used to improve the goal-oriented error estimation in a substructured context. Assessments on 2D static linear mechanic problems illustrate the relevance of the separation of sources of error and the lower bounds' independence from the substructuring. We also steer the iterative solver by an objective of precision on a quantity of interest. This strategy consists in a sequence of solvings and takes advantage of adaptive remeshing and recycling of search directions.
Keywords: Verification; Error estimation; Finite element method; Domain decomposition methods; FETI; BDD

1 Introduction

Virtual testing is a useful tool for engineers to certify structures without resorting to experimental tests. However, its massive adaption comes with several challenges. Among others, virtual testing requires the capability to solve large problems (several millions degrees of freedom) and to warrant the quality of the results provided by simulations. To tackle these difficulties, we propose to use domain decomposition methods and verification. On the one hand, non-overlapping domain decomposition methods [11, 16, 7] are well-known techniques that enable the solving of large mechanical problems by exploiting parallel computers' performance. On the other hand, verification provides tools to estimate the distance between the unknown exact solution and the computed approximated solution. This distance is the approximation error, it can be estimated by a global energy norm or by local quantities of interest (goal-oriented error estimation).

This paper is the continuation of papers connecting domain decomposition methods and verification. In [22], the authors proposed a parallel error estimator based on the error in constitutive relation [14] in a substructured framework. They described a methodology to construct the required admissible fields for error estimation. Those fields were rebuilt using quantities naturally processed during the solving and preconditioning steps of classical domain decomposition algorithms as inputs for classical equilibration techniques used in parallel on each subdomain. In [28], a new parallel error estimator that separates the discretization error (due to the finite element method) from the algebraic error (due to the iterative solver) was proposed. This new estimator enables the definition of a new stopping criterion for the iterative solver, no longer defined regardless the discretization, which avoids over-solving. Finally, in [27], this work was extended to goal-oriented error estimation. The exact value of a linear quantity of interest defined by an extractor [19, 32, 18] was estimated using global error estimation of a forward problem and an adjoint problem. It was shown that these two problems could be solved simultaneously thanks to a block-Krylov algorithm [30] steered by an objective on the error on the quantity of interest.

Upper bounds are the main concern of verification and the literature on lower bounds is scarce. By exploiting the residual equation [24] and constructing a continuous error estimation, a lower bound of the error can be computed. The question of the construction of a continuous error estimation has already been adressed in many papers (for instance [20, 4, 9]) where the authors benefit the computation of an upper bound for the computation of a lower bound.

The objective of this paper is to investigate the computation of a lower bound of the error in a substructured framework. In line with the papers associating domain decomposition methods and verification, the lower bounds are computed in parallel and they separate the two sources of error (discretization error and algebraic error). Based on the results demonstrated in [20], the computation of the lower bound does not involve significant cost since it exploits the fields computed during the reconstruction of a statically admissible stress fields.

The paper is organized as follows. In section 2, we define the reference problem and recall the principle of the error in constitutive relation. We also recall the domain decomposition methods' principles and highlight the fields built at each iteration. Finally, we recall the parallel error estimators developed in [21, 28] and give a brief state of the art of the computation of continuous fields for lower bounds of the error. In section 3, two theorems providing lower bounds with and without separation of sources of error are demonstrated. The parallel reconstruction of fields required to compute these bounds is detailed. We also show how to benefit from this global information on the error to better the goal-oriented error estimation. In section 4, we apply these lower bounds on two-dimensional mechanical structures. We compare the lower bounds provided by a sequential approach and primal and dual approaches and also study the independence with respect to the substructuring. We illustrate the convergence of the lower bounds during the iterations and illustrate the separation of sources. Finally, in order to reach an objective of precision on a quantity of interest, we apply an auto-adaptive strategy on one of the structures. In this strategy, we use the separation of sources of error to define the stopping criterion for the iterative solver. Benefiting the informations from a first solving, we process adaptive remeshing to better the FE solution and lower the error bounds and we recycle the search directions generated (Krylov subspace recycling, see [25, 10, 29]) to speed up further solvings. Section 5 concludes the paper.

2 Settings

2.1 Reference problem

Let $\mathbb{R}^d$ represents the physical space. Let us consider the static equilibrium of a (polyhedral) structure which occupies the open domain $\Omega \subset \mathbb{R}^d$ and which is subjected to given body force $\underline{f} \in \mathtt{L}^2(\Omega)$ within Ω, to given traction force $\underline{g} \in \mathtt{L}^2(\partial_g\Omega)$ on $\partial_g\Omega$ and to given displacement field $\underline{u}_d$ on the complementary part of the boundary (such that $\text{meas}(\partial_u\Omega) \neq 0$). We assume that the structure undergoes small perturbations and that the material is linear elastic, characterized by Hooke's elasticity tensor $\mathbb{H}$. Let $\underline{u}$ be the unknown displacement field, $\underline{\underline{\varepsilon}}(\underline{u})$ the symmetric part of the gradient of $\underline{u}$, $\underline{\underline{\sigma}}$ the Cauchy stress tensor. Let ω be an open subset of Ω.

We introduce two affine subspaces and one positive form:

- Affine subspace of kinematic admissible fields (KA-fields)

$$\text{KA}(\omega) = \left\{ \underline{u} \in \left(\mathtt{H}^1(\omega)\right)^d,\ \underline{u} = \underline{u}_d \text{ on } \partial\omega \bigcap \partial_u\Omega \right\} \tag{1}$$

  and we note $\text{KA}^0(\omega)$ the following linear subspace:

$$\text{KA}^0(\omega) = \left\{ \underline{u} \in \left(\mathtt{H}^1(\omega)\right)^d,\ \underline{u} = 0 \text{ on } \partial\omega \bigcap \partial_u\Omega \right\} \tag{2}$$

  and $\text{KA}^{00}(\omega)$ the following linear subspace:

$$\text{KA}^{00}(\omega) = \left\{ \underline{u} \in \left(\mathtt{H}^1(\omega)\right)^d,\ \underline{u} = 0 \text{ on } \partial\omega \backslash \partial_g\Omega \right\} \tag{3}$$

  Remark. Note that if $\omega = \Omega$, $\text{KA}^{00}(\omega)$ and $\text{KA}^0(\omega)$ are identical.

- Affine subspace of statically admissible fields (SA-fields)

$$\text{SA}(\omega) = \left\{ \underline{\underline{\tau}} \in \left(\mathtt{L}^2(\omega)\right)^{d\times d}_{\text{sym}};\ \forall \underline{v} \in \text{KA}^{00}(\omega), \int_\omega \underline{\underline{\tau}} : \underline{\underline{\varepsilon}}(\underline{v})\, d\Omega = \int_\omega \underline{f} \cdot \underline{v} d\Omega + \int_{\partial\omega \bigcap \partial_g\Omega} \underline{g} \cdot \underline{v} dS \right\} \tag{4}$$

- Error in constitutive relation [14]

$$e_{CR_\omega}(\underline{u}, \underline{\underline{\sigma}}) = \| \underline{\underline{\sigma}} - \mathbb{H} : \underline{\underline{\varepsilon}}(\underline{u}) \|_{\mathbb{H}^{-1},\omega} \tag{5}$$

  where $\|\underline{\underline{x}}\|_{\mathbb{H}^{-1},\omega} = \sqrt{\int_\omega \left(\underline{\underline{x}} : \mathbb{H}^{-1} : \underline{\underline{x}}\right) d\Omega}$

The mechanical problem set on Ω can be formulated as:

$$\text{Find } \left(\underline{u}_{ex}, \underline{\underline{\sigma}}_{ex}\right) \in \text{KA}(\Omega) \times \text{SA}(\Omega) \text{ such that } e_{CR_\Omega}(\underline{u}_{ex}, \underline{\underline{\sigma}}_{ex}) = 0 \tag{6}$$

The solution to this problem, named "exact" solution, exists and is unique.

Remark. The formulation (6) is equivalent to the classical following formulation:

$$\text{Find } \underline{u} \in \mathrm{KA}(\Omega) \text{ such that } \forall \underline{v} \in \mathrm{KA}^{00}(\Omega),\ a(\underline{u},\underline{v}) = L(\underline{v}) \tag{7}$$

with

$$a(\underline{u},\underline{v}) = \int_\Omega \underline{\underline{\varepsilon}}(\underline{u}) : \mathbb{H} : \underline{\underline{\varepsilon}}(\underline{v})\, d\Omega \tag{8}$$

and

$$L(\underline{v}) = \int_\Omega \underline{f} \cdot \underline{v} d\Omega + \int_{\partial_g\Omega \bigcap \partial\Omega} \underline{g} \cdot \underline{v} dS \tag{9}$$

2.1.1 Finite element approximation

Let us consider a mesh of Ω to which we associate the finite-dimensional subspace $\mathrm{KA}_H(\Omega)$ of $\mathrm{KA}(\Omega)$. $\mathcal{T}$ is the set of elements of the mesh and $\mathcal{V}$ is the set of vertexes. The classical finite element displacement approximation consists in searching:

$$\begin{aligned} &\underline{u}_H \in \mathrm{KA}_H(\Omega) \\ &\underline{\underline{\sigma}}_H = \mathbb{H} : \underline{\underline{\varepsilon}}(\underline{u}_H) \\ &\int_\Omega \underline{\underline{\sigma}}_H : \underline{\underline{\varepsilon}}(\underline{v}_H)\, d\Omega = \int_\Omega \underline{f} \cdot \underline{v}_H d\Omega + \int_{\partial_g\Omega} \underline{g} \cdot \underline{v}_H dS, \qquad \forall \underline{v}_H \in \mathrm{KA}^0_H(\Omega) \end{aligned} \tag{10}$$

Of course the approximation is due to the fact that in most cases $\underline{\underline{\sigma}}_H \notin \mathrm{SA}(\Omega)$.

After introducing the matrix $\underline{\boldsymbol{\varphi}}_H$ of shape functions which form a basis of $\mathrm{KA}_H(\Omega)$ (extended to Dirichlet degrees of freedom) and the vector of nodal unknowns $\mathbf{u}$ so that $\underline{u}_H = \underline{\boldsymbol{\varphi}}_H \mathbf{u}$, the classical finite element method leads to the linear system:

$$\begin{pmatrix} \mathbf{K}_{rr} & \mathbf{K}_{rd} \\ \mathbf{K}_{dr} & \mathbf{K}_{dd} \end{pmatrix} \begin{pmatrix} \mathbf{u}_r \\ \mathbf{u}_d \end{pmatrix} = \begin{pmatrix} \mathbf{f}_r \\ \mathbf{f}_d \end{pmatrix} + \begin{pmatrix} 0 \\ \boldsymbol{\lambda}_d \end{pmatrix} \tag{11}$$

where $\mathbf{K}$ is the (symmetric semi positive definite) stiffness matrix and $\mathbf{f}$ is the vector of generalized forces; Subscript d stands for Dirichlet degrees of freedom (where displacements are prescribed) and Subscript r represents the remaining degrees of freedom so that unknowns are $\mathbf{u}_r$ and $\boldsymbol{\lambda}_d$ where Vector $\boldsymbol{\lambda}_d$ represents the nodal reactions:

$$\boldsymbol{\lambda}_d^T = \int_\Omega \underline{\underline{\sigma}}_H : \underline{\underline{\varepsilon}}\left(\underline{\boldsymbol{\varphi}}_{H\,d}\right) d\Omega - \int_\Omega \underline{f} \cdot \underline{\boldsymbol{\varphi}}_{H\,d} d\Omega - \int_{\partial_g\Omega} \underline{g} \cdot \underline{\boldsymbol{\varphi}}_{H\,d} dS \tag{12}$$

where $\underline{\boldsymbol{\varphi}}_{H\,d}$ is the matrix of shape functions restricted to the Dirichlet nodes and $\underline{n}$ the outer normal vector.

2.1.2 A posteriori error estimation

Upper bound of the discretization error The estimator we choose is based on the error in constitutive relation, which gives a guaranteed estimator for the discretization error.

The fundamental relation is the following (Prager-Synge theorem, see for instance [15]):

$$\begin{aligned} &\forall (\hat{\underline{u}}, \hat{\underline{\underline{\sigma}}}) \in \mathrm{KA}(\Omega) \times \mathrm{SA}(\Omega), \\ &\left\| \underline{\underline{\varepsilon}}(\underline{u}_{ex}) - \underline{\underline{\varepsilon}}(\hat{\underline{u}}) \right\|^2_{\mathbb{H},\Omega} + \left\| \underline{\underline{\sigma}}_{ex} - \hat{\underline{\underline{\sigma}}} \right\|^2_{\mathbb{H}^{-1},\Omega} = e^2_{CR_\Omega}(\hat{\underline{u}}, \hat{\underline{\underline{\sigma}}}) \end{aligned} \tag{13}$$

We note $|||\underline{v}|||_\Omega = \left\| \underline{\underline{\varepsilon}}(\underline{v}) \right\|_{\mathbb{H},\Omega}$ the energy norm of the displacement, and since we can choose $\hat{\underline{u}} = \underline{u}_H \in \mathrm{KA}(\Omega)$, we retain the following upper bound for the error $\underline{e}_{discr} = \underline{u}_{ex} - \underline{u}_H$:

$$e_{discr} := |||\underline{e}_{discr}|||_\Omega \leqslant e_{CR_\Omega}(\underline{u}_H, \hat{\underline{\underline{\sigma}}}) \tag{14}$$

The construction of $\hat{\underline{\underline{\sigma}}} \in \mathrm{SA}(\Omega)$ is a complex problem solved by various approaches [14, 20, 23, 26].

The techniques [14, 23, 26] are two-steps procedures. The first step consist in building a set of equilibrated tractions or works along the edges of the elements of the mesh. Each method proposes its own strategy to reconstruct such tractions. The second step, common to all methods, is the solving of Neumann problems on each element using the equilibrated tractions as Neumann conditions.

The technique developed in [20] does not require equilibrated fluxes but only the solving local problems on star-patches (a star patch, denoted by ω_i, is composed of the elements sharing the vertex i and corresponds to the support of the shape function associated to the vertex i). This is the reason why the technique is sometimes called the flux-free technique.

Local problems (on element or star-patch) are usually solved on a space of finite dimension which is richer than the finite element space restrained to the support of the local problem. The space is enriched either thanks to higher degree polynomial shape functions or thanks to mesh refinement (each element being divided into several smaller elements).

Lower bound of the discretization error A lower bound of the true error can be obtained using Cauchy-Schwarz inequality in the following residual equation:

$$\begin{aligned} &\forall \underline{w} \in \mathrm{KA}^0(\Omega) \\ \int_\Omega \underline{\underline{\varepsilon}}\left(\underline{u}_{ex} - \underline{u}_H\right) : \mathbb{H} : \underline{\underline{\varepsilon}}\left(\underline{w}\right) d\Omega &= \int_\Omega \underline{f} \cdot \underline{w} d\Omega + \int_{\partial_g\Omega \bigcap \partial\Omega} \underline{g} \cdot \underline{w} dS - \int_\Omega \underline{\underline{\sigma}}_H : \underline{\underline{\varepsilon}}\left(\underline{w}\right) d\Omega \\ &:= R_H(\underline{w}) \end{aligned} \tag{15}$$

Therefore, every displacement field $\underline{w} \in \mathrm{KA}^0(\Omega)\backslash\{\underline{0}\}$ can be used to obtain the following strict lower bound [24]:

$$|||\underline{u}_{ex} - \underline{u}_H|||_\Omega \geqslant \frac{|R_H(\underline{w})|}{|||\underline{w}|||_\Omega} \tag{16}$$

The accuracy of the lower bound depends on the quality of the continuous field $\underline{w}$, which is often called continuous error estimate. Indeed, the lower bound equals the true error for the continuous field $\underline{w} = \underline{u}_{ex} - \underline{u}_H$. The construction of $\underline{w}$ was mainly studied in [4, 20, 9] of which we recall the main results.

In [4], the error is estimated thanks to an implicit residual-based error estimator with local solvings on elements. A continuous field $\underline{w} \in \mathrm{KA}^0(\Omega)\backslash\{\underline{0}\}$ is constructed from the element estimators by averaging on the edges.

In [20], local problems on star-patches are solved to compute an upper bound of the error :

$$\begin{aligned} &\text{Find } \underline{e}^i \in \mathrm{KA}^0(\omega_i) \text{ such that } \forall \underline{v} \in \mathrm{KA}^0(\omega_i) \\ &a(\underline{e}^i, \underline{v}) = R_H(\varphi_H^i \underline{v}) \end{aligned} \tag{17}$$

where φ_H^i is the shape function associated to the central node i of the star-patch. The previous problem is solved on a space $\mathrm{KA}_h^0(\omega_i)$ richer than $\mathrm{KA}_H^0(\omega_i)$.

The proposed continuous displacement field is:

$$\underline{w} = \Pi_h\left(\sum_{i \in \mathcal{V}} \varphi_H^i \underline{e}^i\right) \tag{18}$$

where Π_h is the projector on the space $\mathrm{KA}_h^0(\omega_i)$. Discontinuities between star-patches vanish thanks to the multiplication by φ_H^i. The projector eases the computation of $R_H(\underline{w})$.

Note that in the same article, an enhanced estimate is proposed to better the lower bound :

$$|||\underline{u}_{ex} - \underline{u}_H|||_\Omega^2 \geqslant \frac{R_H(\underline{w})^2}{|||\underline{w}|||_\Omega^2 - |||\underline{e}_G|||_\Omega^2} \tag{19}$$

where $\underline{e}_G$ is the solution of the following global problem:

$$\begin{aligned} &\text{Find } \underline{e}_G \in \mathrm{KA}_H^0(\Omega) \text{ such that } \forall \underline{v} \in \mathrm{KA}_h^0(\omega_i) \\ &a(\underline{e}_G, \underline{v}) = -a(\underline{w}, \underline{v}) \end{aligned} \tag{20}$$

In [9], the statically admissible stress field is built from a displacement field which is the sum of the solutions of local problems on star-patches with homogeneous boundary Dirichlet conditions:

$$\begin{aligned} &\text{Find } \underline{w}^i \in \mathrm{KA}^{0,\omega_i}(\omega_i) \text{ such that } \forall \underline{v} \in \mathrm{KA}^{0,\omega_i}(\omega_i) \\ \int_{\omega_i} \underline{\underline{\varepsilon}}\left(\underline{w}^i\right) : \mathbb{H} : \underline{\underline{\varepsilon}}\left(\underline{v}\right) d\Omega &= \int_{\omega_i} (\varphi_i \underline{f} - \underline{\underline{\sigma}}_H \underline{\mathrm{grad}}(\varphi_i)) \underline{v} d\Omega - \int_{\omega_i} \varphi_i \underline{\underline{\sigma}}_H : \underline{\underline{\varepsilon}}\left(\underline{v}\right) d\Omega \end{aligned} \tag{21}$$

where KA^{0,ω_i} is the space of continuous displacement fields that equal to zero on the boundary of the star-patch ω_i. The continuous field $\underline{w} \in \mathrm{KA}^0(\Omega)$ is the sum of the solutions of the previous problem:

$$\underline{w} = \sum_{i \in \mathcal{V}} \underline{w}^i \tag{22}$$

To conclude this brief review, there exist various techniques to construct $\underline{w}$. They always take advantage of the field computed during the estimation of an upper bound so that the extra-cost is very limited.

2.1.3 Substructured formulation

Let us consider a decomposition of domain Ω in N_{sd} regular open subsets $(\Omega^{(s)})_s$ such that $\Omega^{(s)} \bigcap \Omega^{(s')} = \varnothing$ for $s \neq s'$ and $\bar{\Omega} = \bigcup_s \bar{\Omega}^{(s)}$. We note $\partial_g\Omega^{(s)} = \partial\Omega^{(s)} \bigcap \partial_g\Omega$ the Neumann border of subdomains.

The mechanical problem on the substructured configuration writes :

$$\forall s \begin{cases} \underline{u}^{(s)} \in \mathrm{KA}(\Omega^{(s)}) \\ \underline{\underline{\sigma}}^{(s)} \in \mathrm{SA}(\Omega^{(s)}) \\ e_{CR_{\Omega^{(s)}}}(\underline{u}^{(s)}, \underline{\underline{\sigma}}^{(s)}) = 0 \end{cases} \tag{23}$$

and

$$\forall (s,s') \text{ such that } \Omega^{(s)} \text{ and } \Omega^{(s')} \text{ are adjacent } \begin{cases} \mathrm{tr}(\underline{u}^{(s)}) = \mathrm{tr}(\underline{u}^{(s')}) \text{ on } \Gamma^{(s,s')} \\ \underline{\underline{\sigma}}^{(s)} \cdot \underline{n}^{(s)} + \underline{\underline{\sigma}}^{(s')} \cdot \underline{n}^{(s')} = \underline{0} \text{ on } \Gamma^{(s,s')} \end{cases} \tag{24}$$

The set of fields $\underline{u}$ defined on Ω such that $\underline{u}_{|\Omega^{(s)}} \in \mathrm{KA}(\Omega^{(s)})$ without interface continuity is a broken space which we note $\mathrm{KA}(\bigcup \Omega^{(s)})$.

2.1.4 Finite element approximation for the substructured problem

We assume that the mesh of $\bar{\Omega}$ and the substructuring are conforming. This hypothesis implies that each element only belongs to one subdomain and nodes are matching on the interfaces. For each subdomain, let us denote by the subscript b the degrees of freedom on the boundary of the subdomain and by the subscript i the degrees of fredom inside the subdomain.

Let $\mathbf{t}^{(s)}$ be the discrete trace operator on the interface. $\mathbf{t}^{(s)}$ enables to cast degrees of freedom from a complete subdomain to its interface. Using an adapted ordering, we have

$$\mathbf{t}^{(s)} = \begin{pmatrix} \mathbf{I}_{bb} & \mathbf{0}_{bi} \end{pmatrix} \tag{25}$$

Therefore ${\mathbf{t}^{(s)}}^T$ is the extension by zero operator: it extends data supported by the boundary to the whole subdomain.

Let us introduce the unknown nodal reaction on the interface $\boldsymbol{\lambda}^{(s)}$, the equilibrium of each subdomain writes:

$$\mathbf{K}^{(s)}\mathbf{u}^{(s)} = \mathbf{f}^{(s)} + {\mathbf{t}^{(s)}}^T \boldsymbol{\lambda}^{(s)} \tag{26}$$

Let $(\mathbf{A}^{(s)})$ and $(\mathbf{B}^{(s)})$ be the primal and dual assembly operator. Those operators are signed boolean operators. The dual operator $(\mathbf{B}^{(s)})$ enables to express the continuity of displacements and the primal operator $(\mathbf{A}^{(s)})$ enables to express the mechanical equilibrium of interface. Their number of columns is equal to the number of boundary degrees of freedom. $\mathbf{A}^{(s)}$ injects the boundary degrees of freedom of $\Omega^{(s)}$ in the global interface. Thus the number of rows of $\mathbf{A}^{(s)}$ is equal to the number of degrees of freedom on the global interface. The number of rows of $\mathbf{B}^{(s)}$ is equal to the number of connections between pairs of neighboring degrees of freedom.

In the case of two subdomains, we have $\sum_s \mathbf{B}^{(s)}\mathbf{t}^{(s)}\mathbf{u}^{(s)} = \mathbf{t}^{(1)}\mathbf{u}^{(1)} - \mathbf{t}^{(2)}\mathbf{u}^{(2)}$ and $\sum_s \mathbf{A}^{(s)}\boldsymbol{\lambda}^{(s)} = \boldsymbol{\lambda}^{(1)} + \boldsymbol{\lambda}^{(2)}$. For more details on the assembly operators, the reader can refer to [11].

The discrete counterpart of the interface admissibility equations is:

$$\begin{cases} \sum_s \mathbf{B}^{(s)}\mathbf{t}^{(s)}\mathbf{u}^{(s)} = 0 \\ \sum_s \mathbf{A}^{(s)}\boldsymbol{\lambda}^{(s)} = 0 \end{cases} \tag{27}$$

Equations (26) and (27) form the discrete substructured system, which is equivalent to the global problem (11).

2.1.5 Domain decomposition solvers and admissible fields

Domain decomposition solvers are well described in many papers (see for instance [11] and the associated bibliography). The principle is to condense the global problem on the interface to create a smaller problem. In classical algorithms such as BDD [16] and FETI [7], this new interface problem is solved iteratively thanks to a projected preconditioned conjugate gradient. Each iteration implies two parallel solvings on subdomains with two dual operators (one for the preconditioning step, one for the direct step) so that local problems with Neumann boundary conditions and local problems with Dirichlet boundary conditions are alternatively solved. In [22] it was proved that the following fields could be processed at no extra cost:

- $(\underline{u}_D^{(s)})_s \in \mathrm{KA}(\Omega)$: displacement field which results from a Dirichlet problem and which is thus globally admissible
- $(\boldsymbol{\lambda}_N^{(s)})_s$: nodal reactions which are balanced at the interface.
- $(\underline{u}_N^{(s)})_s \in \mathrm{KA}(\bigcup \Omega^{(s)})$: displacement field which results from a Neumann problem and which is not globally admissible

- $\underline{\underline{\sigma}}_N^{(s)}$: the stress field associated to $\underline{u}_N^{(s)}$ ($\underline{\underline{\sigma}}_N^{(s)} = \mathbb{H} : \underline{\underline{\varepsilon}}\left(\underline{u}_N^{(s)}\right)$). It can be used (with additional input $\boldsymbol{\lambda}_N^{(s)}$) to build in parallel stress fields $\hat{\underline{\underline{\sigma}}}_N^{(s)}$ which are statically admissible $\hat{\underline{\underline{\sigma}}}_N = (\hat{\underline{\underline{\sigma}}}_N^{(s)})_s \in \mathrm{SA}(\Omega)$ using dedicated methods such as [14, 20, 23, 26].

In [28] we proved the following result where α (denoted $\sqrt{\mathbf{r}^T\mathbf{z}}$ in [28]) is the preconditioner-norm of the residual, a quantity that is actually computed by the solver:

$$\alpha := \|\underline{u}_N - \underline{u}_D\|_{\mathbb{H},\Omega} = \sqrt{\mathbf{r}^T\mathbf{z}} \tag{28}$$

2.2 A posteriori upper bound of the error in substructured context

In [22], a first parallel error estimator in substructured context was introduced. It is based on the error in constitutive relation and reads :

$$|||\underline{u}_{ex} - \underline{u}_D|||_\Omega = \sqrt{\sum_s |||\underline{u}_{ex}^{(s)} - \underline{u}_D^{(s)}|||^2_{\Omega^{(s)}}} \leqslant \sqrt{\sum_s e^2_{CR_{\Omega^{(s)}}}(\underline{u}_D^{(s)}, \hat{\underline{\underline{\sigma}}}_N^{(s)})} \tag{29}$$

In [28], we showed that this estimator mixes two different sources of error : the discretization error which is inherent to the use of the finite element method and the algebraic error which is due to the use of an iterative solver which would not exist for a direct solver. The algebraic error monitors the convergence of the solver and can be made as small as wished. Therefore, a second parallel error estimator was proposed in [28]:

$$|||\underline{u}_{ex} - \underline{u}_N|||_\Omega \leqslant \alpha + \sqrt{\sum_s e^2_{CR_{\Omega^{(s)}}}(\underline{u}_N^{(s)}, \hat{\underline{\underline{\sigma}}}_N^{(s)})} \tag{30}$$

$$|||\underline{u}_{ex} - \underline{u}_D|||_\Omega \leqslant \alpha + \sqrt{\sum_s e^2_{CR_{\Omega^{(s)}}}(\underline{u}_N^{(s)}, \hat{\underline{\underline{\sigma}}}_N^{(s)})} \tag{31}$$

This estimator separates the two sources of error. When the solver has converged the two displacements fields $\underline{u}_N$ and $\underline{u}_D$ are identical and equal to $\underline{u}_H$ (the algebraic error α is very close to zero). $\sqrt{\sum_s e^2_{CR_{\Omega^{(s)}}}(\underline{u}_N^{(s)}, \hat{\underline{\underline{\sigma}}}_N^{(s)})}$ is the estimation of the discretization error. As a consequence, at convergence, the estimators (29), (30) and (31) are identical.

3 Lower bound of the error in substructured context

In this section, we extend sequential results to demonstrate guaranteed lower bounds of the error in substructured context. Moreover, we prove a theorem that enables the separation of sources of error in the lower bound. We also develop the methodology to build a continuous error estimate from parallel error estimation procedure. Finally, we extend those results to goal-oriented error estimation.

3.1 A first lower bound of the error

Theorem 1. *Let $\underline{u}_{ex} \in \mathrm{KA}(\Omega)$ be the exact solution, $(\underline{u}_D^{(s)})_s \in \mathrm{KA}(\Omega)$ the displacement field defined in 2.1.5 and $\underline{w} \in \mathrm{KA}^0(\Omega)\backslash\{\underline{0}\}$ then*

$$|||\underline{u}_{ex} - \underline{u}_D|||_\Omega \geqslant \frac{|R_D(\underline{w})|}{\sqrt{\sum_s |||\underline{w}^{(s)}|||^2_{\Omega^{(s)}}}} \tag{32}$$

with

$$R_D(\underline{w}) = \sum_s R_D^{(s)}(\underline{w}^{(s)}) \tag{33}$$

$$R_D^{(s)}(\underline{w}^{(s)}) := \int_{\Omega^{(s)}} \underline{f}\cdot\underline{w}^{(s)} d\Omega + \int_{\partial_g\Omega^{(s)}} \underline{g}\cdot\underline{w}^{(s)} dS - \int_{\Omega^{(s)}} \underline{\underline{\varepsilon}}\left(\underline{u}_D^{(s)}\right) : \mathbb{H} : \underline{\underline{\varepsilon}}\left(\underline{w}^{(s)}\right) d\Omega^{(s)} \tag{34}$$

Proof. This property is the direct application of (16) where we replace the displacement field $\underline{u}_H \in \mathrm{KA}(\Omega)$ by $(\underline{u}_D^{(s)})_s \in \mathrm{KA}(\Omega)$. The residual $R_D(\underline{w})$ can be rewritten:

$$\begin{aligned} R_D(\underline{w}) &= \sum_s R_D^{(s)}(\underline{w}^{(s)}) \\ &= \sum_s \left(\int_{\Omega^{(s)}} \underline{f}\cdot\underline{w}^{(s)} d\Omega + \int_{\partial_g\Omega^{(s)}} \underline{g}\cdot\underline{w}^{(s)} dS - \int_{\Omega^{(s)}} \underline{\underline{\varepsilon}}\left(\underline{u}_D^{(s)}\right) : \mathbb{H} : \underline{\underline{\varepsilon}}\left(\underline{w}^{(s)}\right) d\Omega^{(s)} \right) \\ &= L(\underline{w}) - a(\underline{u}_D, \underline{w}) \end{aligned} \tag{35}$$

□

In practice, we choose $\underline{w}^{(s)} \in \mathrm{KA}^{00}(\Omega^{(s)})\backslash\{\underline{0}\} \subset \mathrm{KA}^{0}(\Omega)\backslash\{\underline{0}\}$, which corresponds to imposing the nullity along the interface and which is inexpensive since it does not imply exchanges between subdomains. In subsection 3.3, we will give details about the computation of $\underline{w}$. Therefore the computation of a lower bound is as parallel as for the upper bound. In the assessments in section 4, we will verify that this lower bound is as accurate as the one obtained in the sequential context and that the quality neither depends on the approach (primal or dual) nor on the substructuring. Moreover, this lower bound is computable whatever the state of the iterative solver (converged or not).

3.2 Lower bound with separation of sources of error

Continuing the philosophy of separating the sources of error as detailed in [28], we propose a second lower bound :

Theorem 2. *Let $\underline{u}_{ex} \in \mathrm{KA}(\Omega)$ be the exact solution, $(\underline{u}_D^{(s)})_s \in \mathrm{KA}(\Omega)$ and $(\underline{u}_N^{(s)})_s \in \mathrm{KA}(\bigcup \Omega^{(s)})$ the displacement fields defined in 2.1.5 and $\underline{w} \in \mathrm{KA}^{0}(\Omega)\backslash\{\underline{0}\}$, then*

$$|||\underline{u}_{ex} - \underline{u}_D|||_\Omega \geqslant \left| \frac{|R_N(\underline{w})|}{\sqrt{\sum_s |||\underline{w}^{(s)}|||^2_{\Omega^{(s)}}}} - \frac{|a(\underline{u}_D - \underline{u}_N, \underline{w})|}{\sqrt{\sum_s |||\underline{w}^{(s)}|||^2_{\Omega^{(s)}}}} \right| \tag{36}$$

which leads to the coarser bound

$$|||\underline{u}_{ex} - \underline{u}_D|||_\Omega \geqslant \frac{|R_N(\underline{w})|}{\sqrt{\sum_s |||\underline{w}^{(s)}|||^2_{\Omega^{(s)}}}} - \alpha \tag{37}$$

with

$$R_N(\underline{w}) = \sum_s R_N^{(s)}(\underline{w}^{(s)}) \tag{38}$$

$$R_N^{(s)}(\underline{w}^{(s)}) := \int_{\Omega^{(s)}} \underline{f} \cdot \underline{w}^{(s)} d\Omega + \int_{\partial_g \Omega^{(s)}} \underline{g} \cdot \underline{w}^{(s)} dS - \int_{\Omega^{(s)}} \underline{\underline{\varepsilon}}\left(\underline{u}_N^{(s)}\right) : \mathbb{H} : \underline{\underline{\varepsilon}}\left(\underline{w}^{(s)}\right) d\Omega^{(s)} \tag{39}$$

Proof. The proof of the first inequality is based on theorem 1 and on the triangle inequality :

$$\begin{aligned}
|||\underline{u}_{ex} - \underline{u}_D|||_\Omega &\geqslant \frac{|R_D(\underline{w})|}{\sqrt{\sum_s |||\underline{w}^{(s)}|||^2_{\Omega^{(s)}}}} \\
|||\underline{u}_{ex} - \underline{u}_D|||_\Omega &\geqslant \frac{|L(\underline{w}) - a(\underline{u}_D, \underline{w})|}{\sqrt{\sum_s |||\underline{w}^{(s)}|||^2_{\Omega^{(s)}}}} \\
&\geqslant \frac{|L(\underline{w}) - a(\underline{u}_N, \underline{w}) - a(\underline{u}_D - \underline{u}_N, \underline{w})|}{\sqrt{\sum_s |||\underline{w}^{(s)}|||^2_{\Omega^{(s)}}}} \\
&\geqslant \left| \frac{|R_N(\underline{w})|}{\sqrt{\sum_s |||\underline{w}^{(s)}|||^2_{\Omega^{(s)}}}} - \frac{|a(\underline{u}_D - \underline{u}_N, \underline{w})|}{\sqrt{\sum_s |||\underline{w}^{(s)}|||^2_{\Omega^{(s)}}}} \right|
\end{aligned} \tag{40}$$

which proves (36). (37) is simply based on the remark that

$$\left| \frac{|R_N(\underline{w})|}{\sqrt{\sum_s |||\underline{w}^{(s)}|||^2_{\Omega^{(s)}}}} - \frac{|a(\underline{u}_D - \underline{u}_N, \underline{w})|}{\sqrt{\sum_s |||\underline{w}^{(s)}|||^2_{\Omega^{(s)}}}} \right| \geqslant \frac{|R_N(\underline{w})|}{\sqrt{\sum_s |||\underline{w}^{(s)}|||^2_{\Omega^{(s)}}}} - \frac{|a(\underline{u}_D - \underline{u}_N, \underline{w})|}{\sqrt{\sum_s |||\underline{w}^{(s)}|||^2_{\Omega^{(s)}}}} \tag{41}$$

and using twice the Cauchy-Schwarz inequality, we have :

$$\begin{aligned}
|a(\underline{u}_D - \underline{u}_N, \underline{w})| &= |\sum_s \int_{\Omega^{(s)}} \underline{\underline{\varepsilon}}\left(\underline{u}_D^{(s)} - \underline{u}_N^{(s)}\right) : \mathbb{H} : \underline{\underline{\varepsilon}}\left(\underline{w}^{(s)}\right)| \\
&\leqslant \sum_s |||\underline{u}_N^{(s)} - \underline{u}_D^{(s)}|||_{\Omega^{(s)}} |||\underline{w}^{(s)}|||_{\Omega^{(s)}} \\
&\leqslant \sqrt{\sum_s |||\underline{u}_N^{(s)} - \underline{u}_D^{(s)}|||^2_{\Omega^{(s)}}} \sqrt{\sum_s |||\underline{w}^{(s)}|||^2_{\Omega^{(s)}}}
\end{aligned} \tag{42}$$

Finally, using the equality (28):

$$|||\underline{u}_{ex} - \underline{u}_D|||_\Omega \geqslant \frac{|R_N(\underline{w})|}{\sqrt{\sum_s |||\underline{w}^{(s)}|||^2_{\Omega^{(s)}}}} - \alpha \tag{43}$$

□

As said earlier, the term α is a measure of the residual so it is purely algebraic whereas the first term of the inequality is mainly driven by the discretization error. During the first iterations, the second lower bound is not accurate because the algebraic error prevails so that $\frac{|L(\underline{w})-a(\underline{u}_N,\underline{w})|}{\sqrt{\sum_s |||\underline{w}^{(s)}|||^2_{\Omega^{(s)}}}} - \alpha$ is negative and it is a trivial lower bound of the positive true error $|||\underline{u}_{ex} - \underline{u}_D|||_\Omega$. When the solver reaches convergence, the three lower bounds in theorems 1 and 2 are identical.

3.3 Reconstruction of admissible field $\underline{w}$

The upper bounds of the error (30) or (31) require the construction of a statically admissible field $(\underline{\underline{\hat{\sigma}}}_N^{(s)})_s$. In case the flux-free technique is chosen, it is possible to construct a continuous field $\underline{w}^{(s)} \in \mathrm{KA}^{00}(\Omega^{(s)})$ using the methodology developed in [20] for each subdomain in parallel. As a consequence,

$$\left((\Pi_h(\sum_{i\in\mathcal{V}} \phi_H^i \underline{e}^i))^{(s)} \right)_s \in \mathrm{KA}(\bigcup \Omega^{(s)}) \tag{44}$$

In order to have $\underline{w}^{(s)} \in \mathrm{KA}^{00}(\Omega^{(s)})$, we choose not to sum the contributions from the nodes located on the interface of the subdomain (s). They will be denoted as $\mathcal{V}_\Gamma$. $\underline{w}^{(s)}$ is defined by :

$$\underline{w}^{(s)} = (\Pi_h(\sum_{i\in\mathcal{V}\setminus\mathcal{V}_\Gamma} \varphi_H^i \underline{e}^i))^{(s)} \tag{45}$$

Therefore $\underline{w}^{(s)} \in \mathrm{KA}^{00}(\Omega^{(s)})$ and $\underline{w} = (\underline{w}^{(s)})_s \in \mathrm{KA}^0(\Omega)$.

Remark. Using the subtle trick in [20], the computation of the discretized fields is eased. Indeed:

$$\mathbf{w}^{(s)} = \sum_{i\in\mathcal{V}\setminus\mathcal{V}_\Gamma} \boldsymbol{\varphi}_\mathbf{H}^\mathbf{i} \odot \mathbf{e}^{\mathbf{i}(s)} \tag{46}$$

and

$$\mathbf{R_D}^{(s)}(\mathbf{w}^{(s)}) = \sum_{i\in\mathcal{V}-\mathcal{V}_\Gamma} \boldsymbol{\varphi}_\mathbf{H}^\mathbf{i} \odot \mathbf{R_D}^{(s)}(\mathbf{e}^{\mathbf{i}(s)}) \tag{47}$$

where $\boldsymbol{\varphi}_\mathbf{H}^\mathbf{i}$ gathers the nodal values of the shape function φ_H^i projected on the richer space used to solve the star-patch problem whose $\mathbf{e}^{\mathbf{i}(s)}$ is the discretized solution and where $\odot$ represents the term by term multiplication.

Remark. If the method chosen to construct the admissible field is based on elements problems [4], it is always possible to construct a displacement field $\underline{w}^{(s)} \in \mathrm{KA}^{00}(\Omega^{(s)})$ by computing the mean value along the edges inside the subdomains and imposing zero along the interfaces between subdomains.

3.4 Goal-oriented error estimation

Goal-oriented error estimation offers the possibility to have upper and lower bounds of the unknown exact value of a quantity of interest. Among various techniques, extractors (see [1] for instance) are the most common tools to define linear quantities of interest. They lead to the definition and the solving of an adjoint problem.

3.4.1 Definition of the linear quantity of interest and of the adjoint problem

Let $\widetilde{L}$ be the linear functional defining the quantity of interest I:

$$I = \widetilde{L}(\underline{u}) = \int_\Omega (\underline{\underline{\sigma}}_\Sigma : \underline{\underline{\varepsilon}}(\underline{u}) + \underline{f}_\Sigma \underline{u}) d\Omega \tag{48}$$

where $\underline{\underline{\sigma}}_\Sigma$ and $\underline{f}_\Sigma$ are extractors.

We introduce the affine subspace of statically admissible fields (adjoint SA-fields) for the adjoint problem:

$$\widetilde{\mathrm{SA}}(\omega) = \left\{ \underline{\underline{\tau}} \in \left(\mathrm{L}^2(\omega)\right)^{d\times d}_{\mathrm{sym}}; \quad \forall \underline{v} \in \mathrm{KA}^{00}(\omega), \quad \int_\omega \underline{\underline{\tau}} : \underline{\underline{\varepsilon}}(\underline{v})\, d\omega = \tilde{L}(\underline{v}) \right\} \tag{49}$$

The adjoint problem set on Ω can be formulated as:

$$\text{Find } \left(\underline{\tilde{u}}_{ex}, \underline{\underline{\tilde{\sigma}}}_{ex}\right) \in \mathrm{KA}^0(\Omega) \times \widetilde{\mathrm{SA}}(\Omega) \text{ such that } e_{CR_\Omega}(\underline{\tilde{u}}_{ex}, \underline{\underline{\tilde{\sigma}}}_{ex}) = 0 \tag{50}$$

The solution to this problem, named exact solution, exists and is unique.

Remark. The formulation (50) is equivalent to the classical following formulation:

$$\text{Find } \underline{\widetilde{u}}_{ex} \in \mathrm{KA}^0(\Omega) \text{ such that } \forall \underline{v} \in \mathrm{KA}^0(\Omega),\ a(\underline{\widetilde{u}}_{ex}, \underline{v}) = \widetilde{L}(\underline{v}) \tag{51}$$

where a is the classical bilinear form (8).

The adjoint problem is usually solved with the finite element method. The mesh can differ from the one used for the forward problem. The approximated adjoint displacement is $\underline{\widetilde{u}}_{\widetilde{H}}$.

The discretization error for the adjoint problem is

$$e_{discr} = |||\underline{\widetilde{e}}_{discr}|||_{\Omega} = |||\underline{\widetilde{u}}_{ex} - \underline{\widetilde{u}}_{\widetilde{H}}|||_{\Omega} \tag{52}$$

3.4.2 Error estimation of quantities of interest

As said earlier the adjoint problem is meant to extract one quantity of interest in the forward problem. Let $I_{ex} = \tilde{L}(\underline{u}_{ex})$ be the unknown exact value of the quantity of interest. $I_H = \tilde{L}(\underline{u}_D)$ is an approximation of this quantity of interest.

A bounding of the exact value of the quantity of interest I_{ex} is [12, 13]:

$$|I_{ex} - I_H - I_{HH2}| \leqslant \frac{1}{2} e_{CR_\Omega}(\underline{u}_H, \underline{\hat{\sigma}}_H) e_{CR_\Omega}(\underline{\widetilde{u}}_{\widetilde{H}}, \underline{\hat{\widetilde{\sigma}}}_{\widetilde{H}}) \tag{53}$$

where

$$I_{HH2} = \int_{\Omega} \frac{1}{2} (\underline{\hat{\widetilde{\sigma}}}_{\widetilde{H}} + \mathbb{H} : \underline{\varepsilon}(\underline{\widetilde{u}}_{\widetilde{H}})) : \mathbb{H}^{-1} : (\underline{\hat{\sigma}}_H - \mathbb{H} : \underline{\varepsilon}(\underline{u}_H)) d\Omega \tag{54}$$

and where $\underline{\hat{\widetilde{\sigma}}}_{\widetilde{H}} \in \widetilde{\mathrm{SA}}_H(\Omega)$.

The error on the quantity of interest can also be estimated using the parallelogram identity [24]:

$$\begin{cases} I_{ex} - I_H = \widetilde{L}(\underline{e}_{discr}) = a(\underline{e}_{discr}, \underline{\widetilde{e}}_{discr}) \\ \qquad\qquad\quad = a(\kappa \underline{e}_{discr}, \dfrac{1}{\kappa} \underline{\widetilde{e}}_{discr}) \\ \qquad\qquad\quad = \dfrac{1}{4} [|||\kappa \underline{e}_{discr} + \dfrac{1}{\kappa} \underline{\widetilde{e}}_{discr}|||_{\Omega}^2 - |||\kappa \underline{e}_{discr} - \dfrac{1}{\kappa} \underline{\widetilde{e}}_{discr}|||_{\Omega}^2] \end{cases} \tag{55}$$

where κ is a scalar parameter whose optimal value is:

$$\kappa = \frac{e_{CR_\Omega}(\underline{\widetilde{u}}_H, \underline{\hat{\widetilde{\sigma}}}_H)}{e_{CR_\Omega}(\underline{u}_H, \underline{\hat{\sigma}}_H)} \tag{56}$$

This optimal value minimizes the difference and thus improves the quality of the bounding. Introducing upper and lower bounds:

$$\begin{aligned} \beta_{inf}^{+} &\leqslant |||\kappa \underline{e}_{discr} + \frac{1}{\kappa} \underline{\widetilde{e}}_{discr}|||_{\Omega}^2 \leqslant \beta_{sup}^{+} \\ \beta_{inf}^{-} &\leqslant |||\kappa \underline{e}_{discr} - \frac{1}{\kappa} \underline{\widetilde{e}}_{discr}|||_{\Omega}^2 \leqslant \beta_{sup}^{-} \end{aligned} \tag{57}$$

it is possible to obtain lower and upper bounds of the error on the quantity of interest :

$$\frac{1}{4}\beta_{inf}^{+} - \frac{1}{4}\beta_{sup}^{-} \leqslant I_{ex} - I_H \leqslant \frac{1}{4}\beta_{sup}^{+} - \frac{1}{4}\beta_{inf}^{-} \tag{58}$$

3.4.3 Application to the substructured context

Let us suppose that the forward and adjoint problems are solved on the same mesh and on the same substructuring. Since the two problems share the same stiffness matrices on every subdomain, they can be solved simultaneously using a block algorithm. Following the same methodology described in section 2.1.5, one can compute the following admissible fields for the adjoint problem:

- $(\underline{\widetilde{u}}_D^{(s)})_s \in \mathrm{KA}^0(\Omega)$: displacement field which results from a Dirichlet problem and which is thus globally admissible
- $\widetilde{\boldsymbol{\lambda}}_N^{(s)}$: nodal reactions which are balanced at the interface.
- $(\underline{\widetilde{u}}_N^{(s)})_s \in \mathrm{KA}(\bigcup \Omega^{(s)})$: displacement field which results from a Neumann problem and which is not globally admissible
- $\underline{\widetilde{\sigma}}_N^{(s)}$: stress field associated to $\underline{\widetilde{u}}_N^{(s)}$ ($\underline{\widetilde{\sigma}}_N^{(s)} = \mathbb{H} : \underline{\varepsilon}\left(\underline{\widetilde{u}}_N^{(s)}\right)$). It can be used (with additional input $\widetilde{\boldsymbol{\lambda}}_N^{(s)}$) to build in parallel stress fields $\underline{\hat{\widetilde{\sigma}}}_N^{(s)}$ which are statically admissible $\underline{\hat{\widetilde{\sigma}}}_N = (\underline{\hat{\widetilde{\sigma}}}_N^{(s)})_s \in \widetilde{\mathrm{SA}}(\Omega)$.

We also have the equality that expresses the distance between the Neumann and Dirichlet displacement fields in terms of the algebraic residual $\widetilde{\alpha}$ of the adjoint problem :

$$\|\underline{\widetilde{u}}_N - \underline{\widetilde{u}}_D\|_{\mathbb{H},\Omega} = \widetilde{\alpha} \tag{59}$$

For more details about the computation of those fields, the reader can refer to [27].

The upper bounds of the global error presented in section 2.2 and the lower bound in the theorem 1 can be applied on the adjoint problem.

Upper and lower bounds for goal-oriented error estimation We demonstrate two properties that give upper and lower bounds of the terms in the parallelogram identity. The properties are merely the application of the results on goal-oriented error estimation [12, 13, 20] into a substructured context.

Corollary 1. *Using notations of paragraph 2.1.5 and 3.4.3*

$$\begin{aligned} \beta^+_{inf} &\leqslant \left|\!\left|\!\left| \kappa \underline{e}_{discr} + \frac{1}{\kappa}\underline{\widetilde{e}}_{discr} \right|\!\right|\!\right|_\Omega^2 \leqslant \beta^+_{sup} \\ \beta^-_{inf} &\leqslant \left|\!\left|\!\left| \kappa \underline{e}_{discr} - \frac{1}{\kappa}\underline{\widetilde{e}}_{discr} \right|\!\right|\!\right|_\Omega^2 \leqslant \beta^-_{sup} \end{aligned} \tag{60}$$

with

$$\begin{cases} \beta^+_{sup} = 2\sqrt{\sum_s e_{CR_{\Omega^{(s)}}}(\underline{u}_D^{(s)}, \hat{\underline{\sigma}}_N^{(s)})^2}\sqrt{\sum_s e_{CR_{\Omega^{(s)}}}(\underline{\widetilde{u}}_D^{(s)}, \hat{\underline{\widetilde{\sigma}}}_N^{(s)})^2} \\ \qquad + 2\sum_s \int_{\Omega^{(s)}} (\hat{\underline{\widetilde{\sigma}}}_N^{(s)} - \mathbb{H}\underline{\varepsilon}\left(\underline{\widetilde{u}}_D^{(s)}\right)) : \mathbb{H}^{-1} : (\hat{\underline{\sigma}}_N^{(s)} - \mathbb{H} : \underline{\varepsilon}\left(\underline{u}_D^{(s)}\right))d\Omega \\ \beta^+_{inf} = \dfrac{(\kappa R_D(\underline{z}^+) + \frac{1}{\kappa}\widetilde{R}_D(\underline{z}^+))^2}{\sum_s |\!|\!|(\underline{z}^+)^{(s)}|\!|\!|^2_{\Omega^{(s)}}} \\ \qquad \textit{with } (\underline{z}^+)^{(s)} = \kappa \underline{w}^{(s)} + \dfrac{1}{\kappa}\underline{\widetilde{w}}^{(s)} \end{cases} \tag{61}$$

and

$$\begin{cases} \beta^-_{sup} = 2\sqrt{\sum_s e_{CR_{\Omega^{(s)}}}(\underline{u}_D^{(s)}, \hat{\underline{\sigma}}_N^{(s)})^2}\sqrt{\sum_s e_{CR_{\Omega^{(s)}}}(\underline{\widetilde{u}}_D^{(s)}, \hat{\underline{\widetilde{\sigma}}}_N^{(s)})^2} \\ \qquad - 2\sum_s \int_{\Omega^{(s)}} (\hat{\underline{\widetilde{\sigma}}}_N^{(s)} - \mathbb{H}\underline{\varepsilon}\left(\underline{\widetilde{u}}_D^{(s)}\right)) : \mathbb{H}^{-1} : (\hat{\underline{\sigma}}_N^{(s)} - \mathbb{H} : \underline{\varepsilon}\left(\underline{u}_D^{(s)}\right))d\Omega \\ \beta^-_{inf} = \dfrac{(\kappa R_D(\underline{z}^-) - \frac{1}{\kappa}\widetilde{R}_D(\underline{z}^-))^2}{\sum_s |\!|\!|(\underline{z}^-)^{(s)}|\!|\!|^2_{\Omega^{(s)}}} \\ \qquad \textit{with } (\underline{z}^-)^{(s)} = \kappa \underline{w}^{(s)} - \dfrac{1}{\kappa}\underline{\widetilde{w}}^{(s)} \end{cases} \tag{62}$$

One has to pay attention to the computation of the parameter κ which implies exchanges between subdomains. Indeed, this coefficient is defined by:

$$\kappa = \frac{\sqrt{\sum_s e_{CR_{\Omega^{(s)}}}(\underline{\widetilde{u}}_D^{(s)}, \hat{\underline{\widetilde{\sigma}}}_N^{(s)})}}{\sqrt{\sum_s e_{CR_{\Omega^{(s)}}}(\underline{u}_D^{(s)}, \hat{\underline{\sigma}}_N^{(s)})}} \tag{63}$$

Anyhow this exchange between subdomains is already done at the end of the parallel error estimation to obtain global measures.

Despite the possibility to separate contributions in terms $\sqrt{\sum_s e_{CR_{\Omega^{(s)}}}(\underline{\widetilde{u}}_D^{(s)}, \hat{\underline{\widetilde{\sigma}}}_N^{(s)})}$ and $\sqrt{\sum_s e_{CR_{\Omega^{(s)}}}(\underline{u}_D^{(s)}, \hat{\underline{\sigma}}_N^{(s)})}$, the full separation in the lower bounds β^-_{inf} and β^-_{inf} is a complex task since the parameter κ is the ratio of errors mixing algebraic and discretization sources.

However, the separation of sources for both global errors enables steering the iterative solver by an objective of precision of the quantity of interest (see [27]). The computation of β^-_{inf} and β^-_{inf} after convergence improves the bounding.

4 Numerical assessment

For all numerical examples, the behavior is linear, isotropic and elastic. The Young modulus is 1 Pa and the Poisson coefficient is 0.3.

4.1 Structure with exact solution

Let us consider a square linear elastic structure $\Omega = [-3l; 3l] \times [-3l; 3l]$ with homogeneous Dirichlet boundary conditions and plane strain hypothesis. The domain is subjected to a polynomial body force such that the exact solution is known:

$$\underline{u}_{ex} = (x + 3l)(x - 3l)(y + 3l)(y - 3l)\left((y - 3l)^2\underline{e}_x + (y + 3l)\underline{e}_y\right)$$

The mesh is made out of first order Lagrange triangles. As shown in Figure 1 the structure is decomposed into 9 regular subdomains.

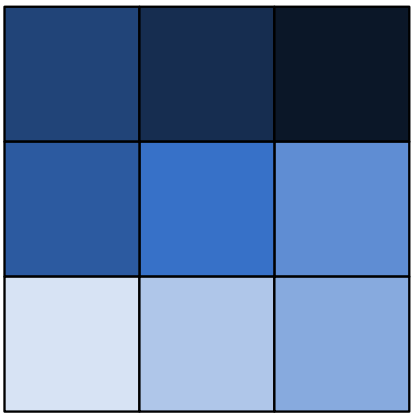


Figure 1: Substructuring

We solve the problem with a BDD solver (primal approach). We use the Flux-free technique [20] to build statically admissible stress fields; each star-patch problem is solved by subdividing each element into 12 elements (h-refinement technique). For the sake of simplicity, we note:

- $\rho = \frac{|L(\underline{w}) - a(\underline{u}_D, \underline{w})|}{\sqrt{\sum_s |||\underline{w}^{(s)}|||^2_{\Omega^{(s)}}}}$ the lower bound of the error
- $\rho_{discr} = \frac{|L(\underline{w}) - a(\underline{u}_N, \underline{w})|}{\sqrt{\sum_s |||\underline{w}^{(s)}|||^2_{\Omega^{(s)}}}}$ the discretization part of the lower bound
- $\rho_{alg} = \frac{|a(\underline{u}_D - \underline{u}_N, \underline{w})|}{\sqrt{\sum_s |||\underline{w}^{(s)}|||^2_{\Omega^{(s)}}}}$ the algebraic part of the lower bound
- $\rho_{bis} = \rho_{discr} - \alpha$ the lower bound with separation of sources of error
- $\theta = \sqrt{\sum_s e^2_{CR_{\Omega^{(s)}}}(\underline{u}_D^{(s)}, \hat{\underline{\underline{\sigma}}}_N^{(s)})}$ the upper bound of the error
- $\theta_{discr} = \sqrt{\sum_s e^2_{CR_{\Omega^{(s)}}}(\underline{u}_N^{(s)}, \hat{\underline{\underline{\sigma}}}_N^{(s)})}$ the discretization part of the upper bound

The quantities with the superscript ^{seq} are computed with a sequential simulation (no substructuring and use of a direct solver).

4.1.1 Effects of the substructuring on the computation of the lower bound at convergence

In this subsection, we compare the lower bound obtained in a sequential simulation with the lower bound obtained in the substructured context when the solver has converged. Since the study is done at convergence, the primal and dual approaches are equivalent.

On figure 2, we observe that the bounds are the same for sequential and substructured computations. We also verify that the exact error is between upper and lower bounds. The convergence is the one expected for such a regular problem (h-slope).

Then, we compare the bounds for several substructuring as illustrated in figure 3. Table 1 gathers the lower bounds for sequential, primal and dual approaches computed at convergence normalized by the bounds for sequential computation. We observe that the substructuring has quasi no influence on the accuracy of the lower bound.

4.1.2 Separation of sources of error

In this subsection, we illustrate the separation of sources of error in the lower bound.

On the first graph in figure 4, we give the evolution of the upper and lower bounds and of the true error until the fifth iteration. We observe the fast convergence of those bounds. On the same graph, we also visualize the

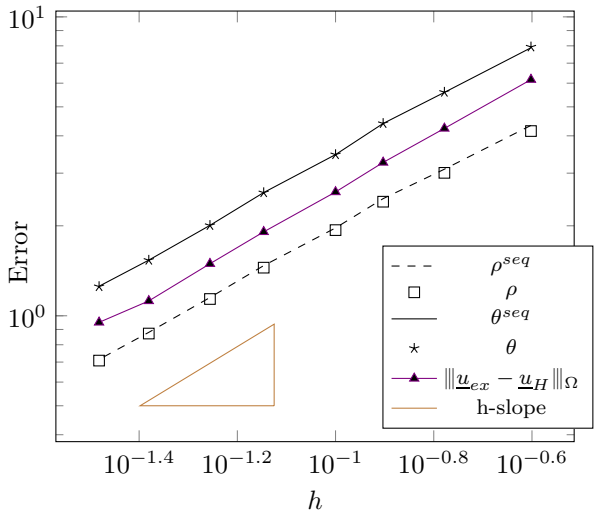


Figure 2: Evolution of the upper and lower bounds of the error in function of the mesh size h

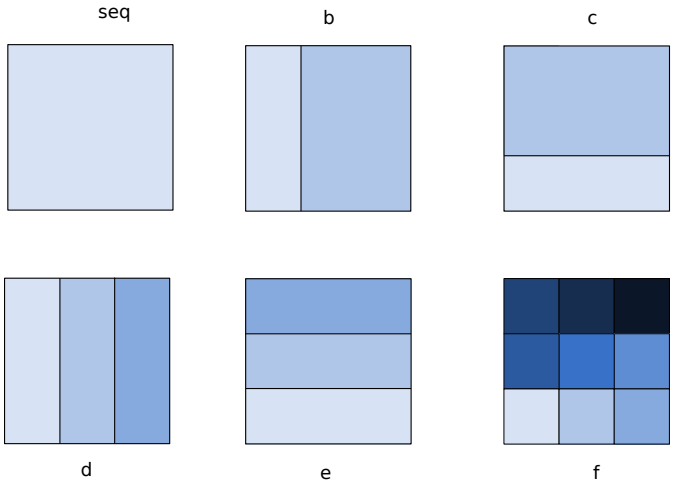


Figure 3: Different substructurings

discretization parts θ_{discr} and ρ_{discr}. The bounding is more precise and enables to define after a few iterations the interval in which the true error is located at convergence. The second graph in figure 4 gives the evolution of the terms in theorem 2. The quantity α scrictly decreases along the iterations, the evolution of the quantity ρ_{alg} is not as smooth.

Figure 5 represents the two lower bounds of theorem 2. As expected, the second lower bound is not precise at the beginning since it gives a negative value. However, the zero of this bound enables to tell the moment when the algebraic error becomes smaller than the discretization error.

4.2 Pre-cracked structure

We now consider a pre-cracked structure decomposed into 16 subdomains as illustrated in Figure 6. The displacements at the base of the structure and on the larger hole are imposed to be zero. The upper-left part and the second hole are subjected to a constant unit pressure. We made the hypothesis of plane stress. The quantity of interest is the mean of the stress component σ_{xx} on a region ω close to the crack. In Figure 6, the loading of the reference problem is in blue and the loading of adjoint problem is in orange. We used the FETI algorithm (dual approach) to solve the interface problem and the statically admissible stress fields are built using the Flux-free technique [3, 20] with h-refinement technique for the solving of local problems on star-patches (each element is divided into 16 elements).

4.2.1 Separation of sources of error in upper and lower bounds

For this paragraph, we consider only the forward problem. The mesh used for this computation is composed of 4370 degrees of freedom. We compute the global upper and lower bounds during the iterations and the discretization upper and lower bounds along the iterations on the same graph in figure 7.

	seq	b	c	d	e	f
$\frac{\theta}{\theta^{seq}}$	1	1.0002	0.9993	0.9989	1.0004	0.9993
$\frac{\rho}{\rho^{seq}}$	1	0.9981	0.9964	0.9926	0.9958	0.9886

Table 1: Relative upper and lower bounds of the error for various substructuring

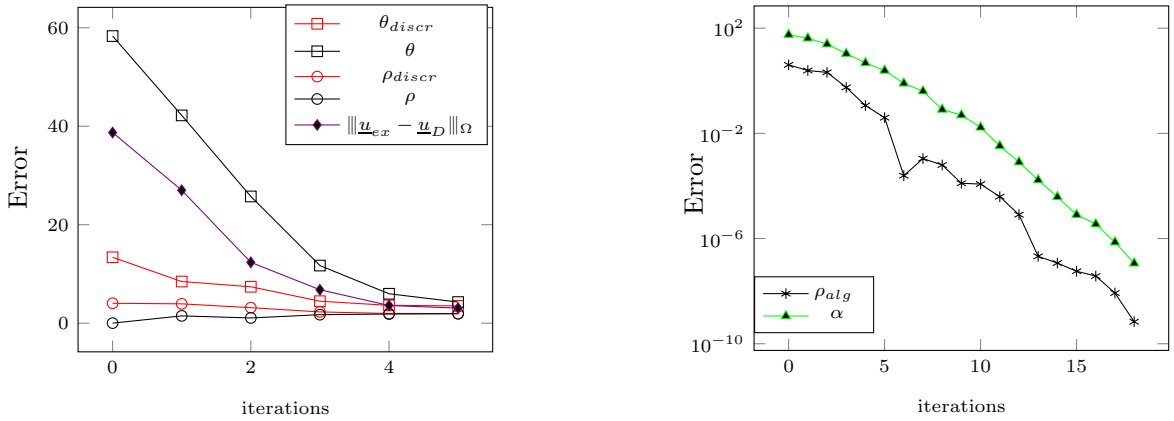


Figure 4: Separation of sources of error in upper and lower bounds

As for the previous example, we observe the fast convergence of the upper and lower bounds. This graph also illustrates the separation of sources of error. Finally, we observe that the residual α decreases with the iterations and that, again, the evolution of ρ_{alg} is not as regular.

In figure 8, we give the discretization upper and lower bounds, which define a discretization envelope, for all iterations on the first graph and until the seventh iteration on the second graph.

The discretization envelope can be used to define the stopping criterion of the solver. In [28], the proposed criterion was to stop when the algebraic error was ten times smaller than the discretization error. On this example, the solver would stop at the sixth iteration. A new stopping criterion could be to stop when the algebraic error is smaller than the discretization part of the lower bound, which would lead to stop to at the fourth iteration.

4.3 Goal-oriented error estimation

In this subsection, we propose an auto-adaptive strategy to steer the iterative solver by an objective of precision on a quantity of interest. In this example, the objective of precision will be five percent. We used the FETI algorithm (dual approach) to solve the interface problem and the statically admissible stress fields are built using the Flux-free technique [20] with h-refinement technique for the solving of local problems on star-patches (each element is divided into 4 elements). Since the forward and adjoint problems are auto-adjoint (due to the symetry of the bilinear form), we solve the two problems simultaneously using a block conjugate gradient (see [27] for more details). The separation of sources of error in the lower bounds of global error on forward and adjoint problem enables the definition of the following criterion : **STOP when** $\alpha < \rho_{discr}$ **and** $\widetilde{\alpha} < \widetilde{\rho}_{discr}$ which expresses the fact that the residual is out of the discretization enveloppe so that the algebraic error is negligible in comparaison with the discretization error. Using equation (58), we have the follwing upper and lower bounds on the unknown exact value of the quantity of interest I_{ex}:

$$I_{ex}^- = I_H + \frac{1}{4}\beta_{inf}^+ - \frac{1}{4}\beta_{sup}^- \leqslant I_{ex} \leqslant I_H + \frac{1}{4}\beta_{sup}^+ - \frac{1}{4}\beta_{inf}^- = I_{ex}^+ \tag{64}$$

We will compare the bounds I_{ex}^- and I_{ex}^+ in case the quantities β_{inf}^+ and β_{inf}^- are computed using expressions in Corollary 1 and in case they are chosen equal to zero ($\beta_{inf}^+ = \beta_{inf}^- = 0$), which is equivalent to the bounding in equation (53) with $\underline{u}_H = \underline{u}_D$, $\widetilde{\underline{u}}_{\widetilde{H}} = \widetilde{\underline{u}}_D$, $\hat{\underline{\sigma}}_H = \hat{\underline{\sigma}}_N$ and $\hat{\widetilde{\underline{\sigma}}}_{\widetilde{H}} = \hat{\widetilde{\underline{\sigma}}}_N$.

We start with a first mesh which is a little bit refined near the quantity of interest in order to have several elements in the region ω. The discretization error is computed at iteration 1 (which is more relevant than the initilization Iteration 0). The criterion is defined and the solver iterates until the criterion is reached. The error is estimated once again to verify that the discretization error has not changed too much. We give in table 2 the evolution of the residual for forward and adjoint problems and the bounds on the global errors for the two problems.

Regarding the quantity of interest, at the sixth iteration, we obtain the values presented in table 3.

In table 4, we give the upper and lower bounds of the unknown exact value I_{ex} of the quantity of interest with and without the use of the lower bounds. We observe that the use of the lower bounds enables to reduce the width by 44 %. At the end of the first solving, the error on the quantity of interest is 22.224 %.

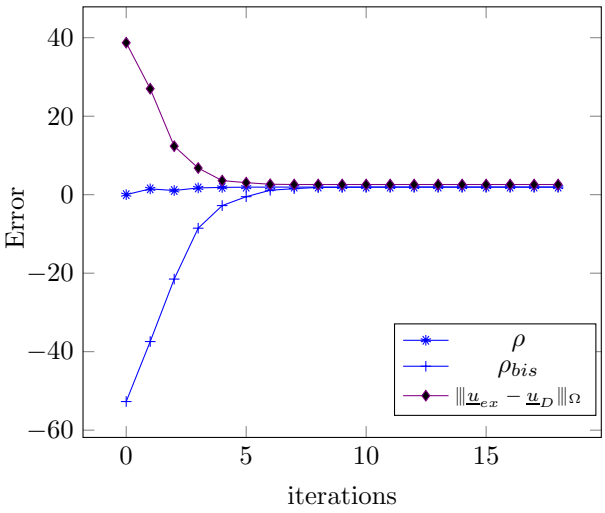


Figure 5: Evoluation of the two lower bounds during iterations

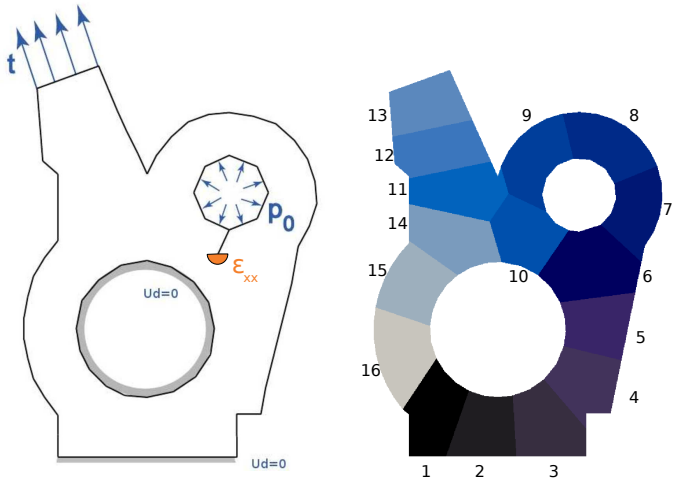


Figure 6: Loading of forward (blue) and adjoint problems (orange), domain decomposition

We also give, for both problems, the contribution from each subdomain thanks to the following quantities :

$$\eta^{(s)} = \frac{e_{CR_{\Omega^{(s)}}}(\underline{u}_N^{(s)}, \hat{\underline{\underline{\sigma}}}_N^{(s)})}{\sqrt{\sum_s e^2_{CR_{\Omega^{(s)}}}(\underline{u}_N^{(s)}, \hat{\underline{\underline{\sigma}}}_N^{(s)})}} \quad \text{and} \quad \tilde{\eta}^{(s)} = \frac{e_{CR_{\Omega^{(s)}}}(\tilde{\underline{u}}_N^{(s)}, \hat{\tilde{\underline{\underline{\sigma}}}}_N^{(s)})}{\sqrt{\sum_s e^2_{CR_{\Omega^{(s)}}}(\tilde{\underline{u}}_N^{(s)}, \hat{\tilde{\underline{\underline{\sigma}}}}_N^{(s)})}} \tag{65}$$

and plot the contributions on the Figure 9.

As expected, for the adjoint problem, the error is mainly located in the sixth subdomain, which is the subdomain with the load. For the forward problem, the error is more diffuse.

In order to reach the objective of precision, we decide to first improve the quality of the solution of the forward problem. To do so, since the error is not located in few subdomains, we decide to refine the mesh on the whole structure. Of course, we could have used the error map provided by the error estimator and a remeshing criterion to process adaptive remeshing (see for instance [33, 5, 17, 2, 6]). For sake of simplicity, a refinement by splitting is performed. We also reuse the search directions computed during the first solving in order to speed up the next one. The 12 interface vectors (2 interface vectors -one for the forward problem, one for the adjoint problem- computed at each iteration of the first solving that converged in 6 iterations) corresponding to the search directions are projected on the new mesh and used as additional constraints thanks to augmented-Krylov methods [31]. Since the dual approach was chosen, we use a two-level FETI algorithm [8] to take into account the additional constraints. It does not modify the methodology to construct admissible fields nor the error estimator. Recycling search directions leading to a better initilization, we compute the error estimation at iteration 0 and define the stopping criterion. Once the criterion is reached, the error is estimated once again and if the criterion is checked, the solver is stopped. We give in the table 5 the evolution of the residuals and of the bounds on the global errors for the forward and adjoint problems. We can observe that the first residual is comparable to the last residual of the first solving.

Regarding the quantity of interest, we obtain the values presented in table 6.

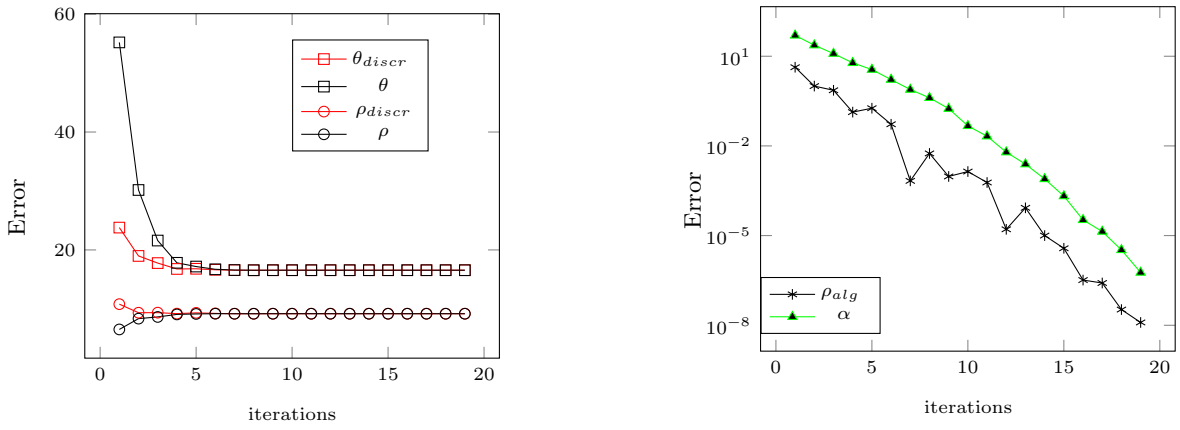


Figure 7: Pre-cracked structure : Evolution of the upper and lower bounds (global, discretization part, algebraic part) and of the residual during the iterations

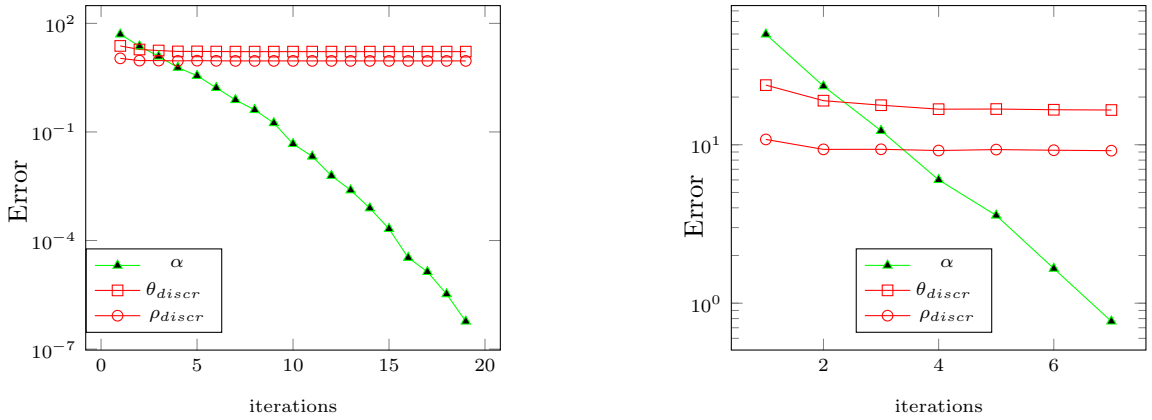


Figure 8: Pre-cracked structure : Discretization envelope and residual against iterations

In table 7, we give the upper and lower bounds of the unknown exact value I_{ex} of the quantity of interest with and without the use of the lower bounds. We observe that the use of the lower bounds improves the bounding. At the end of the second solving, the uncertainty on the quantity of interest is 6.7893 %.

In order to reach the objective of precision, we decide to refine the mesh only for the sixth subdomain to improve the quality of the adjoint solution. Refining this subdomain's discretization introduces incompatibilities at the interface that can be easily managed thanks to transfer matrix as explained in [27]. This incompatibility does not affect the error estimator since the quantity of interest is located far from the subdomain's boundary. Once again, we reuse the search directions of the first two solutions to speed up the third solving. As the discretization of the interface is not modified (see [27]), the interface vectors can be directly used as additional constraints.

Regarding the quantity of interest, we obtain the values presented in table 9.

In table 10, we give the upper and lower bounds of the unknown exact value I_{ex} of the quantity of interest with and without the use of the lower bounds. We observe that the use of the lower bounds enables to reduce the width by 44 %. At the end of the third solving, the objective of precision is reached and the error on the exact value of the quantity of interest is smaller than 4 %.

Finally, we give the global errors and residuals against cumulative iteration on Figure 10 and the evolution of the approximated value of the quantity of interest I_H and the upper and lower bounds for I_{ex} in Figure 11.

5 Conclusion

In this paper, we proposed a strict lower bound of the error in a substructured context which can be computed in parallel using the admissible fields built for the computation of the upper bound. Moreover, a theorem gives a second lower bound that separates the algebraic error from the discretization error. As illustrated on mechanical examples, the lower bounds are quasi independent from the substructuring and are as accurate as the sequential lower bound. The examples also show the separation of sources of error for the lower bound. Finally, we proposed an auto-adaptive strategy to steer the iterative solver by an objective of precision on a quantity of interest. Benefiting from the separation of sources of error to avoid oversolving and the recycling of Krylov subspaces, the strategy automatically defines a sequence of optimized solvings.

Iteration	θ_{discr}	ρ_{discr}	$\widetilde{\theta}_{discr}$	$\widetilde{\rho}_{discr}$	α	$\widetilde{\alpha}$
0					260.07	0.26041
1	11.45	7.0008	0.12867	$1.5457\ 10^{-3}$	48.339	0.14931
2					35.991	$8.5664\ 10^{-2}$
3					16.649	$4.8205\ 10^{-2}$
4					5.5966	$1.2462\ 10^{-2}$
5					3.5765	$8.4996\ 10^{-3}$
6	9.9004	6.7105	0.12682	$1.5893\ 10^{-3}$	0.94805	$1.5156\ 10^{-3}$

Table 2: Pre-cracked structure: First mesh

I_H	β^-_{inf}	β^+_{inf}	$\frac{1}{4}\beta^+_{sup}$	$\frac{1}{4}\beta^-_{sup}$
3.1505	1.0087	1.2129	0.68424	0.57132

Table 3: Pre-cracked structure: First mesh : error estimation

Acknoledgment

The authour would like to thank Professor Pedro Diez (Universitat Politècnica de Catalunya) for the helpful and fruitful discussions.

References

[1] R. Becker and R. Rannacher. A feed-back approach to error control in finite element methods: Basic analysis and examples. *Journal of Numerical Mathematics*, 4:237–264, 1996.

[2] Emmanuel Bellenger and Patrice Coorevits. Adaptive mesh refinement for the control of cost and quality in finite element analysis. *Finite Elements in Analysis and Design*, 41(15):1413 – 1440, 2005.

[3] R. Cottereau, P. Díez, and A. Huerta. Strict error bounds for linear solid mechanics problems using a subdomain-based flux-free method. *Computational Mechanics*, 44(4):533–547, 2009.

[4] P. Díez, N. Parés, and A. Huerta. Recovering lower bounds of the error by postprocessing implicit residual a posteriori error estimates. *International Journal for Numerical Methods in Engineering*, 56(10):1465–1488, 2003.

[5] Pedro Díez and Antonio Huerta. A unified approach to remeshing strategies for finite element h-adaptivity. *Computer Methods in Applied Mechanics and Engineering*, 176(1-4):215–229, 1999.

[6] Pedro Díez, Juan José Ródenas, and Olgierd C. Zienkiewicz. Equilibrated patch recovery error estimates: simple and accurate upper bounds of the error. *International Journal for Numerical Methods in Engineering*, 69(10):2075–2098, 2007.

[7] C. Farhat and F. X. Roux. The dual schur complement method with well-posed local neumann problems. *Contemporary Mathematics*, 157:193–201, 1994.

[8] Charbel Farhat and Jan Mandel. The two-level FETI method for static and dynamic plate problems part I: An optimal iterative solver for biharmonic systems. *Computer Methods in Applied Mechanics and Engineering*, 155(12):129 – 151, 1998.

[9] L. Gallimard. A constitutive relation error estimator based on traction-free recovery of the equilibrated stress. *International Journal for Numerical Methods in Engineering*, 78(4):460–482, 2009.

[10] P. Gosselet, C. Rey, and J. Pebrel. Total and selective reuse of krylov subspaces for the resolution of sequences of nonlinear structural problems. *International Journal for Numerical Methods in Engineering*, 94(1):60–83, 2013.

[11] Pierre Gosselet and Christian Rey. Non-overlapping domain decomposition methods in structural mechanics. *Archives of Computational Methods in Engineering*, 13(4):515–572, 2006.

[12] P. Ladevèze. Upper error bounds on calculated outputs of interest for linear and nonlinear structural problems. *Comptes Rendus Académie des Sciences - Mécanique, Paris*, 334(7):399–407, 2006.

[13] P. Ladevèze. Strict upper error bounds on computed outputs of interest in computational structural mechanics. *Computational Mechanics*, 42(2):271–286, 2008.

[14] P. Ladevèze and D. Leguillon. Error estimate procedure in the finite element method and application. *SIAM Journal of Numerical Analysis*, 20(3):485–509, 1983.

	I_{ex}^{+}	I_{ex}^{-}	width	precision
With $\beta_{inf}^{+} = \beta_{inf}^{-} = 0$	3.8347	2.5791	1.2555	39.852 %
With β_{inf}^{+} and β_{inf}^{-} from table 3	3.5825	2.8824	0.7001	22.224 %

Table 4: Pre-cracked structure: First mesh : bounds for the exact quantity of interest

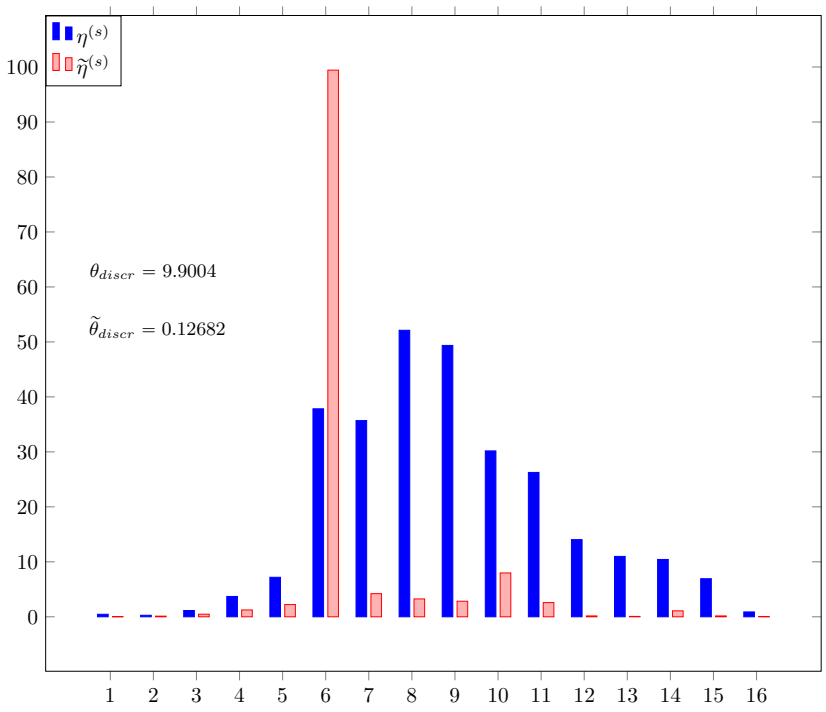


Figure 9: First mesh : Distribution of the error estimator within subdomains

[15] P. Ladevèze and J. P. Pelle. *Mastering Calculations in Linear and Nonlinear Mechanics*. Springer, New York, 2004.

[16] Patrick Le Tallec. Domain decomposition methods in computational mechanics. *Comput. Mech. Adv.*, 1(2):121–220, 1994.

[17] J. T. Oden and S. Prudhomme. Goal-oriented error estimation and adaptivity for the finite element method. *Computers & Mathematics with Applications*, 41(5-6):735–756, 2001.

[18] S. Ohnimus, E. Stein, and E. Walhorn. Local error estimates of fem for displacements and stresses in linear elasticity by solving local neumann problems. *International Journal for Numerical Methods in Engineering*, 52(7):727–746, 2001.

[19] M. Paraschivoiu, J. Peraire, and A. T. Patera. A posteriori finite element bounds for linear-functional outputs of elliptic partial differential equations. *Computer Methods in Applied Mechanics and Engineering*, 150(1-4):289–312, 1997.

[20] N. Parés, P. Díez, and A. Huerta. Subdomain-based flux-free a posteriori error estimators. *Computer Methods in Applied Mechanics and Engineering*, 195(4-6):297–323, 2006.

[21] N. Parés, H. Santos, and P. Díez. Guaranteed energy error bounds for the poisson equation using a flux-free approach: Solving the local problems in subdomains. *International Journal for Numerical Methods in Engineering*, 79(10):1203–1244, 2009.

[22] A. Parret-Fréaud, C. Rey, P. Gosselet, and F. Feyel. Fast estimation of discretization error for fe problems solved by domain decomposition. *Computer Methods in Applied Mechanics and Engineering*, 199(49-52):3315–3323, 2010.

[23] F. Pled, L. Chamoin, and P. Ladevèze. On the techniques for constructing admissible stress fields in model verification: Performances on engineering examples. *International Journal for Numerical Methods in Engineering*, 88(5):409–441, 2011.

[24] S. Prudhomme and J. T. Oden. On goal-oriented error estimation for elliptic problems: application to the control of pointwise errors. *Computer Methods in Applied Mechanics and Engineering*, 176(1-4):313–331, 1999.

Iteration	θ_{discr}	ρ_{discr}	$\widetilde{\theta}_{discr}$	$\widetilde{\rho}_{discr}$	α	$\widetilde{\alpha}$
0	5.7611	3.8882	$7.5633\ 10^{-2}$	$7.551\ 10^{-4}$	5.7017	$7.4954\ 10^{-3}$
1					3.3265	$4.4442\ 10^{-3}$
2					1.2869	$8.4996\ 10^{-4}$
3	5.4304	3.7882	$7.5564\ 10^{-2}$	$7.6332\ 10^{-4}$	0.31197	$2.4632\ 10^{-4}$

Table 5: Pre-cracked structure: Second mesh

I_H	β^-_{inf}	β^+_{inf}	$\frac{1}{4}\beta^+_{sup}$	$\frac{1}{4}\beta^-_{sup}$
3.2266	0.32673	0.43838	0.23661	0.17372

Table 6: Pre-cracked structure: Second mesh : error estimation

[25] Christian Rey and Franck Risler. A rayleigh–ritz preconditioner for the iterative solution to large scale nonlinear problems. *Numerical Algorithms*, 17(3/4):279–311, 1998.

[26] V. Rey, P. Gosselet, and C. Rey. Study of the strong prolongation equation for the construction of statically admissible stress fields: Implementation and optimization. *Computer Methods in Applied Mechanics and Engineering*, 268(0):82 – 104, 2014.

[27] Valentine Rey, Pierre Gosselet, and Christian Rey. Strict bounding of quantities of interest in computations based on domain decomposition. *Computer Methods in Applied Mechanics and Engineering*, 287(0):212 – 228, 2015.

[28] Valentine Rey, Christian Rey, and Pierre Gosselet. A strict error bound with separated contributions of the discretization and of the iterative solver in non-overlapping domain decomposition methods. *Computer Methods in Applied Mechanics and Engineering*, 270(0):293 – 303, 2014.

[29] F. Risler and C. Rey. Iterative accelerating algorithms with krylov subspaces for the solution to large-scale non-linear problems. *Numerical algorithms*, 23:1–30, 2000.

[30] Y. Saad. *Iterative methods for sparse linear systems*. Society for Industrial and Applied Mathematics, Philadelphia, 3rd edition edition, 2003.

[31] Yousef Saad. Analysis of augmented krylov subspace methods. *SIAM Journal on Matrix Analysis and Applications*, 18(2):435–449, 1997.

[32] T. Strouboulis, I. Babuška, D. K. Datta, K. Copps, and S. K. Gangaraj. A posteriori estimation and adaptive control of the error in the quantity of interest. part I: A posteriori estimation of the error in the von mises stress and the stress intensity factor. *Computer Methods in Applied Mechanics and Engineering*, 181(1–3):261–294, 2000.

[33] R. Verfürth. *A Review of A Posteriori Error Estimation and Adaptive Mesh-refinement Techniques*. Wiley-Teubner, Stuttgart, 1996.

	I_{ex}^{+}	I_{ex}^{-}	width	precision
With $\beta_{inf}^{+} = \beta_{inf}^{-} = 0$	3.4632	3.0528	0.41034	12.717 %
With β_{inf}^{+} and β_{inf}^{-} from table 6	3.3815	3.1624	0.21906	6.7893 %

Table 7: Pre-cracked structure: Second mesh : bounds for the exact quantity of interest

Iteration	θ_{discr}	ρ_{discr}	$\widetilde{\theta}_{discr}$	$\widetilde{\rho}_{discr}$	α	$\widetilde{\alpha}$
0	5.1501	3.5841	$4.6197\ 10^{-2}$	$2.9485\ 10^{-4}$	0.32764	$6.2878\ 10^{-4}$
1	5.1267	3.5824	$4.619\ 10^{-2}$	$2.9482\ 10^{-4}$	0.10619	$1.6588\ 10^{-4}$

Table 8: Pre-cracked structure: Third mesh

I_H	β_{inf}^{-}	β_{inf}^{+}	$\frac{1}{4}\beta_{sup}^{+}$	$\frac{1}{4}\beta_{sup}^{-}$
3.2625	0.18986	0.25873	0.13618	0.10058

Table 9: Pre-cracked structure: Third mesh : error estimation

	I_{ex}^{+}	I_{ex}^{-}	width	precision
With $\beta_{inf}^{+} = \beta_{inf}^{-} = 0$	3.3986	3.1619	0.2367	7.2574 %
With β_{inf}^{+} and β_{inf}^{-} from table 9	3.3512	3.2265	0.1247	3.8199%

Table 10: Pre-cracked structure: Third mesh : bounds for the exact quantity of interest

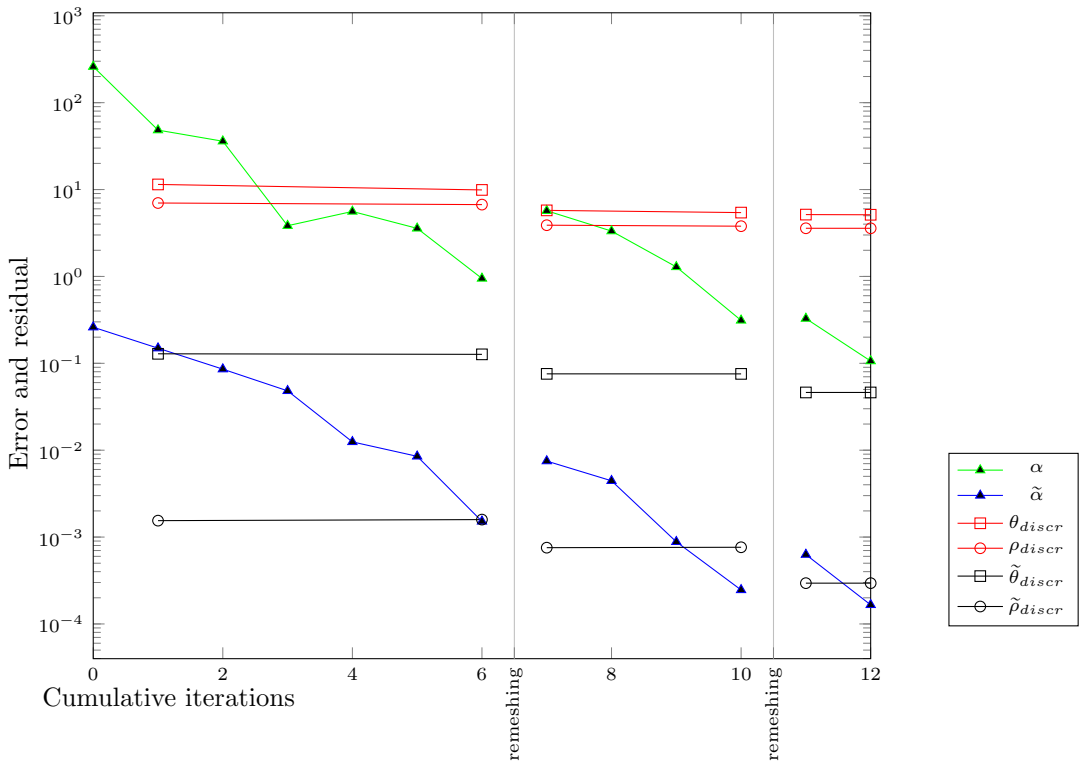


Figure 10: Pre-cracked structure : Discretization envelope and residual against cumulative iterations

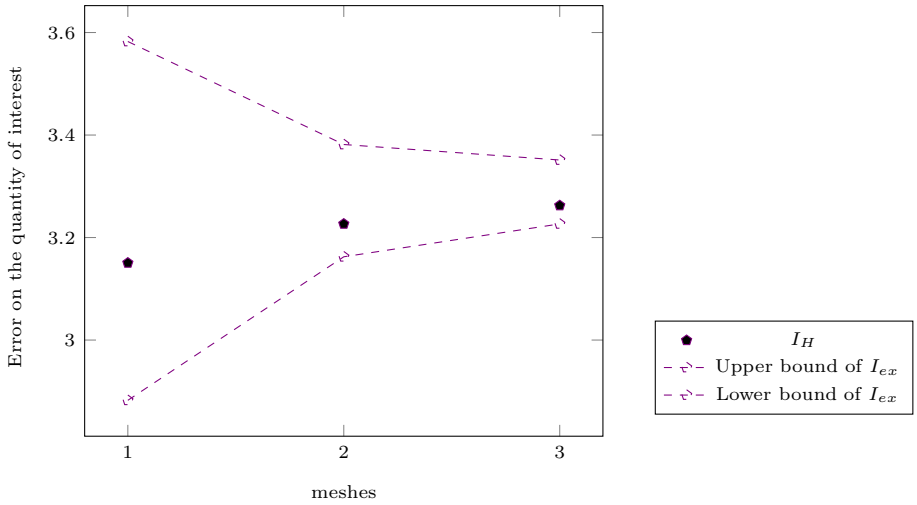


Figure 11: Pre-cracked structure : Evolution of the approximated quantity of interest I_H and of the bounds of the exact value I_{ex}

H Brouillon d'un article sur la méthode non-intrusive

Ce document est l'ébauche d'un article récapitulatif sur le couplage non-intrusif. Outre un intérêt didactique car les algorithmes sont présentés de manière très proche de la mise en œuvre avec des interprétations mécaniques simple, on inscrit la méthode au sein des algorithmes de Schwarz optimisés ce qui permet de bénéficier d'un cadre théorique très développé. On montre en outre comment appliqué des techniques de gradient conjugué en linéaire et non-linéaire, et une façon originale de traiter les maillages complètement non-conformes.

Sur le même thème, le lecteur pourra consulter [Gendre *et al.*, 2009] où la méthode a été présentée pour la première fois et [Gendre *et al.*, 2011] où l'aspect mixte de la méthode était exploité sans être réellement formalisé.

Noname manuscript No.
(will be inserted by the editor)

Non-invasive global-local coupling as a Schwarz domain decomposition method: acceleration and generalization

Pierre Gosselet · Maxime Blanchard · Olivier Allix · Guillaume Guguin

Received: date / Accepted: date

Abstract Keywords non-invasive coupling · Schwarz domain decomposition · non-conforming patch · Krylov acceleration.

PACS 46.70.De · 02.60.Lj

Mathematics Subject Classification (2010) 65N55 · 65N85

1 Introduction

2 Derivation of the non-invasive algorithm

The algorithm we study is very general and applies to the study of many PDEs; in order to fix the ideas, we consider nonlinear quasi-static structure mechanics problem under the small strain hypothesis. We note u the displacement field, ε the symmetric part of the gradient, σ Cauchy's stress tensor. For a domain Ω submitted to given body force f, Dirichlet conditions u_d is imposed on the part $\partial_d\Omega$ of the boundary and Neumann condition g is imposed on the complement part $\partial_n\Omega$. In order to manage viscous materials, the study is conducted over a time interval $\mathcal{T} = [0, T]$, and the following equations are meant to be satisfied at any time $t \in \mathcal{T}$, which we omit to write except when necessary.

Let $V(\Omega) = \left\{v \in H^1(\Omega), v = u_d \text{ on } \partial_d\Omega\right\}$ be the affine space of admissible displacement and $V^0(\Omega)$ the associated vector space. The conservation of momentum

This work is supported by project Icare of French National Research Agency, ANR-12-MONU-0002-04

Pierre Gosselet, Maxime Blanchard, Olivier Allix, Guillaume Guguin
LMT-Cachan, ENS-Cachan/CNRS/Université Paris Saclay,
61 avenue du Président Wilson, 94235 Cachan, France
Tel.: +33-1-47402735
Fax: +33-1-47402785
E-mail: gosselet@lmt.ens-cachan.fr, blanchard@ens-cachan.fr,allix@lmt.ens-cachan.fr, guguin@lmt.ens-cachan.fr

can be written as:

$$\int_\Omega \sigma(u) : \varepsilon(v)\, dx = \int_\Omega f \cdot v\, dx + \int_{\partial_n \Omega} g \cdot v\, dS,\ \forall v \in V^0(\Omega) \tag{1}$$

The notation $\sigma(u)$ stands for the following kind of dependence:

$$\sigma(x,t) \in B(\varepsilon(u(x,\tau)), \tau < t),\ x \in \Omega,\ t \in \mathcal{T} \tag{2}$$

This modeling of the mechanical behavior is typically suited for elastoviscoplastic materials, it is often complemented with the introduction of internal variables to characterize the dependence with respect to the history.

The mechanical problem above takes the following classical form:

$$\text{Find } u \in V(\Omega) \ /\ a(u,v) = l(v), \forall v \in V^0(\Omega) \tag{3}$$

where l is a continuous linear form, and a is continuous coercive linear in the second variable.

In the following we handle several space subdomains and models, when any quantity will be specifically attached to one model, a superscript will mention it.

The Reference problem (superscript ^{R}) is set on the domain Ω^R which is the assembly of two non-overlapping subdomains: the zone of interest where a Fine model is required for a reliable simulation (superscript ^{F}), and a Complement zone (superscript ^{C}) where a simpler model is sufficient (and which in general covers most of the structure). The interface is $\Gamma = \partial\Omega^C \cap \partial\Omega^F$, it is thus immersed in Ω^R. Note that using several zones of interest presents no difficulty as long as they do not overlap; note also that the Reference problem is never formed in practice.

$$\textbf{Reference problem} \begin{cases} \text{Find } u \in V(\Omega^R),\ \text{such that } \forall v \in V^0(\Omega^R), \\ a^R(u,v) := a^C(u,v) + a^F(u,v) = l^C(v) + l^F(v) =: l^R(v) \end{cases} \tag{4}$$

We assume that we have another representation of the zone of interest, named Auxiliary representation (superscript ^{A}) which shares the same characteristics as the Complement zone, and which is thus coarser than the Fine representation. Typically if Ω^F is a zone where material coefficients have strong variations, the Fine representation would follow the exact distribution whereas the Auxiliary representation could consist in using an homogeneized behavior; load could also be simplified. A classical application is the case where the Fine model is stochastic whereas the Auxiliary model is deterministic Chevreuil et al. (2013). We insert the Auxiliary representation of the zone of interest in the Reference problem:

$$\begin{aligned} &\text{Find } u \in V(\Omega^R), \text{ such that } \forall v \in V^0(\Omega^R), \\ &a^C(u,v) + a^F(u,v) = l^C(v) + l^F(v) \\ &\underbrace{a^C(u,v) + a^A(u,v)}_{a^G(u,v)} = \underbrace{l^C(v) + l^A(v)}_{l^G(v)} + \Big(a^A(u,v) - l^A(v)\Big) - \Big(a^F(u,v) - l^F(v)\Big) \end{aligned} \tag{5}$$

The Global problem, (supersript ^{G}), is the assembly of the Complement zone with the Auxiliary (coarse) representation of the zone of interest, this problem is in practice assembled and dealt with by legacy software.

From previous equation, we could derive the following stationary iteration:

$$a^G(u_{n+1}, v) = l^G(v) + \Big(a^A(u_n, v) - l^A(v)\Big) - \Big(a^F(u_n, v) - l^F(v)\Big) \tag{6}$$

which would correspond to a fixed point of the Reference problem preconditioned by the coarse Global system. Not only convergence would be slow but also the right-hand-side terms would not be easy to compute in practice. Moreover, this iteration needs the Auxiliary domain Ω^A to be coincident with the Fine domain Ω^F which is a limitation we want to get rid of. In the following, the only assume that the interface is on the boundary of the Auxiliary domain $\Gamma \subset \partial\Omega^A$, so that Γ is on the boundary of all subdomains.

We thus choose to associate the right hand side of (5) with the evaluation of local problems. Starting from $p_0 = 0$ (the mathematical space for p_n is discussed later), the basic global/local iteration is then the following:

$$\textbf{Global problem} \begin{cases} \text{Find } u_n^G \in V(\Omega^G), \text{ such that } \forall v \in V^0(\Omega^G), \\ a^G(u_n^G, v) = l^G(v) + \langle p_n, v\rangle \end{cases} \tag{7a}$$

$$\textbf{Fine problem} \begin{cases} \text{Find } (u_n^F, \lambda_n^F) \in V(\Omega^F) \times H^{-1/2}(\Gamma), \\ \text{ such that } \forall (v, \mu) \in V^0(\Omega^F) \times H^{-1/2}(\Gamma), \\ \begin{cases} a^F(u_n^F, v) = l^F(v) + \langle \lambda_n^F, v\rangle_\Gamma \\ \langle \mu, u_n^F - u_n^G\rangle_\Gamma = 0 \end{cases} \end{cases} \tag{7b}$$

$$\textbf{Auxiliary problem} \begin{cases} \text{Find } (u_n^A, \lambda_n^A) \in V(\Omega^A) \times H^{-1/2}(\Gamma), \\ \text{ such that } \forall (v, \mu) \in V^0(\Omega^A) \times H^{-1/2}(\Gamma), \\ \begin{cases} a^A(u_n^A, v) = l^A(v) + \langle \lambda_n^A, v\rangle_\Gamma \\ \langle \mu, u_n^A - u_n^G\rangle_\Gamma = 0 \end{cases} \end{cases} \tag{7c}$$

$$\textbf{Update} \begin{cases} \langle p_{n+1}, v\rangle = \Big(a^A(u_n^A, v^A) - l^A(v^A)\Big) - \Big(a^F(u_n^F, v^F) - l^F(v^F)\Big) \\ \text{with } v^A_{|\Gamma} = v^F_{|\Gamma} = v \end{cases} \tag{7d}$$

In words, the global problem is the coarse problem with extra load p, the fine and auxiliary systems are resolutions on the domain of interest with imposed Dirichlet conditions on Γ. We chose a Lagrangian formulation for these problems in order to make appear the reaction forces λ^F and λ^A. The update is simply the equivalent of (6) with fields issuing from the local solves instead of the global one.

Remark 1 Of course, the Lagrange multipliers are equal to the normal stress:

$$\lambda^X = \sigma^X \cdot n^X \tag{8}$$

where $X \in \{C, A, F\}$ and n^X is the outer normal vector.

We have the following properties:

– Assuming the fine and auxiliary problems were solved exactly, we have:

$$p_{n+1} = \Big(\lambda_n^A - \lambda_n^F\Big) \in H^{-1/2}(\Gamma) \tag{9}$$

the corrective load p is then an immersed surface traction. In the following, we always assume the exactness of the resolution; note that using inexact solvers was investigated in Passieux et al. (2013) where the method is identified with a localized multigrid iteration.

– Because the auxiliary problem corresponds to the restriction of the global problem on the zone of interest with global displacement imposed, we directly have:

$$u_n^A = u_{n|\Omega^A}^G \tag{10}$$

The introduction of the auxiliary problem is thus not mandatory, it is just a workaround in case of software unable to compute the reaction in an immersed surface. Of course, the auxiliary problem can be solved in parallel with the fine problem.

– We can also define the reaction from the complement zone:

$$a^C(u_n^G, v) = l^C(v) + \langle \lambda_n^C, v \rangle_\Gamma, \ \forall v \in V(\Omega^C) \tag{11}$$

Then we see that:

$$\lambda_n^C + \lambda_n^A = p_n \tag{12}$$

The surface traction p_n generates a discontinuity in the normal stress of the global problem.

– If we replace the auxiliary reaction by the complement one, we have:

$$p_{n+1} = p_n + r_n \text{ with } r_{n+1} = -\left(\lambda_n^F + \lambda_n^C\right) \tag{13}$$

in words, the correction brought to p_{n+1} corresponds to the lack of balance between the complement zone and the fine representation of the zone of interest. This lack of balance is the residual r of the algorithm. The algorithm converges when the two representations are in equilibrium ($r = 0$, in which case the extra load p shall not evolve anymore).

– The algorithm makes no use of domain integral to communicate between subdomains; only interface data (on Γ) are exchanged, namely the displacement u^G and the reactions λ^F and λ^A (or λ^C). As long as the interface Γ is well represented in all models, it is not necessary to use the exact Ω^F in the auxiliary problem, any coarser representation is possible (Ω^A). Typically microperforations or micro cracks need not be represented in the auxiliary problem. Of course modifying the representation of the zone of interest may have consequences on the convergence of the algorithm (but not on its limit which is the reference solution).

3 Analysis of the algorithm

3.1 Notations

In order to further analyze the algorithm and be more practical, we now consider the finite element discretization of the problem. We use the following notations: $\mathbf{f}$ for the generalized forces, $\mathbf{u}$ for the nodal displacement and $\boldsymbol{\lambda}$ for the nodal reactions and $\mathbf{p}$ for the nodal component on the immersed surface effort. When indexing degrees of freedom, F, A, C stand for the internal degrees of freedom

whereas Γ stands for nodes on the interface (whose description is identical in all models). We tried to use minimal notations, but sometimes a quantity defined on the interface is issued from one side specifically, in which case we make it clear by an extra superscript. In the linear(ized) case notation $\mathbf{K}$ is used for the stiffness matrices.

Remark 2 We recall that the nodal reaction is not the discretization of the Lagrange multiplier. Indeed for a boundary degree of freedom i associated with shape function ϕ_i, we have:

$$\begin{aligned}\boldsymbol{\lambda}_i^X &= \int_{\Omega^X} (\sigma_h : \varepsilon(\phi_i) - f \cdot \phi_i)\, dx - \int_{\partial_n \Omega^X} g \cdot \phi_i \, dS \\ &= \int_{\Gamma} (\sigma_h^X \cdot n^X) \cdot \phi_i \, dS - \int_{\Omega^X} (\mathrm{div}(\sigma_h) + f) \cdot \phi_i \, dx + \int_{\partial_n \Omega} \Big(\sigma_h \cdot n^X - g\Big) \cdot \phi_i \, dS\end{aligned} \tag{14}$$

where $X \in \{C, A, F\}$, n^X is the outer normal vector and σ_h is the stress tensor obtained from the finite element computation. Thus the nodal reactions $\boldsymbol{\lambda}^X$ can be computed either by using a Lagrangian formulation for the Dirichlet condition (which is fairly common in legacy software) or by using the formula above to post-process it from the finite element stress (which may be complex to implement in legacy software); hence the use of the auxiliary model to compute reactions on the immersed interface.

We assume the well-posedness of the Global, Auxiliary and Fine problems, and we introduce the following operators:

$$\begin{aligned}\text{Global problem (7a)} : \mathbf{u}^G &= \mathcal{S}_G^{-1}(\mathbf{p}; \mathbf{f}^G) \\ \text{Fine problem (7b)} : \boldsymbol{\lambda}^F &= \mathcal{S}_F(\mathbf{u}^G; \mathbf{f}^F) \\ \text{Auxiliary problem (7c)} : \boldsymbol{\lambda}^A &= \mathcal{S}_A(\mathbf{u}^G; \mathbf{f}^A)\end{aligned} \tag{15}$$

$\mathcal{S}_X$ are nonlinear Dirichlet-to-Neumann operators, the effect of given load appear as parameter of the method, they will be omitted in the absence of ambiguity. Note that, because of the additivity of integral with respect to the domain, the Global operator verifies the following decomposition $\mathcal{S}_G = \mathcal{S}_C + \mathcal{S}_A$ and the reference operator also writes $\mathcal{S}_R = \mathcal{S}_C + \mathcal{S}_F$.

In the case of linear problems, it is possible to give an explicit formula for the Dirichlet to Neumann operators. As an illustration, the equilibrium of the Fine problem writes:

$$\begin{pmatrix} \mathbf{K}_{\Gamma\Gamma}^F & \mathbf{K}_{\Gamma F} \\ \mathbf{K}_{F\Gamma} & \mathbf{K}_{FF} \end{pmatrix} \begin{pmatrix} \mathbf{u}_\Gamma \\ \mathbf{u}_F \end{pmatrix} = \begin{pmatrix} \mathbf{f}_\Gamma^F \\ \mathbf{f}_F \end{pmatrix} + \begin{pmatrix} \boldsymbol{\lambda}^F \end{pmatrix} \tag{16}$$

which can be condensed as:

$$\boldsymbol{\lambda}^F = \mathcal{S}_F(\mathbf{u}, \mathbf{f}^F) = \mathbf{S}^F \mathbf{u}_\Gamma - \mathbf{b}^F \text{ with } \begin{cases} \mathbf{S}^F = \mathbf{K}_{\Gamma\Gamma}^F - \mathbf{K}_{\Gamma F}\mathbf{K}_{FF}^{-1}\mathbf{K}_{F\Gamma} \\ \mathbf{b}^F = \mathbf{f}_\Gamma^F - \mathbf{K}_{\Gamma F}\mathbf{K}_{FF}^{-1}\mathbf{f}_F \end{cases} \tag{17}$$

In that case, $\mathcal{S}_F$ is an affine operator: $\mathbf{S}^F$ is the well known Schur complement of the Fine domain on the interface. Linearity allows to set apart the contrition of the given load, with $\mathbf{b}^F$ the condensed right-hand side. We have $\mathbf{S}^R = \mathbf{S}^F + \mathbf{S}^C$, $\mathbf{b}^R = \mathbf{b}^F + \mathbf{b}^C$, $\mathbf{S}^G = \mathbf{S}^A + \mathbf{S}^C$, $\mathbf{b}^G = \mathbf{b}^A + \mathbf{b}^C$.

Remark 3 In the following, we allow the coarse and fine mesh not to be conforming at the interface Γ. We only impose Γ to coincide with boundary of elements in both meshes (in other words the geometry of Γ is well represented in both meshes).

If needed a transfer matrix $\mathbf{T}$ is assumed to have been computed (for instance using interpolation or Mortar techniques) such that the interface continuity is $\mathbf{T}\mathbf{u}_\Gamma^C - \mathbf{u}_\Gamma^F = 0$. In that case, the conservation of the mechanical works leads to the following interface balance: $\boldsymbol{\lambda}^C + \mathbf{T}^T\boldsymbol{\lambda}^F = 0$.

In order to simplify expressions, matrix $\mathbf{T}$ is omitted by setting $\mathbf{S}^F \leftarrow \mathbf{T}^T\mathbf{S}^F\mathbf{T}$, $\mathbf{b}^F \leftarrow \mathbf{T}^T\mathbf{b}^F$ and $\boldsymbol{\lambda}^F \leftarrow \mathbf{T}^T\boldsymbol{\lambda}^F$, which means that the interface in the Global model is considered to be the master.

Remark 4 When describing algorithms, we make use of the following functional notations:

- $[\mathbf{u}^G] = \texttt{SolveGlobal}(\mathbf{p};\mathbf{f}^G)$, $\mathbf{u}^G$ is defined on the whole global model and in particular we have $\mathbf{u}_\Gamma^G = \mathcal{S}_G^{-1}(\mathbf{p};\mathbf{f}^G)$.
- $[\mathbf{u}^F,\boldsymbol{\lambda}^F,\boldsymbol{\lambda}^A] = \texttt{SolveFine//Aux}(\mathbf{u}^G;\mathbf{f}^F,\mathbf{f}^A)$, which in particular corresponds to $\boldsymbol{\lambda}^F = \mathcal{S}^F(\mathbf{u}_\Gamma^G;\mathbf{f}^F)$ and $\boldsymbol{\lambda}^A = \mathcal{S}^A(\mathbf{u}_\Gamma^G;\mathbf{f}^A)$. The Fine and Auxiliary solves are gathered in one method because the computations can be run in parallel.

3.2 Stationary iterations

The global/local coupling iterations of equation (7) can formally be written as:

$$\begin{aligned}\mathbf{p}_{n+1} &= (\mathcal{S}_A - \mathcal{S}_F)\circ\mathcal{S}_G^{-1}(\mathbf{p}_n)\\ &= \mathbf{p}_n - \mathcal{S}_R\circ\mathcal{S}_G^{-1}(\mathbf{p}_n)\end{aligned} \tag{18}$$

One recognizes fixed point iterations. The convergence is controlled by the contraction property of the operator $I - \mathcal{S}_R\circ\mathcal{S}_G^{-1} = (\mathcal{S}_A - \mathcal{S}_F)\circ(\mathcal{S}_A + \mathcal{S}_C)^{-1}$.

Remark 5 In the linear case, the operator that controls the convergence can be written as:

$$(\mathbf{S}^A - \mathbf{S}^F)(\mathbf{S}_A + \mathbf{S}_C)^{-1} \tag{19}$$

since all Schur complements are symmetric positive definite, a trivial sufficient condition for the operator to be a contraction is $\mathbf{S}^A \geqslant \mathbf{S}^F$ (for the quadratic norm ordering). Mechanically speaking this means that the Auxiliary model shall be stiffer that the Fine one; this is usually the case when the Fine model has a refined mesh or holes. Moreover, we can expect the Auxiliary model to be a good approximation of the Fine model leading to ($\mathbf{S}^A \simeq \mathbf{S}^F$) and fast convergence.

A classical tweak to fixed point iterations is to use relaxation. This enables to grant some system contraction property or to improve the convergence rate. In that case, the iteration writes:

$$\mathbf{p}_{n+1} = \mathbf{p}_n - \omega_n\,\mathcal{S}_R\circ\mathcal{S}_G^{-1}(\mathbf{p}_n), \qquad |\omega| > \epsilon > 0 \tag{20}$$

For linear problems, it is well known that convergence is ensured for $0 < \omega < 2/\rho\left(\mathbf{S}^{G^{-1}}\mathbf{S}^R\right)$ and the optimal value is $\omega = 2/(\lambda_{\min}+\lambda_{\max})$ where the λs' stand for the minimal and maximal eigenvalues of $\mathbf{S}^{G^{-1}}\mathbf{S}^R$.

The existence result can be extended to the case of monotone problems. Indeed the method can be interpreted as an operator splitting technique (Ryu and Boyd 2015) on the condensed problems which inherits the useful properties of the original system (in particular monotonicity and coercivity).

In practive, it is convienent to have ω adapted at each step. A good heuristic for the sequence (ω_n) is provided by Aitken's Δ^2. It was first tried in the global/local framework in Liu et al. (2014). The strategy is summed-up in algorithm 1.

Algorithm 1: Non-invasive stationary iterations with relaxation

Arbitrary initialization $\mathbf{p}_0$
for $j \in [0, \cdots, m]$ **do**
  $[\mathbf{u}_j^G] = \texttt{SolveGlobal}(\mathbf{p}_j; \mathbf{f}^G)$
  $[\mathbf{u}^F, \boldsymbol{\lambda}_j^F, \boldsymbol{\lambda}_j^A] = \texttt{SolveFine//Aux}(\mathbf{u}_j^G; \mathbf{f}^F, \mathbf{f}^A)$
  Residual: $\mathbf{r}_j = -(\boldsymbol{\lambda}_j^F + \mathbf{p}_j - \boldsymbol{\lambda}_j^A)$
  Update: $\mathbf{p}_{j+1} = \mathbf{p}_j + \mathbf{r}_j$;
  Aitken Δ^2: $\mathbf{p}_{j+1} \leftarrow \omega_{j+1}\mathbf{p}_{j+1} + (1-\omega_{j+1})\mathbf{p}_j$ with $\omega_{j+1} = -\omega_j \frac{\mathbf{r}_{j-1}^T(\mathbf{r}_j - \mathbf{r}_{j-1})}{\|\mathbf{r}_j - \mathbf{r}_{j-1}\|^2}$
end

3.3 Quasi-Newton's approaches for linear Global model

The system to solve associated to the fixed point iterations (18) writes:

$$\text{Find } \mathbf{p} \ / \ \mathcal{S}_R \circ \mathcal{S}_G^{-1}(\mathbf{p}) = 0 \tag{21}$$

which mechanically means that we seek the surface traction to impose inside the Global coarse model (ie the stress discontinuity) such that the Fine model with Dirichlet conditions issued from the Global solve is in balance with the Complement zone.

Applying a Newton iteration to system (21) leads to the sequence:

$$(\mathbf{D}\mathcal{S}_R)\,(\mathbf{D}\mathcal{S}_G)^{-1}\,(\mathbf{p}_{n+1} - \mathbf{p}_n) = -\mathcal{S}_R \circ \mathcal{S}_G^{-1}(\mathbf{p}_n) \tag{22}$$

which was investigated in nonlinear relocalization techniques Negrello et al. (2016); Cresta et al. (2007); Pebrel et al. (2007); Hinojosa et al. (2014). but which is is not possible in general in a non-invasive framework. Anyhow, in the case of a linear Global model, it is possible to derive a use quasi-Newton approach.

If the Global problem is linear then the differential of the Global problem is constant and it is equal to the Schur complement $\mathbf{D}\mathcal{S}_G = \mathbf{S}^G$. Regarding the nonlinear part, we have:

$$\mathbf{D}\mathcal{S}_R = \mathbf{S}^C + \mathbf{D}\mathcal{S}_F = \underbrace{\mathbf{S}^C + \mathbf{S}^A}_{\mathbf{S}^G} + \underbrace{\mathbf{D}\mathcal{S}_F - \mathbf{S}^A}_{\mathbf{X}} \tag{23}$$

of course $\mathbf{X}$ is not computable in a non-invasive manner, but a low rank approximation is possible using quasi-Newton formulas. In particular, SR1 formula was tried

with success in Gendre et al. (2009). In practice, line search is not applied which makes the low rank update slightly lighter than usual, we note $\boldsymbol{\delta}_j^u = (\mathbf{u}_j - \mathbf{u}_{j-1})_{|\Gamma}$ and $\boldsymbol{\delta}_j^p = (\mathbf{p}_j - \mathbf{p}_{j-1})$, the increment of the interface quantities:

$$\begin{aligned} \mathbf{D}\mathcal{S}_{R,0} &= \mathbf{S}^G \\ \mathbf{D}\mathcal{S}_{R,i} &= \mathbf{D}\mathcal{S}_{R,i-1} + \frac{\mathbf{r}_i \mathbf{r}_i^T}{\mathbf{r}_i^T \boldsymbol{\delta}_i^u} \\ &= \mathbf{S}^G + \mathbf{R}_i \boldsymbol{\Delta}_i^{-1} \mathbf{R}_i^T \end{aligned} \tag{24}$$

with $\mathbf{R}_i = [\mathbf{r}_0 \dots \mathbf{r}_i]$ and $\boldsymbol{\Delta}_i = \mathrm{diag}(\mathbf{r}_i^T \boldsymbol{\delta}_i^u)$. Sherman Morrison formula leads to:

$$\mathbf{D}\mathcal{S}_{R,i}^{-1} = \mathbf{S}^{G^{-1}} - \mathbf{S}^{G^{-1}} \mathbf{R}_i \left(\boldsymbol{\Delta}_i + \mathbf{R}_i^T \mathbf{S}^{G^{-1}} \mathbf{R}_i \right)^{-1} \mathbf{R}_i^T \mathbf{S}^{G^{-1}} \tag{25}$$

It makes sense to first evaluate $\mathbf{S}^{G^{-1}} \mathbf{r}_i$ then apply corrections. For efficiency reasons, we also store the matrix $\mathbf{W}_i := \mathbf{S}^{G^{-1}} \mathbf{R}_i$. Note that the $(i+1) \times (i+1)$ gains one row and column at each iteration but previous coefficient are not modified. The method is recapitulated in the algorithm 2.

Algorithm 2: Non-invasive SR1 Quasi-Newton iterations

Arbitrary initialization $\mathbf{p}_0$
$[\mathbf{u}_0^G] = \texttt{SolveGlobal}(\mathbf{p}_0; \mathbf{f}^G)$
$[\mathbf{u}_0^F, \boldsymbol{\lambda}_0^F, \boldsymbol{\lambda}_0^A] = \texttt{SolveFine//Aux}(\mathbf{u}_0^G; \mathbf{f}^F)$
Residual: $\mathbf{r}_0 = -(\boldsymbol{\lambda}_0^F + \mathbf{p}_0 - \boldsymbol{\lambda}_0^A)$
$\mathbf{B}_0 = \mathbf{I}$
for $j \in [1, \cdots, m]$ **do**
  Predictor: $\tilde{\mathbf{p}}_j = \mathbf{p}_{j-1} + \mathbf{r}_{j-1}$
  $[\tilde{\mathbf{u}}_j^G] = \texttt{SolveGlobal}(\tilde{\mathbf{p}}_j; \mathbf{f}^G)$
  Set: $\mathbf{W}_j = [\mathbf{W}_{j-1}, \tilde{\mathbf{u}}_j^G - \mathbf{u}_j^G]$
  Compute: $\mathbf{x} = (\boldsymbol{\Delta}_j + \mathbf{W}_j^T \mathbf{R}_j)^{-1} \mathbf{W}_j^T \mathbf{r}_j$
  Corrector $\mathbf{u}_j^G = \tilde{\mathbf{u}}_j^G - \mathbf{W}_j \mathbf{x}$ and $\mathbf{p}_j = \tilde{\mathbf{p}}_j - \mathbf{R}_j \mathbf{x}$
  Set: $\boldsymbol{\delta}_j^u = \mathbf{u}_{j|\Gamma} - \mathbf{u}_{j-1|\Gamma}$
  $[\mathbf{u}_j^F, \boldsymbol{\lambda}_j^F, \boldsymbol{\lambda}_j^A] = \texttt{SolveFine//Aux}(\mathbf{u}_j^G; \mathbf{f}^F)$
  Residual: $\mathbf{R}_j = [\mathbf{R}_{j-1}, -(\boldsymbol{\lambda}_j^F + \mathbf{p}_j - \boldsymbol{\lambda}_j^A)]$
end

3.4 Conjugate gradients

3.4.1 Full linear case

This case occurs when all models are linear. Non-invasive global/local coupling can still be of interest to introduce complex local heterogeneities, stochastic behaviors or complex geometries in the Fine model.

For linear problems, it is rather classical to use Krylov accelerator on a stationary iteration. In our case, the problem to solve(21) is governed by the operator

$\mathbf{S}^R\mathbf{S}^{G^{-1}}$ which is symmetric in the $\mathbf{S}^{G^{-1}}$ inner-product. We then can derive a right-preconditioned conjugate gradients.

Beside the improved convergence compared to stationary iterations, using conjugate gradients allows an unconditional convergence (without necessity for the Auxiliary model to be stiffer than the Fine model).

Algorithm 3: Non-invasive right-preconditioned conjugate gradient

Arbitrary initialization $\mathbf{p}_0 = \mathbf{0}$
$[\mathbf{u}_0^G] = \texttt{SolveGlobal}(\mathbf{p}_0; \mathbf{f}^G)$
$[\mathbf{u}_0^F, \boldsymbol{\lambda}_0^F, \boldsymbol{\lambda}_0^A] = \texttt{SolveFine//Aux}(\mathbf{u}_0^G; \mathbf{f}^F, \mathbf{f}^A)$
Post-process:
$\boldsymbol{\lambda}_0^C = \mathbf{p}_0 - \boldsymbol{\lambda}_0^A$
$\mathbf{r}_0 = -(\boldsymbol{\lambda}_0^F + \boldsymbol{\lambda}_0^C)$
Initialization: $\underline{\mathbf{p}}_0 = \mathbf{r}_0$
Preconditioning: $[\underline{\mathbf{u}}_0^G] = \texttt{SolveGlobal}(\mathbf{p}_0 + \mathbf{r}_0; \mathbf{f}^G) - [\mathbf{u}_0^G]$
$\mathbf{y} = \underline{\mathbf{u}}_0^G$
for $j \in [0, \cdots, m]$ **do**
  $[\underline{\mathbf{u}}_j^F, \underline{\boldsymbol{\lambda}}_j^F, \underline{\boldsymbol{\lambda}}_j^A] = \texttt{SolveFine//Aux}(\mathbf{y} + \mathbf{u}_j^G; \mathbf{f}^F, \mathbf{f}^A) - [\boldsymbol{\lambda}_j^F, \boldsymbol{\lambda}_j^A]$
  Post-process: $d\boldsymbol{\lambda}_j^C = \mathbf{p}_j + \underline{\mathbf{p}}_j - [\boldsymbol{\lambda}_j^A + \underline{\boldsymbol{\lambda}}_j^A] - \boldsymbol{\lambda}_j^C$
  Direction of the variation of the residual: $\underline{\mathbf{r}}_j = \underline{\boldsymbol{\lambda}}_j^F + \underline{\boldsymbol{\lambda}}_j^C$
  Optimal step size: $\alpha_j = (\mathbf{r}_j^T \mathbf{y})/(\mathbf{y}^T \underline{\mathbf{r}}_j)$
  Updates: $\mathbf{p}_{j+1} = \mathbf{p}_j + \alpha_j \underline{\mathbf{p}}_j$; $\mathbf{r}_{j+1} = \mathbf{r}_j - \alpha_j \underline{\mathbf{r}}_j$
  $\mathbf{u}_{j+1}^G = \mathbf{u}_j^G + \alpha_j \mathbf{y}$; $\mathbf{u}_{j+1}^F = \mathbf{u}_j^F + \alpha_j \underline{\mathbf{u}}^F j$; $\boldsymbol{\lambda}_{j+1}^{C,A,F} = \boldsymbol{\lambda}_j^{C,A,F} + \alpha_j \underline{\boldsymbol{\lambda}}_j^{C,A,F}$
  Preconditioning: $[\tilde{\underline{\mathbf{u}}}_{j+1}^G] = \texttt{SolveGlobal}(\mathbf{p}_{j+1} + \mathbf{r}_{j+1}; \mathbf{f}^G) - [\mathbf{u}_{j+1}^G]$
  Orthogonalization: $\underline{\mathbf{p}}_{j+1} = \mathbf{r}_{j+1} - \beta_j \underline{\mathbf{p}}_j$ with $\beta_j = -(\underline{\mathbf{r}}_j^T \tilde{\underline{\mathbf{u}}}_{j+1})/(\underline{\mathbf{r}}_j^T \mathbf{y})$
  Correction: $\mathbf{y} = \tilde{\underline{\mathbf{u}}}_{j+1}^G + \beta_j \mathbf{y}$
end

3.4.2 Nonlinear case

Conjugate gradients can be extended to nonlinear cases using two ingredients:

– A line search algorithm to optimize the length of the steps. For a given search direction $\underline{\mathbf{p}}$, one tries to find the optimal length *alpha* in term of the minimization of some norm of the residual. This can be done in a non-invasive manner by a sampling technique with several lengths (α_i) tested in parallel. Classically these samples are used to interpolate the objective function and decide the final α. Because of the cost of the estimation of one configuration (one global solve followed by one local solve), we prefer to use directly the best sample already computed (except if the interpolated minimal let us expect a significantly better configuration).

– A "conjugation" technique for the definition of the new search direction $\underline{\mathbf{p}}$ given by one of the following heuristics (using the notations of algorithm 4):

$$\begin{aligned}
\underline{\mathbf{p}}_{j+1} &= -\mathbf{r}_{j+1} + \beta_j \underline{\mathbf{p}}_j \\
\text{Fletcher-Reeves} \qquad \beta_j &= \frac{\mathbf{r}_{j+1}^T \mathbf{r}_{j+1}}{\mathbf{r}_j^T \mathbf{r}_j} \\
\text{Polac-Ribière} \qquad \beta_j &= \frac{\mathbf{r}_{j+1}^T (\mathbf{r}_{j+1} - \mathbf{r}_j)}{\mathbf{r}_j^T \mathbf{r}_j} \\
\text{Hestenes-Stiefel} \qquad \beta_j &= \frac{\mathbf{r}_{j+1}^T (\mathbf{r}_{j+1} - \mathbf{r}_j)}{\underline{\mathbf{p}}_j^T (\mathbf{r}_{j+1} - \mathbf{r}_j)} \\
\text{Dai-Yuan} \qquad \beta_j &= \frac{\mathbf{r}_{j+1}^T \mathbf{r}_{j+1}}{\underline{\mathbf{p}}_j^T (\mathbf{r}_{j+1} - \mathbf{r}_j)}
\end{aligned} \tag{26}$$

Moreover it is often chosen to avoid negative steps by using $\beta_j \leftarrow \max(0, \beta_j)$

Algorithm 4: Non-invasive nonlinear conjugate gradient

Arbitrary initialization $\mathbf{p}_0$
$[\mathbf{u}_0^G] = \texttt{SolveGlobal}(\mathbf{p}_0; \mathbf{f}^G)$
$[\boldsymbol{\lambda}_0^F, \boldsymbol{\lambda}_0^A] = \texttt{SolveFine//Aux}(\mathbf{u}_0^G; \mathbf{f}^F, \mathbf{f}^A)$
Post-process: $\boldsymbol{\lambda}_0^C = \mathbf{p}_0 - \boldsymbol{\lambda}_0^A$, $\mathbf{r}_0 = -(\boldsymbol{\lambda}_0^F + \boldsymbol{\lambda}_0^C)$
Initialization: $\underline{\mathbf{p}}_0 = \mathbf{r}_0$
for $j \in [0, \cdots, m]$ **do**
  for *samples* $(\alpha_{j,i})_i$ **do**
    Preconditioning: $[\mathbf{u}_{j,i}^G, \boldsymbol{\lambda}_{j,i}^C] = \texttt{SolveGlobal}(\mathbf{p}_j + \alpha_{j,i}\underline{\mathbf{p}}_j; \mathbf{f}^G)$
    $[\boldsymbol{\lambda}_{j,i}^F, \boldsymbol{\lambda}_{j,i}^A] = \texttt{SolveFine//Aux}(\mathbf{u}_{j,i}^G; \mathbf{f}^F, \mathbf{f}^A)$
    Post-process: $\boldsymbol{\lambda}_{j,i}^C = \mathbf{p}_{j,i} - \boldsymbol{\lambda}_{j,i}^A$, $\mathbf{r}_{j,i} = -(\boldsymbol{\lambda}_{j,i}^F + \boldsymbol{\lambda}_{j,i}^C)$
  end
  Optimal step size (Line search): choose $I = \arg\min_i \|\mathbf{r}_{i,j}\|$
  Updates: $\mathbf{p}_{j+1} = \mathbf{p}_j + \alpha_{j,I}\underline{\mathbf{p}}_j$; $\mathbf{u}_{j+1}^{F,G} \leftarrow \mathbf{u}_{j,I}^{F,G}$; $\mathbf{r}_{j+1} \leftarrow \mathbf{r}_{j,I}$
  Pseudo-orthogonalization: $\underline{\mathbf{p}}_{j+1} = \mathbf{r}_{j+1} - \beta_j \underline{\mathbf{p}}_j$ with β_j given by formulas (26)
end

4 Connexion to optimized Schwarz method

The question of linking the non-invasive global-local coupling method to the many variants of domain decomposition and associated algorithms, like *chimera*, was studied in other publications like Hecht et al. (2009). Here we propose to connect the method with the iterations of a non-overlapping optimized Schwarz method. The theoretical framework of Schwarz method will allow us natural extensions to the method, in particular the use of overlaps to treat mesh incompatibilities.

4.1 Link to optimized Schwarz method

We consider a linear elasticity problem and two non-overlapping subdomains, Ω^C and Ω^F, the finite element system to solve writes:

$$\begin{aligned} \mathbf{K}^F\mathbf{u}^F &= \mathbf{f}^F + \mathbf{t}^{F^T}\boldsymbol{\lambda}^F & \boldsymbol{\lambda}^F + \boldsymbol{\lambda}^C &= 0 \\ \mathbf{K}^C\mathbf{u}^C &= \mathbf{f}^C + \mathbf{t}^{C^T}\boldsymbol{\lambda}^C & \mathbf{t}^F\mathbf{u}^F - \mathbf{t}^C\mathbf{u}^C &= 0 \end{aligned} \tag{27}$$

where $\mathbf{t}$ is the trace operator which extracts the interface degrees of freedom from a subdomain vector (thus $\mathbf{t}^T$ is the extension-by-zero operator). Left column corresponds to the equilibrium of subdomains submitted to the reaction from neighbors $\boldsymbol{\lambda}$; right column corresponds to interface conditions, namely balance of reactions and continuity of displacements (for simplicity reasons, we assumed conforming discretization).

The optimized Schwarz method consists in using Robin conditions at the interface, materialized by SPD matrices $\mathbf{Q}^C$ and $\mathbf{Q}^F$. The interface conditions rewrite:

$$\begin{aligned} (\boldsymbol{\lambda}^F + \boldsymbol{\lambda}^C) - \mathbf{Q}^C(\mathbf{t}^F\mathbf{u}^F - \mathbf{t}^C\mathbf{u}^C) &= 0 \\ (\boldsymbol{\lambda}^F + \boldsymbol{\lambda}^C) + \mathbf{Q}^F(\mathbf{t}^F\mathbf{u}^F - \mathbf{t}^C\mathbf{u}^C) &= 0 \end{aligned} \tag{28}$$

which can be combined with the equilibrium:

$$\begin{aligned} (\mathbf{K}^F + \mathbf{t}^{F^T}\mathbf{Q}^F\mathbf{t}^F)\mathbf{u}^F &= \mathbf{f}^F - \mathbf{t}^{F^T}\boldsymbol{\lambda}^C + \mathbf{t}^{F^T}\mathbf{Q}^F\mathbf{t}^C\mathbf{u}^C \\ (\mathbf{K}^C + \mathbf{t}^{C^T}\mathbf{Q}^C\mathbf{t}^C)\mathbf{u}^C &= \mathbf{f}^C - \mathbf{t}^{C^T}\boldsymbol{\lambda}^F + \mathbf{t}^{C^T}\mathbf{Q}^C\mathbf{t}^F\mathbf{u}^F \end{aligned} \tag{29}$$

Hence the alternate optimized Schwarz stationary iterations ($\boldsymbol{\lambda}_0^F = 0$, $\mathbf{u}_0^F = 0$):

$$\begin{aligned} (\mathbf{K}^C + \mathbf{t}^{C^T}\mathbf{Q}^C\mathbf{t}^C)\mathbf{u}^C_{n+\frac{1}{2}} &= \mathbf{f}^C - \mathbf{t}^{C^T}\boldsymbol{\lambda}^F_n + \mathbf{t}^{C^T}\mathbf{Q}^C\mathbf{t}^F\mathbf{u}^F_n \\ \boldsymbol{\lambda}^C_{n+\frac{1}{2}} &= \mathbf{t}^C\left(\mathbf{K}^C\mathbf{u}^C_{n+\frac{1}{2}} - \mathbf{f}^C\right) \\ (\mathbf{K}^F + \mathbf{t}^{F^T}\mathbf{Q}^F\mathbf{t}^F)\mathbf{u}^F_{n+1} &= \mathbf{f}^F - \mathbf{t}^{F^T}\boldsymbol{\lambda}^C_{n+\frac{1}{2}} + \mathbf{t}^{F^T}\mathbf{Q}^F\mathbf{t}^C\mathbf{u}^C_{n+\frac{1}{2}} \\ \boldsymbol{\lambda}^F_{n+1} &= \mathbf{t}^F\left(\mathbf{K}^F\mathbf{u}^F_{n+1} - \mathbf{f}^F\right) \end{aligned} \tag{30}$$

After condensation, the optimal values for $\mathbf{Q}^C$ and $\mathbf{Q}^F$ can be easily identified: $\mathbf{Q}^C = \mathbf{S}^F$ and $\mathbf{Q}^F = \mathbf{S}^C$, the impedance should be the condensed stiffness of the remaining part of the structure.

The global-local algorithm corresponds to the choice $\mathbf{Q}^C = \mathbf{S}^A$, which can be expected to be a good approximation of the optimal, and formally $\mathbf{Q}^F = \infty$ (Dirichlet condition).

The framework of Schwarz method enables us to recover the following features:

- Krylov acceleration: replacing stationary iterations by Krylov solvers is classical in Schwarz methods Gander and Halpern (2012). The conjugate gradients algorithm of section 3.4 is an example of such acceleration.
- Mixed approach: the condition $\mathbf{Q}^F = \infty$ is a poor approximation of the optimal choice. Many work exist on how to improve this approximation Gander et al. (2002), in Gendre et al. (2011) a two scale approximation of $\mathbf{S}^C$ was proposed.

- Parallel processing: the global-local method corresponds to the multiplicative version of the optimized Schwarz method. The additive (parallel) version could be tried in the non-invasive context. Note that this would only make sense in the presence of multiple Fine zones with finite Fine impedance $\mathbf{Q}^F < \infty$.
- Nonlinearity: stationary iterations can directly be transferred to nonlinear problems, in particular the ones with monotone operators (positive hardening) Badea (1991); Ladevèze and Simmonds (1999). The local-global method was successfully applied in many nonlinear problems like plasticity or cracking Gendre et al. (2009); Passieux et al. (2013)
- Overlapping version: optimized Schwarz methods also exist with overlaps. In Guguin et al. (2014), the overlap was used as a buffer zone to dampen edge effects in plate/3D coupling. In section 4.2, we present another application, the handling of non-matching meshes.

4.2 Overlapping version – handling of incompatible patches

In previous sections, we had assumed that the interface was described as the boundary of elements for all models. In practice this hypothesis is not so restrictive because most often the zone of interest is detected after an initial computation on the coarse global model, and it is constituted as a set of coarse elements satisfying a certain criterion. Even after remeshing, the boundary of the fine description of the zone of interest matches a set of coarse faces (edges in 2D). Then a "simple" transfer matrix $\mathbf{T}$ was required to communicate between models on the interface. In particular, the easy choice of $\mathbf{T}$ being the interpolation matrix of the coarse kinematics in the fine kinematics can be implemented in most software. More evolved choices like mortar connections can also be employed in certain software.

We propose here an alternative strategy which makes use of the possibility to have the models overlap. In that case, there is no restriction on the definition of the meshes. This idea can directly be connected to overlapping optimized Schwarz methods, yet we propose a mechanical interpretation of it.

The starting point is the observation that the method can be formulated as the search for $\mathbf{p}$ which is the stress discontinuity on the global model between the complement zone and the auxiliary description of the zone of interest. This discontinuity must be such that the complement zone is in equilibrium with the fine description of zone of interest loaded with Dirichlet conditions (21).

Since $\mathbf{p}$ is a discontinuity, in order it to be well described in the coarse finite element model, it must be supported by the boundary of coarse elements. But there is no need for the support of $\mathbf{p}$ to match the boundary of the zone of interest.

We thus propose to follow the figure 1. The zone of interest Ω^F is positioned where needed, its mesh is independent of the coarse mesh. We note $\Gamma^F = \partial\Omega^F$ the boundary of the zone of interest. The auxiliary zone is the largest set of coarse elements fully contained in the zone of interest. We note Γ^A the boundary of the auxiliary zone. The two interfaces Γ^F and Γ^A thus do not coincide. Ω^C is defined as the Complement to Ω^A in the Global problem.

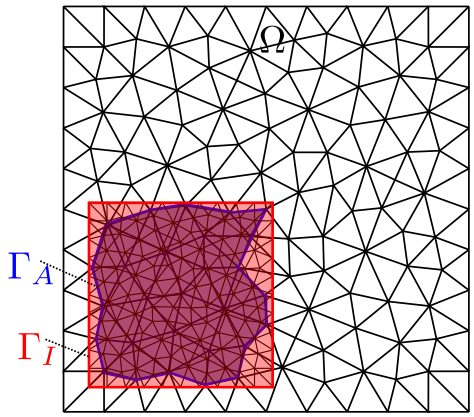


Fig. 1: Technique with overlap for non-conforming meshes

The stationary iteration goes that way:

$$
\begin{array}{rl}
\textbf{Global problem} & \mathbf{K}^G\mathbf{u}^G = \mathbf{f}^G + \mathbf{t}^T_{\Gamma^A}\mathbf{p} \\
\textbf{Aux. postprocess} & \left|\begin{array}{l} \boldsymbol{\lambda}^A = \mathbf{K}^A_{\Gamma^A\Gamma^A}\mathbf{u}^G_{\Gamma^A} + \mathbf{K}_{\Gamma^A A}\mathbf{u}^G_A - \mathbf{f}^A_{\Gamma^A} \\ \boldsymbol{\lambda}^C = \mathbf{K}^C_{\Gamma^A\Gamma^A}\mathbf{u}^G_{\Gamma^A} - \mathbf{K}_{\Gamma^A C}\mathbf{u}_C + \mathbf{f}^C_{\Gamma^A} = \mathbf{p} - \boldsymbol{\lambda}^A \end{array}\right. \\
\textbf{Fine problem} & \left|\begin{array}{l} \mathbf{K}^F\mathbf{u}^F = \mathbf{f}^F, \text{ with Dirichlet condition} \\ \mathbf{u}^F_j = u^G(x_j) \text{ for } j \text{ a Fine dof of } \Gamma^F \end{array}\right. \\
\textbf{Fine postprocess} & \left|\begin{array}{l} \text{for } i \text{ spanning all Global degrees of freedom on } \Gamma^A \\ \boldsymbol{\lambda}^F_i = \displaystyle\int_{\Omega^A}\left(\sigma^F_h : \varepsilon(\phi^G_i) - f\cdot\phi^G_i\right)dx - \int_{\partial_n\Omega^A} g\cdot\phi^G_i\, dS \end{array}\right. \\
\textbf{Residual} & \mathbf{r} = \boldsymbol{\lambda}^C + \boldsymbol{\lambda}^F \\
\textbf{Update} & \mathbf{p} = \boldsymbol{\lambda}^A - \boldsymbol{\lambda}^F = \mathbf{p} - \mathbf{r}
\end{array}
\tag{31}
$$

The main difficulty of this algorithm is the computation of the fine reaction on Γ^A with Ω^A not exactly represented on the coarse grid. This computation mixes the Fine stress σ^F_h and the coarse shape functions ϕ^G_i. Even if complex, this computation is feasible in many software. Anyhow in the nonlinear case, σ^F_h is only known at Fine Gauss points and the integral can only be approximated.

There is another conceptual difficulty which is common to many methods with overlap (Ben Dhia 1998): the definition of a monolithic reference problem. Indeed the domain between the interfaces (the overlap) is a buffer zone Ω^B ($\Gamma^A \cup \Gamma^F = \partial\Omega^B$) where the coarse and the fine model coexist. At convergence they are equivalent in the sense that they have the same value (up to interpolation error) on Γ^F (Dirichlet) and Γ^A (Neumann) and they solve the same problem in the overlap.

Note that the use of the overlap can be advantageous in the situations where edge effects can affect the fine model, even if meshes are conforming at the interfaces (Guguin et al. 2014).

5 Conclusion

Acknowledgement: this work was partially funded by the French National Research Agency as part of project ICARE (ANR-12-MONU-0002-04).

References

L. Badea, On the schwarz alternating method with more than two subdomains for nonlinear monotone problems. SIAM Journal on Numerical Analysis **28**(1), 179–204 (1991). doi:10.1137/0728010

H. Ben Dhia, Multiscale mechanical problems: the Arlequin method. Comptes Rendus de l'Academie des Sciences Series IIB Mechanics Physics Astronomy **326**(12), 899–904 (1998)

M. Chevreuil, A. Nouy, E. Safatly, A multiscale method with patch for the solution of stochastic partial differential equations with localized uncertainties. Computer Methods in Applied Mechanics and Engineering **255**(0), 255–274 (2013). doi:10.1016/j.cma.2012.12.003

P. Cresta, O. Allix, C. Rey, S. Guinard, Nonlinear localization strategies for domain decomposition methods: application to post-buckling analyses. Computer Methods in Applied Mechanics and Engineering **196**(8), 1436–1446 (2007)

M. Gander, F. Magoulès, F. Nataf, Optimized schwarz methods without overlap for the helmoltz equation. SIAM J. Sci. Comput. **24**(1), 38–60 (2002)

M.J. Gander, L. Halpern, Mathématiques pour l'ingénieur, in *Méthodes de décomposition de domaines*, (Techniques de l'Ingénieur, Saint-Denis, France, 2012)

L. Gendre, O. Allix, P. Gosselet, A two-scale approximation of the schur complement and its use for non-intrusive coupling. International Journal for Numerical Methods in Engineering **87**(9), 889–905 (2011)

L. Gendre, O. Allix, P. Gosselet, F. Comte, Non-intrusive and exact global/local techniques for structural problems with local plasticity. Computational Mechanics **44**(2), 233–245 (2009)

G. Guguin, O. Allix, P. Gosselet, S. Guinard, Nonintrusive coupling of 3d and 2d laminated composite models based on finite element 3d recovery. International Journal for Numerical Methods in Engineering **98**(5), 324–343 (2014)

F. Hecht, A. Lozinski, O. Pironneau, Numerical Zoom and the Schwarz Algorithm, in *Proceedings of the 18th conference on domain decomposition methods*, 2009

J. Hinojosa, O. Allix, P.A. Guidault, P. Cresta, Domain decomposition methods with nonlinear localization for the buckling and post-buckling analyses of large structures. Advances in Engineering Software **70**, 13–24 (2014)

P. Ladevèze, J.G. Simmonds, *Nonlinear computational structural mechanics: new approaches and non-incremental methods of calculation* (Springer Berlin, ???, 1999)

Y.J. Liu, Q. Sun, X.L. Fan, A non-intrusive global/local algorithm with non-matching interface: Derivation and numerical validation. Computer Methods in Applied Mechanics and Engineering **277**, 81–103 (2014)

C. Negrello, P. Gosselet, C. Rey, J. Pebrel, Substructured formulations of nonlinear structure problems — influence of the interface condition. International Journal for Numerical Methods in Engineering **online** (2016). doi:10.1002/nme.5195

J.-C. Passieux, J. Réthoré, A. Gravouil, M.-C. Baietto, Local/global non-intrusive crack propagation simulation using a multigrid X-FEM solver. Computational Mechanics, 1–13 (2013)

J. Pebrel, P. Gosselet, C. Rey, Une approche par patchs pour les non linéarités localisées en calcul de structures. 18ème Congrès Français de Mécanique (Grenoble 2007) (2007)

E.K. Ryu, S. Boyd, 2015, Primer on monotone operator methods

Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd edn. (Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2003). ISBN 0898715342

I Article [Guguin *et al.*, 2016], méthode appliquée à la simulation de liaisons boulonnées

Cet article montre comment la méthode de couplage non-intrusive peut apporter une réponse à une problématique industrielle forte : la simulation des liaisons boulonnées.

Les structures élancées sont fréquemment représentées à l'aide d'éléments finis plaque ou coque. Ces modèles de calcul permettent à moindre coût d'obtenir une représentation fiable des grands flux d'efforts au sein d'une structure. Néanmoins, ils sont notoirement inexacts près des accidents géométriques et en particulier au niveau des liaisons qui sont en plus généralement représentées par des éléments connecteurs extrêmement grossiers dont le domaine de validité est très réduit.

L'idée que nous défendons est de « remplacer non-intrusivement » le connecteur par une représentation 3D fine du boulon (avec contact frottant et éventuellement comportement non-linéaire). Le connecteur initial devient un simple paramètre du calcul qui influe sur la convergence de la méthode mais pas sur le modèle obtenu. Cette application s'appuie sur la technique de couplage plaque 3D décrite dans [Guguin *et al.*, 2014] et sur les accélérations présentées dans l'annexe précédente.

Guguin *et al. Adv. Model. and Simul. in Eng. Sci.(2016)3:16*
DOI 10.1186/s40323-016-0069-5

Advanced Modeling and Simulation in Engineering Sciences

RESEARCH ARTICLE Open Access

On the computation of plate assemblies using realistic 3D joint model: a non-intrusive approach

Guillaume Guguin[1*], Olivier Allix[1], Pierre Gosselet[1] and Stéphane Guinard[2]

*Correspondence: guguin@lmt.ens-cachan.fr
[1]LMT-Cachan, ENS-Cachan/CNRS/Université Paris-Saclay, 61 avenue du Président Wilson, 94235 Cachan, France
Full list of author information is available at the end of the article

Abstract

Most large engineering structures are described as assemblies of plates and shells and they are computed as such using *adhoc* Finite Element packages. In fact their computation in 3D would be much too costly. In this framework, the connections between the parts are often modeled by means of simplified tying models. In order to improve the reliability of such simulations, we propose to apply a non-intrusive technique so as to virtually substitute the simplified connectors by a precise 3D nonlinear model, without modifying the global plate model. Moreover each computation can be conducted on independent optimized software. After a description of the method, examples are used to analyze its performance, and to draw some conclusions on the validity and limitation of both the modeling of junction by rigid connectors and the use of submodeling techniques for the estimation of the carrying capacity of bolted plates.

Keywords: Assembly, Non-intrusive coupling, Bolt

Background

The simulation of large structures undergoing complex local nonlinear phenomena is still a major scientific and industrial challenge. One of the main difficulties originates from the difference of length scale between the global response of the structure and the localized phenomena. To address those problems a first type of computational approach is based on homogenization, as FE^2 [1] but it works well as long as the scales are sufficiently separated. To overcome this limitation concurrent multiscale methods have been developed. They are often based on domain decomposition techniques like FETI [2], FETI-DP [3] or the LATIN multiscale method [4,5] and its optimization for the multiscale treatment of nonlinear problems [6,7].

Moreover most of large industrial structures are described as an assembly of plates and shells, whereas local phenomena often require 3D models to be properly analyzed. To deal with such problems, several methods have been applied or developed for the coupling of 2D and 3D models, like the Arlequin method [8,9], transition elements [10,11], MPCs approaches [12,13] or Nitsche's method [14].

Most of these methods are quite demanding in terms of software development and therefore they are seldom used in industrial packages. To overcome these drawbacks, non-

intrusive approaches have recently been proposed [15]. They are nowadays the subject of extensions and developments: thermoelasticity with GFEM/FEM coupling [16], crack propagation in XFEM/FEM coupling [17], stochastic simulations [18] and dynamics [19, 20].

In [21] a non-intrusive coupling between plate and 3D models was proposed in the case of linear behaviors. The present paper concerns the extension of this approach to the simulation of bolted assemblies of plates where bolts are described with full 3D nonlinear models. Such structures are good candidates for the iterative global-local non-intrusive strategy for two main reasons. First, the detailed computation of a tightened bolt with frictional contact on all surfaces is a very hard and time-consuming task to perform using commercial software. Second, the construction of the reference model, which would correspond to the assembly of a plate model and a 3D model for the bolt, would be very complex. The non-intrusive framework provides answers to both these problems. First, it allows the use of dedicated software for the local computations (in our case, COFAST a parallel software based on the LATIN domain decomposition method [22]). Second, it allows the easy coupling of a general Finite Element software (here Code_Aster from EDF) for the plate computation, with COFAST, because the global model is unchanged during the iterative process. These properties were exploited in [23] for the simulation of damage in composite laminates at the meso and the micro scales using dedicated pieces of software.

The non-intrusive framework aims at solving the reference problem iteratively, by solving at each iteration both the global problem with prescribed residual traction at the interface and the local nonlinear problems submitted to prescribed displacement. Several techniques have been proposed to improve the convergence rate of the method by means of acceleration techniques [15,17,24] or improved interface conditions [25]. The method has therefore common points with so called nonlinear domain decomposition methods (or nonlinear relocalization techniques) [26,27] which proved their efficiency and gain in robustness in the case of buckling [28], post-buckling [29] and damage analysis [30,31]. Other proposals have been made to take into account the fact that the phenomena of interest are localized, aiming at a better representation of the target model [16,32,33].

The paper is organized as follows. In "The reference problem" section, a summary of the reference problem corresponding to the coupling of 2D and 3D models according to [21] is presented. The non-intrusive algorithm is presented in "The non-intrusive iterative algorithm" section. In "Analysis of the iterative corrections at the global and local levels" section, the results of the iterative coupling are analyzed in the case of a bolted joint. These results are compared to those corresponding to the plate solution and to the submodeling technique. In "Control of some parameters of the method and acceleration technique" section, the influence of some parameters is assessed regarding the rate of convergence of the iterative process and regarding the accuracy of the coupled model compared to a full 3D solution.

The reference problem

We consider the typical problem of two plates connected by a bolt. Starting from a simplified model of the assembly using plates elements and a simple connector (typically a beam), our aim is to perform the non-intrusive substitution of the connector by a full 3D model of the bolt. A sketch of the two models is presented in Fig. 1. In what follows lower

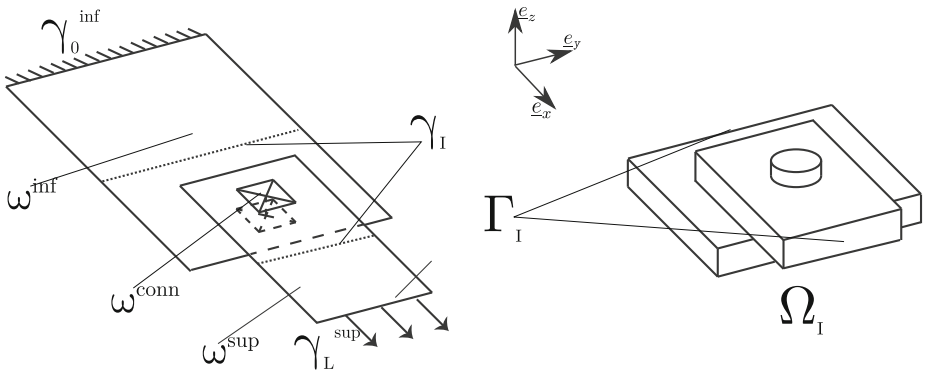


Fig. 1 *Left*, global model: two plates assembled by one 1D connector. *Right*, local model: full representation of the bolt with surrounding plates

case Greek characters are used to describe the geometry of the plates whereas capital Greek characters are used for the geometry of the 3D domains: any 3D domain Ω_X possesses a plate counterpart ω_X, so does any interface Γ_Y whose plate counterpart is written γ_Y.

The target hybrid model, which will be precisely defined in this section, corresponds to plate models connected by a full 3D model of the bolt. At the convergence of the iterations, the solution of the hybrid model is obtained, as illustrated in Fig. 2. Let us note that since the problem is symmetric [in the $(\underline{e}_x, \underline{e}_z)$ plane] only half of the bolt is computed and shown on the figure.

The plate model

The global plate model is the assembly of two plates, ω^{inf} and ω^{sup} respectively the lower plate and the top plate, and a rigid connector between them ω^{conn} Fig. 1 (left). The plates are 20 mm thick and 280 mm long. The lower plate is 160 mm wide whereas and the top plate is 80 mm wide. In order to simplify the presentation, the plates are assumed to be made out of homogeneous isotropic linear elastic material with Young modulus $E = 200$ GPa and Poisson ratio $\nu = 0.3$; the handling of orthotropic composite plates is explained in [21].

The lower plate ω^{inf} is clamped on the part γ_0^{inf} of its boundary and a prescribed tension displacement ($\underline{e}_x$ direction) is applied on the part γ_L^{sup} of the top plate's boundary, Fig. 1:

$$\begin{cases} \underline{U} = \underline{0} & \text{on } \gamma_0^{inf} \\ \underline{U} = U\,\underline{e}_x & \text{on } \gamma_L^{sup} \end{cases} \tag{1}$$

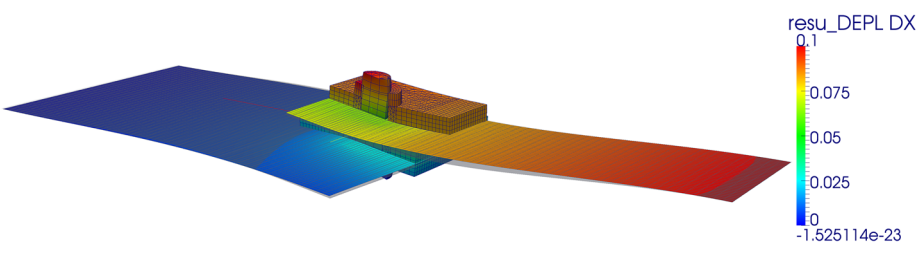


Fig. 2 Deformed shape of the targeted coupled problem

Note that this tension is applied after preload (tightening) was applied to the bolt (see the description of the patch).

A classical Reissner-Mindlin plate formulation is used, and the contact is not taken into account at this stage. The kinematic is given by:

$$\underline{U}(m,z) = \underline{V}(m) + w(m)\underline{e}_z - \underline{e}_z \wedge \underline{\theta} = \underline{V}(m) + w(m)\underline{z} - z \begin{bmatrix} \theta_y(m) \\ -\theta_x(m) \\ 0 \end{bmatrix} \tag{2}$$

where $m \in \omega^{inf} \cup \omega^{sup}$ is a point of the mid-surface of the plates and $\underline{e}_z$ the normal direction (in the thickness). The stress in the plates is described with the eight generalized forces ($\underline{\underline{N}}$ tension, $\underline{\underline{M}}$ bending, $\underline{Q}$ shear):

$$\mathbb{F} = \left(N_{xx}, N_{yy}, N_{xy}, M_{xx}, M_{yy}, M_{xy}, Q_x, Q_y\right) \tag{3}$$

Since the connector will ultimately be virtually replaced by the 3D model described in next subsection, a very crude connector is chosen: a perfectly rigid beam element surrounded by a rigid region of the size of the bolt (20 mm) in order to avoid localization around one node, see Fig. 3.

Beside its simplicity, this connector has the advantage to be very rigid compared to its 3D counterpart, which is a sufficient condition to ensure the convergence of the coupling algorithm [15,34].

The 3D patch

The local patch is designed to replace the connector. Thus, a full 3D representation of the bolt is used with unilateral frictional contact interfaces between each part of the assembly. The dimensions of the bolt are given in Fig. 4a. The problem being symmetric, only half of the bolt is computed. The material used for the screw and the nut is linear elastic with Young's modulus $E = 300$ GPa and Poisson's ratio $\nu = 0.3$. Coulomb's friction coefficient is equal to 0.3 on all interfaces.

To solve this model, we use the dedicated contact solver COFAST, developed in the LMT-Cachan by Champaney [22]. This software uses the LATIN method [35] where interfaces between subdomains are the support of the contact conditions, see Fig. 4b.

Note that before coupling with the global problem, preload is applied by enforcing relative displacements between the nut and the screw. It is adjusted to match realistic values of tension in the screw (200 MPa).

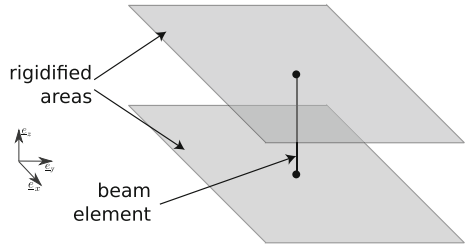


Fig. 3 The definition of the connector

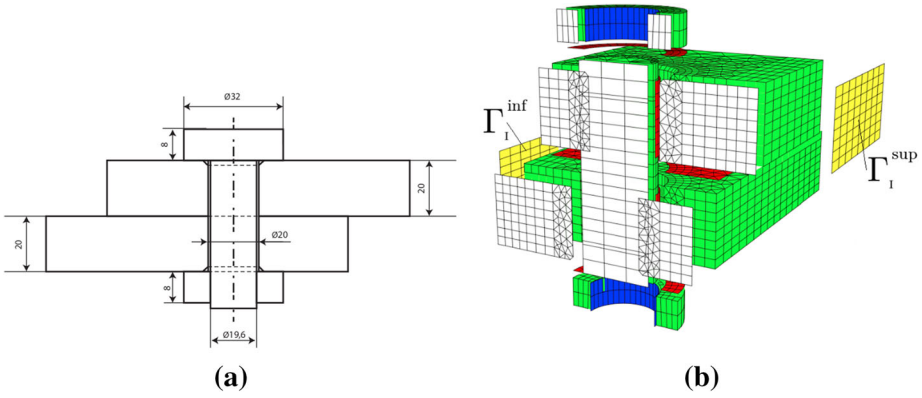


Fig. 4 3D representation of the bolt. **a** Dimensions of the bolt. **b** COFAST modeling

Connections between the models

The plates + connector model describes the entirety of the structure and occupies the (2D + 1D) domain ω. The 3D model of the bolt occupies what we call the zone of interest Ω_I. In the original method [15], the coarse modeling of $\omega_I = \omega \cap \Omega_I$ was simply replaced by Ω_I through iterations; in [21], it was shown that because of the edge effects which affect plate solutions, it is interesting to introduce a zone of transition between the two models.

This idea is sketched in Fig. 5: the 3D model is the only one taken into account in the inner zone of interest $\Omega_{\widetilde{I}} \subset\subset \Omega_I$, the plate model is the only one taken into account in the outer complement zone $\omega_{\widetilde{C}} \subset\subset \omega_C$, there exists an overlap $\Omega_I \cap \omega_C$ also called buffer zone where the two models are equivalent in a certain sense.

Two interfaces are defined: $\gamma_I = \partial\Omega_I \cap \omega_C$ and $\gamma_C = \partial\omega_C \cap \Omega_I$. In order to simplify the example, regular and ruled quadrilateral meshes are used along the interfaces. Thus, γ_I and γ_C are straight lines between two conforming meshes, see Fig. 6.

The substituted solution is expressed with the following form:

$$\underline{u}^{hyb} = \begin{cases} \underline{u}^L & \text{in } \Omega_I \text{ solves the 3D equations} \\ \underline{u}^G & \text{in } \omega_C \text{ solves the plate equations} \end{cases} \tag{4}$$

and it satisfies the following transmission conditions (in a sense which is made more precise afterwards):

$$\begin{cases} \underline{u}^L \text{ and } \underline{u}^G \text{ are continous on } \Gamma_I \\ \text{3D stress and plate generalized forces are balanced on } \gamma_C \end{cases} \tag{5}$$

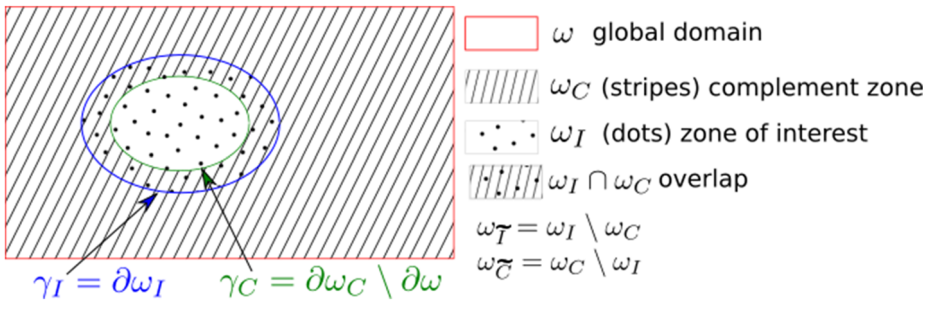


Fig. 5 The domain decomposition with overlap and the interfaces

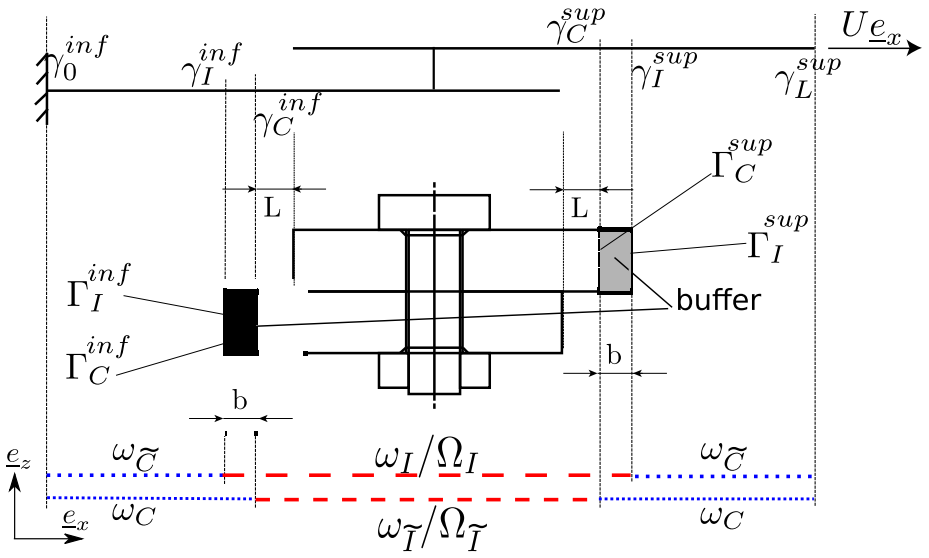


Fig. 6 Definition of the size of the local model and of the buffer zone

To be more specific, let us assume that a finite element discretization is used: $\mathbf{u}^G$ is the vector of plate unknowns defined on ω, $\mathbf{u}^L$ is the vector of 3D unknowns defined on Ω_I, $\mathbf{f}_{ext}$ stands for the generalized forces (they include the effect of imposed displacements), $\mathbf{K}$ for the stiffness matrices, and $\mathbf{f}_{int}$ for the internal forces (work of the stress field in the finite element subspace).

The global plate model is enriched by an extra loading $\boldsymbol{\delta}$ which is only non-zero on the degrees of freedom of the interface γ_C (the values of $\boldsymbol{\delta}$ are determined by the coupling algorithm). The global plate model equilibrium thus can be written as:

$$\mathbf{K}^G\mathbf{u}^G = \mathbf{f}_{ext}^G + \boldsymbol{\delta} \tag{6}$$

Note that the assumption of a linear global model simplifies the explanation and speeds up the solving (because the factorization is done only once) but it is not a requirement of the coupling algorithm.

The local 3D equilibrium submitted to boundary conditions inherited from the plate computation can be written as:

$$\begin{cases} \mathbf{f}_{int}^L(\mathbf{u}^L) = -\mathbf{f}_{ext}^L & \text{in } \Omega_I \\ \mathbf{u}^L = \mathbf{R}\mathbf{u}^G + \mathbf{w}^G(\mathbf{u}^G) & \text{on } \Gamma_I \end{cases} \tag{7}$$

Matrix $\mathbf{R}$ represents the classical 3D rigid section movements. It enables to express the 3D Reissner-Mindlin kinematic from the plate degrees of freedom ($\mathbf{u}^G$). Because the rigid section kinematic is a very coarse hypothesis, we supplement it with warping displacements $\mathbf{w}^G(\mathbf{u}^G)$, proportional to the plate stress state. To be more precise, the applied displacement on Γ_I has the following form:

$$\underline{u}^L(m,z) = \underbrace{\underline{u}(m) + w(m)\underline{e}_z - \underline{e}_z \wedge \underline{\theta}}_{\mathbf{R}\mathbf{u}^G} + \underbrace{\sum_{i=1}^{8} \underline{\mu}_i^I(z)\mathbb{F}_i^{\widetilde{C}}}_{\mathbf{w}^G(\mathbf{u}^G)}$$

with $(\mathbb{F}_i^{\widetilde{C}})$ the eight generalized forces extracted along the interface γ_I (3). $(\underline{\mu}_i^I(z))$ is the basis of warping adapted to the stacking and materials which was determined before the computation by solving preliminary problems [21]. These preliminary problems conducted on a 3D representative volume element of the composite plate enable us to compute distributions of stress and displacement (warping) associated with any solution of Saint-Venant problems. In turn, Saint-Venant problems are associated with specific components of the plate generalized stresses.

From the global and local computations, it is possible to post-process the plate and 3D nodal reactions on the γ_C/Γ_C interface:

$$\begin{aligned}\boldsymbol{\lambda}^C &= \left(\mathbf{K}^C_{|\omega_C}\mathbf{u}^G - \mathbf{f}^C_{ext|\omega_C}\right)_{|\gamma_C}\\ \boldsymbol{\lambda}^L &= -\left(\mathbf{f}^L_{int|\Omega_{\bar{I}}}(\mathbf{u}^L) + \mathbf{f}^L_{ext|\Omega_{\bar{I}}}\right)_{|\Gamma_C}\end{aligned} \tag{8}$$

Finally the balance of the nodal reactions is enforced on the static plate quantities, thanks to the transpose operator $\mathbf{R}^T$ which enables to compute the generalized forces associated to the 3D nodal reactions $\boldsymbol{\lambda}^L$:

$$\boldsymbol{\lambda}^C + \mathbf{R}^T\boldsymbol{\lambda}^L = 0 \tag{9}$$

In spite of the care taken to recover the 3D boundary conditions on Γ^I from plate quantities, small edge effects may occur, in particular in the case of stratified plates where suppressing edges effects require high order theories which are not always available in finite element analysis software or which could difficultly be implement in a non-intrusive manner. The role of the buffer zone (the overlap between Ω^I and ω^C) is then to dampen these edge effects and make the evaluation of the 3D stress reliable on Γ^C. In the example below, the size of the buffer zone is set to $b = 5\,\text{mm}$, which was sufficient to absorb the local artificial effects. It corresponds to four elements between the two interfaces Γ_C and Γ_I. The dimension b is represented on Fig. 6, together with dimension L which characterizes the size of the 3D domain of interest Ω_I. L corresponds to the size of the part of the 3D domain which is not strictly necessary to represent the bolt correctly but which was inserted as a way to keep the 3D/2D transition away from the zone dominated to by 3D effects.

The non-intrusive iterative algorithm

The system (6, 7, 8, 9) can be interpreted as finding the traction $\boldsymbol{\delta}$ to be imposed to the global plate model on the inner interface γ_C in order to generate a reaction λ^C in balance with the reaction of the inner zone of interest submitted to the recovery of the plate displacement on its boundary Γ_I.

Starting from $\boldsymbol{\delta}_0 = 0$, this can be achieved through the following iterations:

(1) Run a global plate analysis with extra load $\boldsymbol{\delta}$:

$$\mathbf{u}^G_n = \mathbf{K}^{G^{-1}}\left(\mathbf{f}^G_{ext} + \boldsymbol{\delta}\right)$$

(2) Post-process the reaction on γ_C:

$$\boldsymbol{\lambda}^C_n = \left(\mathbf{K}^C\mathbf{u}^G_{n|\omega_C} - \mathbf{f}^C_{ext}\right)_{|\gamma_C}$$

(3) Recover the 3D displacement on Γ_I:

$$\mathbf{u}^L_{n|\Gamma^I} = \mathbf{R}\mathbf{u}^G_{n|\Gamma_I} + \mathbf{w}^G(\mathbf{u}^G)_n$$

(4) Solve the local problem with imposed displacement on Γ_I:

$$\begin{cases} \mathbf{f}_{int}^L(\mathbf{u}_n^L) + \mathbf{f}_{ext}^L = 0 \\ \mathbf{u}_{n|\Gamma_I}^L \text{ given} \end{cases}$$

(5) Post-process the local reaction of Γ_C:

$$\boldsymbol{\lambda}_n^L = -\left(\mathbf{f}_{int}^L(\mathbf{u}_n^L) + \mathbf{f}_{ext}^L\right)_{|\Gamma_C}^{\widetilde{I}}$$

(6) Compute the residual on γ_C:

$$\mathbf{r}_n = \boldsymbol{\lambda}_n^C + \mathbf{R}^T \boldsymbol{\lambda}_n^L$$

(7) If residual is small enough then exit, else update the extra load:

$$\boldsymbol{\delta}_{n+1|\gamma_C} = \boldsymbol{\delta}_{n|\gamma_C} - \mathbf{r}_n$$

and go back to 1.

One important point is that the *global step* 1 and the *local step* 4 can be processed with different software. For the test-case developed here, Code_Aster is used for the global plate problem, and COFAST3D is used for the local 3D contact problem.

In the end the local/global method is a fixed point algorithm similar to a modified Newton algorithm. The rate of convergence of this algorithm can be quite slow. However acceleration techniques can be applied:

- quasi-Newton acceleration like SR1 algorithm [25] or BFGS,
- dynamic relaxation like Aitken's algorithm (Δ^2) [24],
- mixed boundary conditions [25],
- Krylov solvers (only in the linear case) [34].

Based on [34], it appears that, among those methods, SR1 "Quasi-Newton acceleration" technique leads to better performance while being simple to implement in a non-intrusive manner.

Note that running one global analysis (step 1) followed by a local reanalysis with given Dirichlet conditions (steps 3–4) without iterations corresponds to the industrialists' practice called submodeling (or sometimes structural zoom). Such an approach is "purely descending" in the sense that there is no feedback from the local computation towards the global scale.

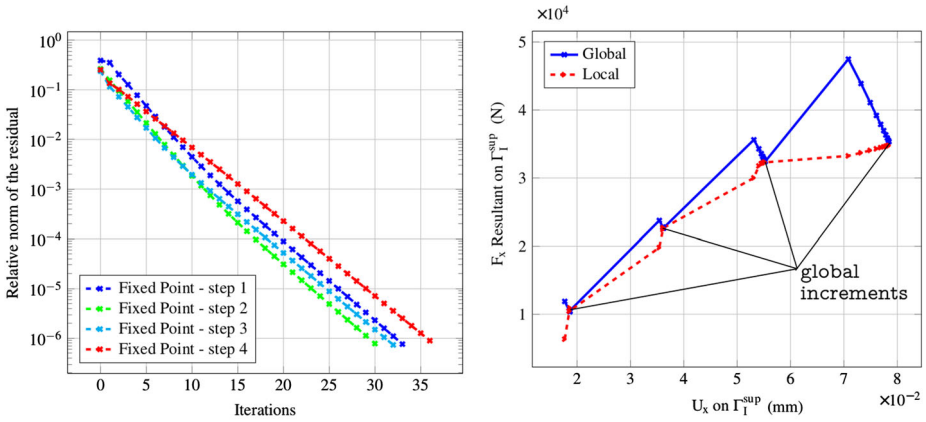
Analysis of the iterative corrections at the global and local levels

In order to evaluate the convergence of the algorithm, the relative norm of the residual at iteration i is defined as:

$$r_i^{rel} = \frac{\|\boldsymbol{\lambda}_i^C + \mathbf{R}^T \boldsymbol{\lambda}_i^L\|_2}{\|\boldsymbol{\lambda}_i^C\|_2} \tag{10}$$

In this section, the corrections associated with the coupling along the iterations are analyzed for the fixed-point algorithm (i.e., without acceleration techniques) with convergence threshold set to 10^{-6} which is clearly enough for mechanical quantities to have converged.

A typical convergence curve is shown Fig. 7a. This curve shows that the convergence is more or less independent of the level of nonlinearity in the bolt (weak for the first global



(a) Evolution of the residual during iterations. **(b)** Traction F_x along Γ_I^{sup} (top plate.)

Fig. 7 Convergence of the fixed-point algorithm: **a** Relative norm of the residual. **b** Evolution of one component of the global and local nodal reactions

time step, strong for the last one). This type of result is typical of nonlinear relocalization methods [28–31]. Let us note that, as shown in Fig. 7b, four global time steps are used for the plate computation, which appears to be sufficient, see "Control of some parameters of the method and acceleration technique" section. Anyhow, sub-stepping is used for the computation of the bolt; balance between the local and global model is only reached at the global time steps in Fig. 7b. For more involved problems, a better control of the global time steps and of the local sub-stepping process could be obtained using error indicators on the time discretization like in [36].

Remark 1 With the proposed technique, for each global increment the first iteration corresponds to a classical submodeling approach: this enables us to easily measure the quality of a that approach, which is a question often raised by engineers. In this application, the level of the error of the submodeling approach is about 25 %.

Global effect of the correction

Effect of the preload of the bolt

The global effect of the tightening of the bolt cannot be predicted using a plate theory because the associated forces are equal to zero. Nevertheless this tightening has a non-negligible global effect as can be seen in Fig. 8.

Additional global effects of the bolt on the plate solution

After the preload of the bolt, the plate is loaded in tension, in four global time steps. The analysis of the number of global increments is presented in "Control of some parameters of the method and acceleration technique" section. The global solution is modified along the iterations to match with the 3D model of the bolt, as can be seen on Fig. 9 which shows the values of the transverse displacement in the mid-section of the plate for the initial plate solution and the corrected one.

One can notice that the correction increases with the time steps, and it becomes significant for the third step and very large for the last one. This is of course due to the fact that for the first two time steps the bolt acts more or less as a 3D rigid connector. Whereas

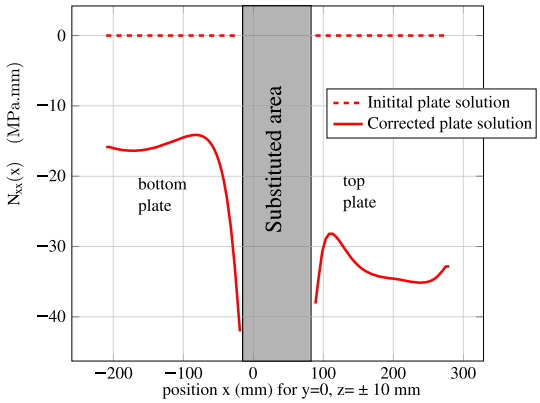


Fig. 8 Effect of the tightening (preload): value of the $\mathbf{N}_{xx}$ component of the plate stress tensor

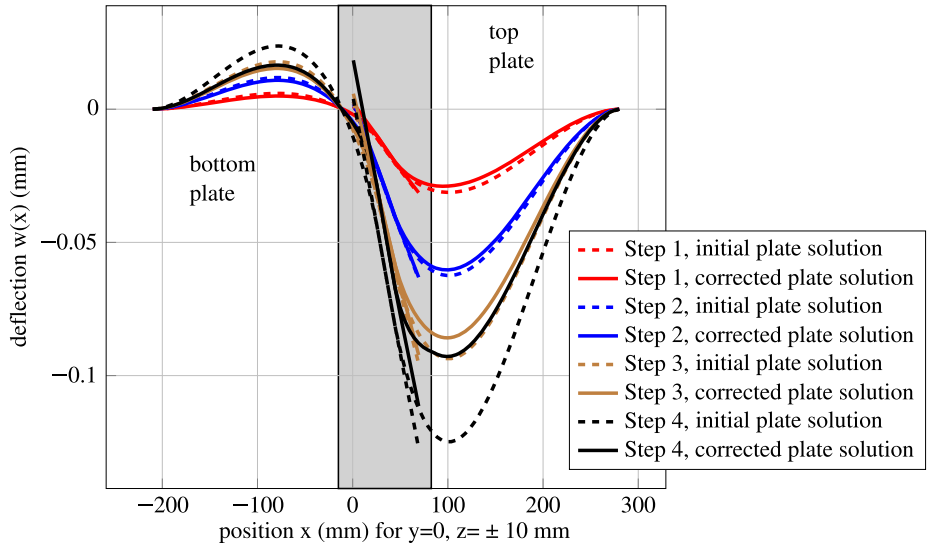


Fig. 9 Deformation of the plates, before and after iterations

the sliding of the bolt becomes significant for the third time step. During the last stage of the loading loosening of the bolt happens, as can be checked from the local analysis of the solution presented in the next sub-section. Let us recall that the initial plate solution also corresponds to what would be obtained by a submodeling approach since submodeling only improves the local reanalyzed area.

Local effect of the correction

In this section the solution within the bolt is compared for three different approaches: a reference full 3D simulation, a submodeling approach and the mixed 2D–3D model with $L = 0$ (minimal size of the local 3D model) obtained at the convergence of the iterations (convergence threshold is 10^{-6}).

For example Fig. 10 shows the comparison of the σ_{xz} stress field between the submodeling solution, the hybrid solution and the reference one. It can be seen that the solution associated with the non-intrusive approach matches very well the 3D reference solution

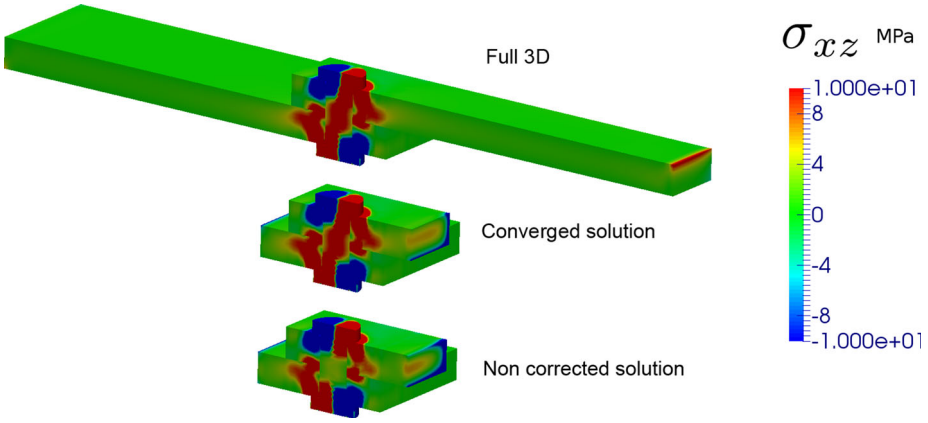


Fig. 10 Transverse shear stress disbribution: reference (*top*), converged 2D–3D solution (*middle*), submodeling (*bottom*)

while the submodeling gives inaccurate results inside the screw where the shear stress is greatly underestimated.

We now compare global quantities associated to the bolt (like the amount of dissipation due to sliding or the global axial component of the force acting on the joint) as well as local quantities (like the pointwise values of the sliding) for the three models. On Fig. 11a, b, local stress quantities are extracted on a point defined on Fig. 11d. It appears that the relative residual provides an efficient indicator for the errors on global quantities. Typically the submodeling leads to 20 % of error on the dissipation. Note that the error committed by submodeling on local quantities can be much larger. In comparison, the converged 2D–3D solution and the full 3D solution are quite close.

Figure 12 presents the sliding obtained by the converged 2D–3D model; it matches the 3D reference (few percents of deviation on the maximal sliding). On the contrary, the submodeling approach underestimates the sliding especially for the third and fourth global time steps, as can be seen on Fig. 13 where the error on the maximal sliding is more than 30 %.

To conclude, the coupling approach seems reliable contrarily to the submodeling approach for which the level of error is 20–25 % on global quantities and can be much more for local quantities of interest. The lack of conservatism of the submodeling extends similar results obtained in previous studies on localized plasticity or buckling for example.

Control of some parameters of the method and acceleration technique

As already discussed several parameters can be tuned for the non-intrusive 2D–3D coupling strategy:

- the number of global time steps,
- the choice of the position of the interface in the global model (dimension L in Fig. 6),
- the width of the buffer zone (dimension b in Fig. 6).

The size of the buffer zone has been chosen from our experience on the 2D–3D coupling involving composite plates and orthotropic plies [21], in order to minimize the problem

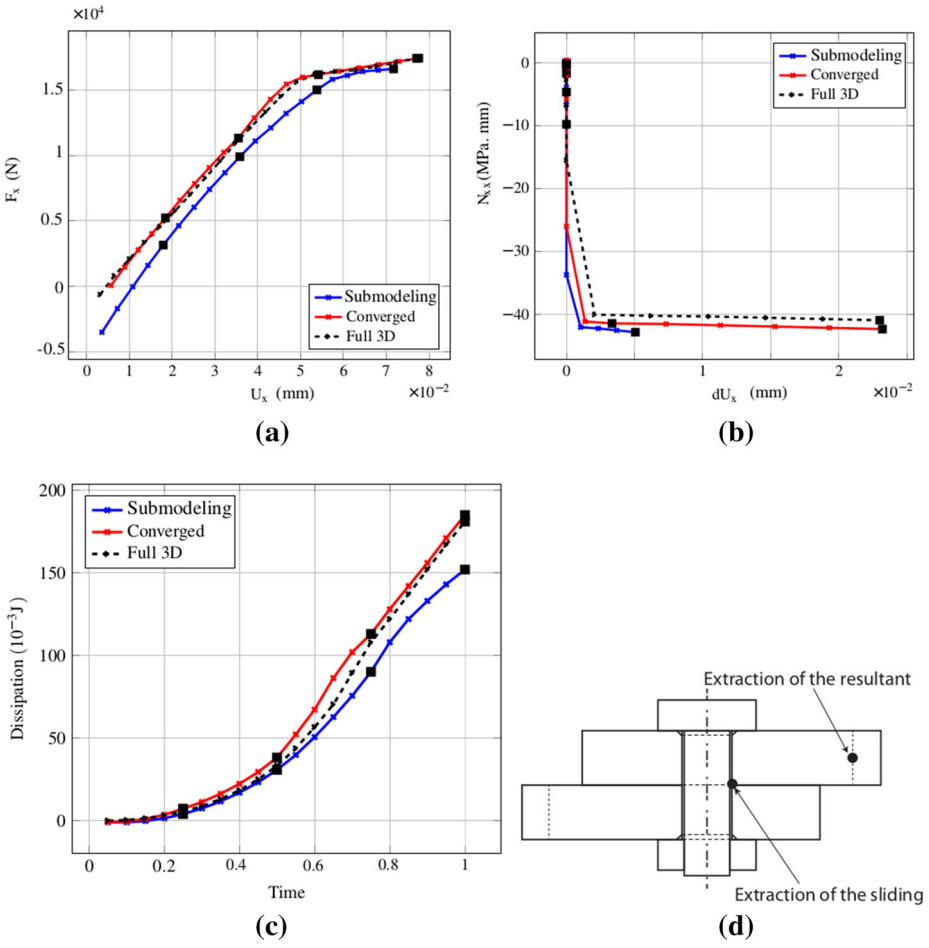


Fig. 11 Comparison between full 3D, submodeling and 2D–3D approaches. **a** Response of the bolt on Γ_C^{sup}. **b** N_{xx} vs sliding near the hole. **c** Dissipation between the plate. **d** Extraction points

of artificial edge effects between the two models. Besides the parameters of the nonlinear solver used for the bolt computation have been tuned in order to ensure a high precision according to dedicated papers like [22].

Influence of the number of global time steps

The number of time steps has been simply chosen on the basis of the comparison of solutions, starting from one global time step and increasing this number. On Fig. 14, a comparison between 1, 2 and 4 time steps is shown. The solutions with 2 and 4 time steps are considered here as quite close, whereas the one using only one time step deviates largely in the middle of the load sequence. Let us note that the use of large time steps at the global level is made possible thanks to the chosen algorithm with nonlinear localization. Indeed the possibility to use sub-stepping at the local scale greatly reduces the potential difficulties of convergence.

Influence of the position of the interface between the 2D and the 3D models

The question of the position of the interface raises in fact the question of the validity of the plate theory with respect to the 3D theory and has been largely discussed in [21].

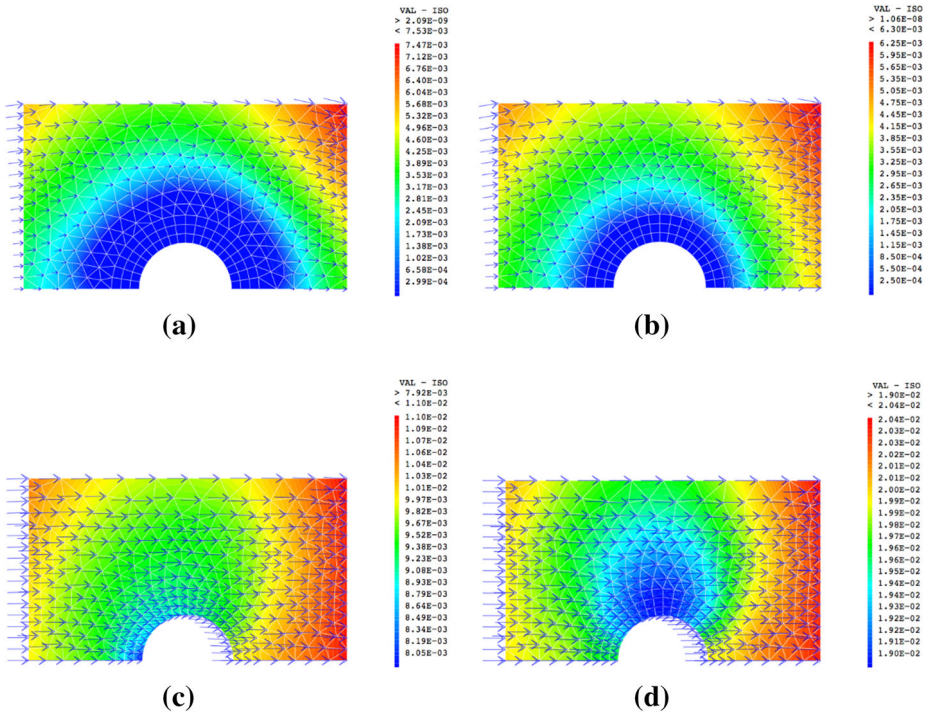


Fig. 12 Sliding at each global time step of the converged solution (mm). **a** Step 1; **b** Step 2; **c** Step 3; **d** Step 4

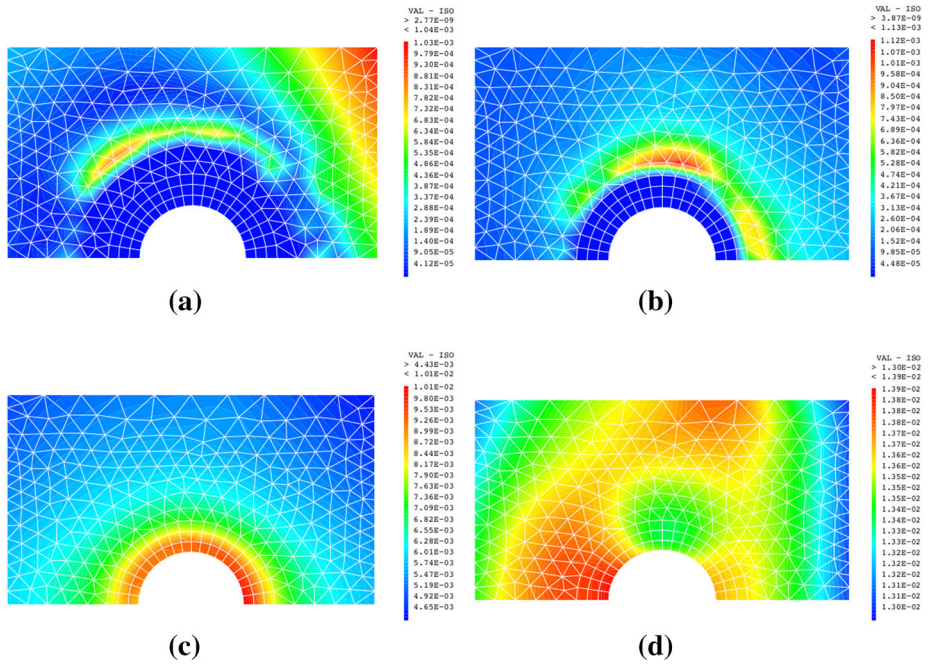


Fig. 13 Maps of the difference of sliding between converged and submodeling solutions at each global time step (mm). **a** Step 1; **b** Step 2; **c** Step 3; **d** Step 4

From what is known on the validity of the plate and shell theories, in the case of isotropic materials, one expects the 2D–3D model to be a good approximation of the 3D reference, for an interface situated from the bolt at a distance superior to the thickness of the plate.

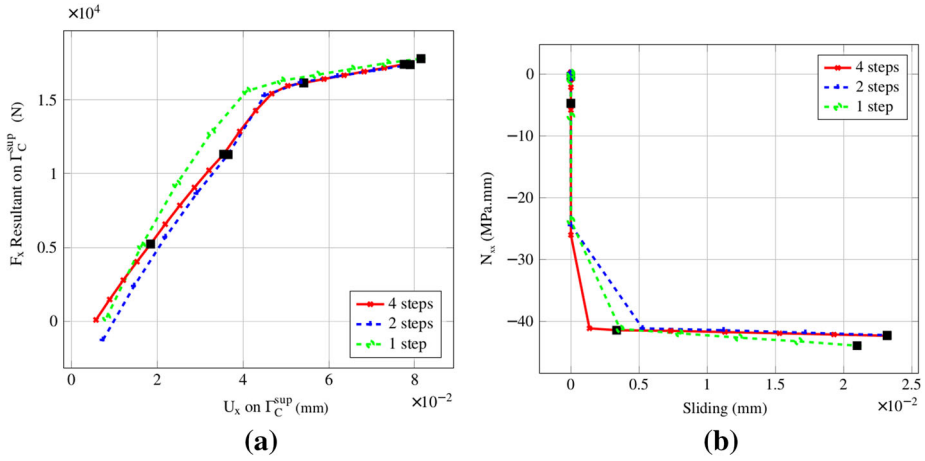


Fig. 14 Comparison of solutions for different number of global time steps. **a** Observation of the bolt response on Γ_C^{sup}. **b** N_{xx} vs sliding near the screw

It is therefore interesting to analyze the influence of the position of the interface and to compare the cases of $L = 0$ mm and $L = 15$ mm (see Fig. 6).

From a global point of view, the final deformed shapes of the corrected solutions in Fig. 15c are very close in the common plate domain. They only slightly differ in the local area of interest which are different for the two models.

From a local point of view, on Fig. 15b, both cases give close results. This is confirmed when analyzing the difference of the shear stress within the bolt $\Delta\sigma_{xz} = \sigma_{xz}^{conv} - \sigma_{xz}^{0}$, as shown in the Fig. 16. As already analyzed, if for the first two steps, most of the differences are localized around the nut, during the sliding phase the correction mostly concerns the nut itself. Figure 17 shows the evolution of the local tangential jump with respect to the prescribed displacement. This is an interesting quantity in order to observe the initiation of the sliding. The 2D–3D models give close predictions for both values of L. They are much more closer to the reference than what is predicted by the simple submodeling.

On the Fig. 15a, it appears, as expected, that a larger number of iterations has to be carried out when the interface is located on the bolt boundaries ($L = 0$). This effect of the interface location on the convergence rate is corrected when using a SR1 acceleration technique as can be seen in the next subsection.

Quasi-Newton acceleration

A final component of the non-intrusive substitution is the implementation of acceleration techniques. In that context, the SR1 quasi-Newton acceleration was tested in [25], and the Aitken Delta-2 dynamic relaxation method in [17]. For this study, we only implement SR1 acceleration.

Figure 18 presents convergence plots. The convergence is roughly three times faster with SR1. Moreover the rate of convergence is almost independent from the load step and the position of the interface. This property can be explained by the fact that the corrections induced by the SR1 acceleration technique are adapted to take into account the main differences between the 2D and 3D models.

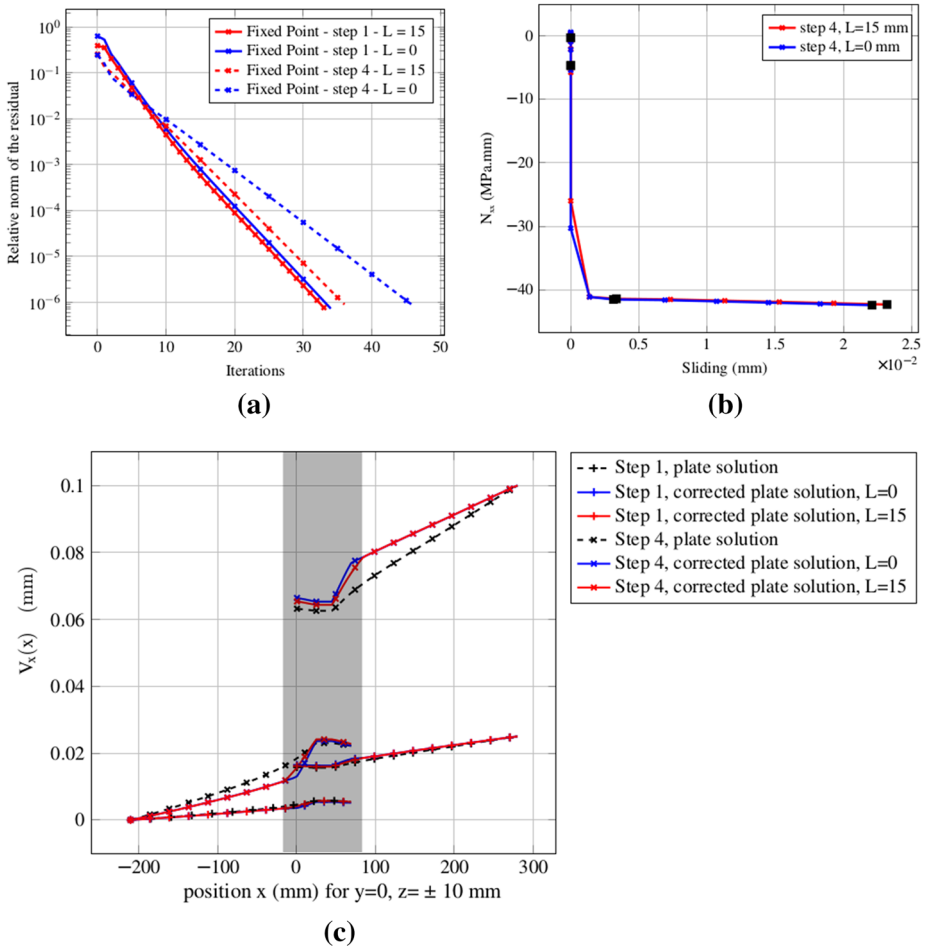


Fig. 15 Comparison of the results for $L = 0$ mm and $L = 15$ mm. **a** Residual during the iteration, **b** N_{xx} vs sliding near the hole, **c** axial displacement

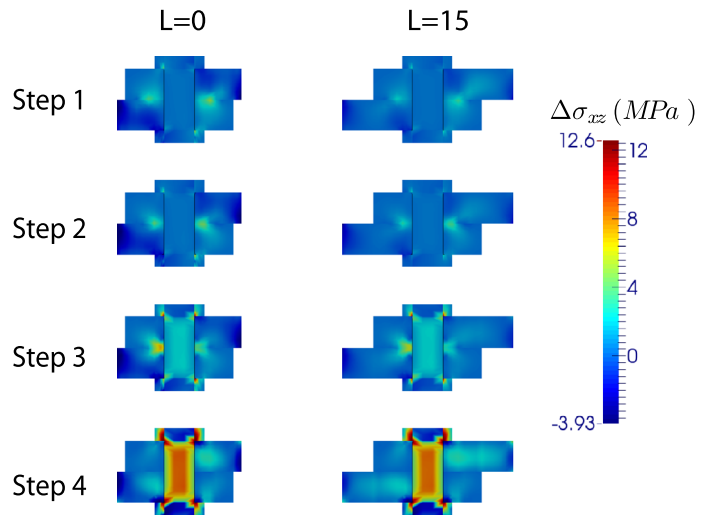


Fig. 16 Error in the local domain for two positions of the interface and at each time step

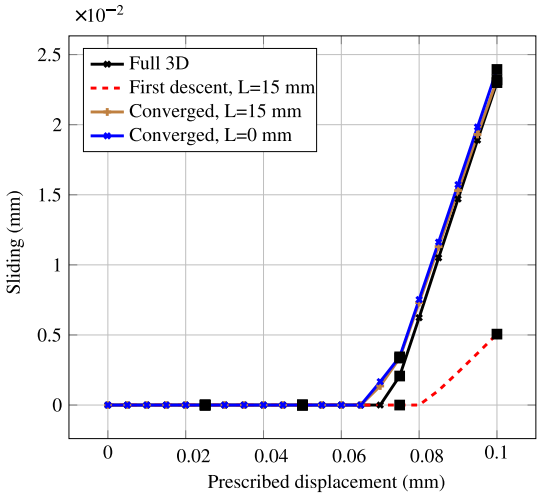


Fig. 17 Sliding between the two plates near the nut for different configurations

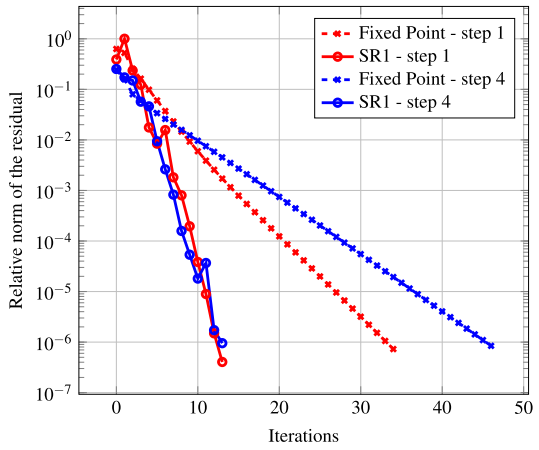


Fig. 18 Effect of the SR1 acceleration on the global–local rate of convergence

Conclusions

In this paper, a non-intrusive coupling between plate models and 3D models [21] has been extended to deal with the precise computations of bolted plates: the plate model with simplified connector was coupled with a full 3D nonlinear model of the bolt. The flexibility of the method was exploited to easily define the coupled model and to use of two dedicated pieces of software: Code_Aster for the plate computations, and COFAST3D for the nonlinear computation of the bolt (including many surfaces of friction).

The proposed technique enabled us to analyze the possibilities and limits of the use of plate connectors and submodeling approach in that case. It appears that, even when the bolt response is globally linear, the 3D effects induced by the bolt largely modify the plate solution itself. This shows that a rigid connector is a poor model to describe the connection between two plates. As expected, when important sliding occurs, such modeling becomes irrelevant. Moreover, the submodeling technique may lead to significant local errors and non-conservative results.

Another important feature for future applications concerning the treatment of multiple bolts in interaction, is that using an interface located at the limit of the bolt leads to acceptable results when compared to the reference solution. Works on that type of problem is in progress.

Another issue concerns the modeling of the bolt itself. In practice, some of the parameters of the bolted assembly are not precisely determined, as the preload of the nut or the friction coefficient. Such types of problems have been analyzed by dedicated techniques for the bolt computation [37,38], including multiresolution [39]. The use of the proposed non-intrusive techniques allows to extend these studies to the case of 2D–3D structural analyses in a straightforward manner. In addition, the use of model reduction techniques for the global model itself, as proposed in [40] should lead to a very significant reduction of the computational time.

Authors' contributions

OA initiated the method with the help of PG. GG implemented the method and conducted the numerical experiments on the test cases provided by SG. OA and GG drafted the manuscript, PG finalized it. All authors read and approved the final manuscript.

Author details

[1] LMT-Cachan, ENS-Cachan/CNRS/Université Paris-Saclay, 61 avenue du Président Wilson, 94235 Cachan, France , Airbus Group Innovation, 18 rue Marius Tercé Zac St Martin du Touch, 31000 Toulouse, France.

Acknowledgements

This work was partially funded by the French National Research Agency as part of project ICARE (ANR-12-MONU-0002-04).

Competing interests

The authors declare that they have no competing interests.

Received: 26 August 2015 Accepted: 26 April 2016
Published online: 21 May 2016

References

1. Feyel F, Chaboche J-L. FE2 multiscale approach for modelling the elastoviscoplastic behaviour of long fibre SiC/Ti composite materials. Comput Methods Appl Mech Eng. 2000;183(3):309–30.
2. Wyart E, Coulon D, Duflot M, Pardoen T, Remacle J-F, Lani F. A substructured FE-shell/XFEM-3D method for crack analysis in thin-walled structures. Int J Numer Methods Eng. 2007;72(7):757–79.
3. Amini AM, Dureisseix D, Cartraud P. Multi-scale domain decomposition method for large-scale structural analysis with a zooming technique: Application to plate assembly. Int J Numer Methods Eng. 2009;79(4):417–43.
4. Ladevèze P, Dureisseix D. A micro/macro approach for parallel computing of heterogeneous structures. Int J Comput Civil Struct Eng. 2000;1:18–28.
5. Ladevèze P, Loiseau O, Dureisseix D. A micro–macro and parallel computational strategy for highly heterogeneous structures. Int J Numer Methods Eng. 2001;52:121–38.
6. Kerfriden P, Allix O, Gosselet P. A three-scale domain decomposition method for the 3D analysis of debonding in laminates. Comput Mech. 2009;44(3):343–62.
7. Saavedra K, Allix O, Gosselet P. On a multiscale strategy and its optimization for combined delamination and buckling simulation. Int J Numer Methods Eng. 2012;91:772–98.
8. Dhia HB. Problèmes mécaniques multi-échelles: la méthode arlequin. Comptes Rendus de l'Académie des Sciences-Series IIB-Mechanics-Physics-Astronomy. 1998;326(12):899–904.
9. Dhia HB, Rateau G. The arlequin method as a flexible engineering design tool. Int J Numer Methods Eng. 2005;62(11):1442–62.
10. Gmür TC, Kauten RH. Three-dimensional solid-to-beam transition elements for structural dynamics analysis. Int J Numer Methods Eng. 1993;36(9):1429–44.
11. Garusi E, Tralli A. A hybrid stress-assumed transition element for solid-to-beam and plate-to-beam connections. Comput Struct. 2002;80(2):105–15.
12. McCune RW, Armstrong CG, Robinson DJ. Mixed-dimensional coupling in finite element models. Int J Numer Methods Eng. 2000;49(6):725–50.
13. Donaghy RJ, McCune W, Bridgett SJ, Armstrong CG, Robinson DJ, McKeag RM. Dimensional reduction of analysis models. In: 5th international meshing roundtable. Sandia National Laboratories; 1996. p. 307–20.
14. Nguyen VP, Kerfriden P, Brino M, Bordas SP, Bonisoli E. Nitsche's method for two and three dimensional nurbs patch coupling. Comput Mech. 2014;53(6):1163–82.
15. Gendre L, Allix O, Gosselet P, Comte F. Non-intrusive and exact global/local techniques for structural problems with local plasticity. Comput Mech. 2009;44(2):233–45.
16. Plews J, Duarte C, Eason T. An improved nonintrusive global-local approach for sharp thermal gradients in a standard fea platform. Int J Numer Methods Eng. 2012;91(4):426–49.

17. Duval M, Passieux JC, Salaün M, Guinard S. Non-intrusive coupling: recent advances and scalable nonlinear domain decomposition. Arch Comput Methods Eng. 2016;23(1):17–38. doi:10.1007/s11831-014-9132-x.
18. Chevreuil M, Nouy A, Safatly E. A multiscale method with patch for the solution of stochastic partial differential equations with localized uncertainties. Comput Methods Appl Mech Eng. 2013;255:255–74.
19. Bettinotti O, Allix O, Malherbe B. A coupling strategy for adaptive local refinement in space and time with a fixed global model in explicit dynamics. Comput Mech. 2014;53(4):561–74. doi:10.1007/s00466-013-0917-9.
20. Bettinotti O, Allix O, Perego U, Oancea V, Malherbe B. A fast weakly intrusive multiscale method in explicit dynamics. Int J Numer Methods Eng. 2014;8:577–94.
21. Guguin G, Allix O, Gosselet P, Guinard S. Nonintrusive coupling of 3d and 2d laminated composite models based on finite element 3d recovery. Int J Numer Methods Eng. 2014;98(5):324–43.
22. Champaney L, Cognard JY, Ladéveze P. Modular analysis of assemblages of three-dimensional structures with unilateral contact conditions. Comput Struct. 1999;73(1–5):249–66.
23. Daghia F, Ladevèze P. A micro–meso computational strategy for the prediction of the damage and failure of laminates. Compos Struct. 2012;94(12):3644–53.
24. Liu Y, Sun Q, Fan X. A non-intrusive global/local algorithm with non-matching interface: derivation and numerical validation. Comput Methods Appl Mech Eng. 2014;277:81–103.
25. Gendre L, Allix O, Gosselet P. A two-scale approximation of the schur complement and its use for non-intrusive coupling. Int J Numer Methods Eng. 2011;87(9):889–905.
26. Klawonn A, Lanser M, Rheinbach O. Nonlinear FETI-DP and BDDC methods. SIAM J Sci Comput. 2014;36(2):737–65.
27. Negrello C, Gosselet P, Rey C, Pebrel J. Substructured formulations of nonlinear structure problems — influence of the interface condition. Int J Numer Methods Eng. 2016. **(online)**.
28. Cresta P, Allix O, Rey C, Guinard S. Nonlinear localization strategies for domain decomposition methods: application to post-buckling analyses. Comput Methods Appl Mech Eng. 2007;196:1436–46.
29. Hinojosa J, Allix O, Guidault PA, Cresta P. Domain decomposition methods with nonlinear localization for the buckling and post-buckling analyses of large structures. Adv Eng Softw. 2014;70:13–24.
30. Bordeu F, Boucard PA, Gosselet P. Balancing domain decomposition with nonlinear relocalization: parallel implementation for laminates. In: Proceedings of the 1st international conference on parallel, distributed and grid computing for engineering, vol 90. 2009. p. 46–57.
31. Allix O, Kerfriden P, Gosselet P. A relocalization technique for the multiscale computation of delamination in composite structures. Comput Model Eng Sci. 2010;55(3):271–92.
32. Düster A, Scholz D, Rank E. pq-adaptive solid finite elements for three-dimensional plates and shells. Comput Methods Appl Mech Eng. 2007;197(1):243–54.
33. Düster A, Niggl A, Rank E. Applying the hp-d version of the FEM to locally enhance dimensionally reduced models. Comput Methods Appl Mech Eng. 2007;196(37–40):3524–33.
34. Guguin G. Stratégie non-intrusive de couplage plaque/3D pour la simulation des assemblages de plaques composites stratifiées. PhD thesis, ENS de Cachan, 61 av. Président Wilson, 94235 Cachan, France. 2014.
35. Ladevèze P, Simmonds JG. Nonlinear computational structural mechanics: new approaches and non-incremental methods of calculation. Berlin: Springer; 1999.
36. Allix O, Kerfriden P, Gosselet P. On the control of the load increments for a proper description of multiple delamination in a domain decomposition framework. Int J Numer Methods Eng. 2010;83:1518–40.
37. Gant F, Champaney L, Rouch P. Modeling of the bolted joint behavior variability with the lack of knowledge theory. In: ICCES 2010 international conference on computational and experimental engineering and sciences, Las Vegas, vol. 28. 2010. p. 97–8.
38. Gant F, Rouch P, Champaney L. Updating of uncertain joint models using the lack-of-knowledge theory. Comput Struct. 2013;128:128–35.
39. Allix O, Vidal P. A new multi-solution approach suitable for structural identification problem. CMAME. 2002;191:2727–58.
40. Kerfriden P, Passieux J-C, Bordas SP-A. Local/global model order reduction strategy for the simulation of quasi-brittle fracture. Int J Numer Methods Eng. 2012;89(2):154–79.