Schwarz methods and boundary integral equations

Pierre Marchand

Sorbonne Université
Inria

Doctoral School **Sciences Mathématiques de Paris Centre (ED 386)**

University Department **Laboratoire Jacques-Louis Lions**

Thesis defended by **Pierre Marchand**

Defended on **21st January, 2020**

In order to become Doctor from Sorbonne Université

Academic Field **Applied mathematics**

Schwarz methods and boundary integral equations

Thesis supervised by Xavier Claeys
Frédéric Nataf

Committee members

Referees	Xavier Antoine	Professor at Université de Lorraine	
	Timo Betcke	Professor at University College London	
Examiners	Stéphanie Chaillat-Loseille	Junior Researcher at CNRS	
	Frédéric Hecht	Professor at Sorbonne Université	Committee President
Supervisors	Xavier Claeys	Associate Professor at Sorbonne Université	
	Frédéric Nataf	Senior Researcher at CNRS	

Sorbonne Université
Inria

Doctoral School **Sciences Mathématiques de Paris Centre (ED 386)**

University Department **Laboratoire Jacques-Louis Lions**

Thesis defended by **Pierre Marchand**

Defended on **21st January, 2020**

In order to become Doctor from Sorbonne Université

Academic Field **Applied mathematics**

Schwarz methods and boundary integral equations

Thesis supervised by Xavier Claeys
Frédéric Nataf

Committee members

Referees	Xavier Antoine	Professor at Université de Lorraine	
	Timo Betcke	Professor at University College London	
Examiners	Stéphanie Chaillat-Loseille	Junior Researcher at CNRS	
	Frédéric Hecht	Professor at Sorbonne Université	Committee President
Supervisors	Xavier Claeys	Associate Professor at Sorbonne Université	
	Frédéric Nataf	Senior Researcher at CNRS	

Sorbonne Université
Inria

École doctorale **Sciences Mathématiques de Paris Centre (ED 386)**

Laboratoire de recherche **Laboratoire Jacques-Louis Lions**

Thèse présentée par **Pierre Marchand**

Soutenue le **21 janvier 2020**

En vue de l'obtention du grade de docteur de Sorbonne Université

Discipline **Mathématiques appliquées**

Méthodes de Schwarz et équations intégrales de frontière

Thèse dirigée par Xavier Claeys
Frédéric Nataf

Composition du jury

Rapporteurs	Xavier Antoine	professeur à l'Université de Lorraine	
	Timo Betcke	professeur à l'University College London	
Examinateurs	Stéphanie Chaillat-Loseille	chargée de recherche au CNRS	
	Frédéric Hecht	professeur à Sorbonne Université	président du jury
Directeurs de thèse	Xavier Claeys	maître de conférence à Sorbonne Université	
	Frédéric Nataf	directeur de recherche au CNRS	

Keywords: numerical analysis, domain decomposition methods, Schwarz methods, boundary integral equations, boundary element method, two-level preconditioners, hierarchical matrices

Mots clés : analyse numérique, méthodes de décomposition de domaine, méthodes de Schwarz, équations intégrales de frontière, méthode des éléments finis de frontière, préconditionneurs à deux niveaux, matrices hiérarchiques

This thesis has been prepared at the following research units.

Laboratoire Jacques-Louis Lions
4 place Jussieu
75005 Paris
France
(+33) 01 44 27 42 98
Web Site http://ljll.math.upmc.fr/

Inria Paris
2 rue Simone Iff
75012 Paris
France
(+33) 01 80 49 40 00
Web Site https://www.inria.fr/centre/paris

Schwarz methods and boundary integral equations

Abstract

The objective of this thesis is to use domain decomposition methods to develop new efficient methods for high performance computing and boundary integral equations. One can think of two approaches for domain decomposition. One can make a decomposition of the original domain where the solution is sought, a *volume decomposition*, and then formulate a boundary integral equation in each subdomain with some ways of coupling them. Or we could first set up a boundary integral equation and then apply a domain decomposition of the boundary, a *surface decomposition*. In the first approach, we show that the local variant of the multi-trace formulation, which is naturally well-adapted to parallelization, has optimal properties of convergence in the case of constant coefficients in the whole domain for a geometry without junction points. This property is similar to the convergence of the optimal Schwarz method, and we actually prove an equivalence between these two methods. Numerical tests are provided to illustrate the convergence property and show the potentialities and the limits of this approach when coefficients are piecewise constants instead of constants in the whole domain. In the second approach, we present how we use the framework of the fictitious space lemma and the approach of the GenEO (Generalized Eigenproblems in the Overlap) coarse space to define several two-level preconditioners for the hypersingular operator associated with any symmetric positive definite equation. Numerical experiments are provided to show scalability in terms of iterations using the conjugate gradient method and GMRes. To be able to use Schwarz preconditioners and the boundary element method, we also need to adapt a compression method to a distributed-memory parallel framework. This led us to implement Htool, a C++ library for hierarchical matrices parallelized using MPI and OpenMP.

Keywords: numerical analysis, domain decomposition methods, Schwarz methods, boundary integral equations, boundary element method, two-level preconditioners, hierarchical matrices

---

Méthodes de Schwarz et équations intégrales de frontière

Résumé

L'objectif de cette thèse est d'utiliser des méthodes de décomposition de domaine pour mettre au point de nouvelles méthodes pour le calcul haute performance et les équations intégrales de frontière. Dans le cas des équations intégrales de frontière, on peut penser à deux approches de décomposition de domaine. Nous pouvons faire une décomposition du domaine où la solution est recherchée, une *décomposition volumique*, puis formuler une équation intégrale de frontière dans chaque sous-domaine en les couplant. Ou nous pouvons d'abord établir une équation intégrale de frontière et ensuite appliquer une décomposition de domaine à la frontière, une *décomposition surfacique*. Dans la première approche, nous montrons que la variante locale de la formulation multi-trace, naturellement bien adaptée à la parallélisation, possède des propriétés de convergence optimales dans le cas de coefficients constants dans tout le domaine pour une géométrie sans points de jonction. Cette propriété est similaire à la convergence de la méthode optimale de Schwarz, et nous prouvons en réalité une équivalence entre ces deux méthodes. Des tests numériques sont fournis pour illustrer la propriété de convergence et montrer les potentialités et les limites de cette approche lorsque les coefficients sont constants par morceaux au lieu de constants dans l'ensemble du domaine. Dans la deuxième approche, nous présentons comment nous utilisons le cadre du lemme de l'espace fictif et l'approche de l'espace grossier GenEO (Generalized Eigenproblems in the Overlap) pour définir plusieurs préconditionneurs à deux niveaux pour l'opérateur hypersingulier associé à toute équation symétrique et définie positive. Des expériences numériques sont fournies pour montrer leur extensibilité en termes d'itérations avec la méthode du gradient conjugué et GMRes. Pour pouvoir utiliser les préconditionneurs de Schwarz et la méthode des éléments finis de frontière, nous devons également adapter une méthode de compression à un environnement parallèle à mémoire distribuée. Cela nous a conduit à implémenter une bibliothèque C++ pour les matrices hiérarchiques parallélisée en utilisant MPI et OpenMP.

Mots clés : analyse numérique, méthodes de décomposition de domaine, méthodes de Schwarz, équations intégrales de frontière, méthode des éléments finis de frontière, préconditionneurs à deux niveaux, matrices hiérarchiques

---

Laboratoire Jacques-Louis Lions
4 place Jussieu – 75005 Paris – France

Remerciements

J'ai eu la chance d'aimer mon travail durant ces trois dernière années, et je le dois en très grande partie à mes deux directeurs de thèse, Xavier Claeys et Frédéric Nataf, que je remercie tout particulièrement. Ils m'ont chacun introduit à leur domaine de recherche respectif, et c'est dans l'effort de tisser des liens entre ces deux domaines que nous avons travaillé. Ils m'ont fait découvrir ce défi très représentatif de la recherche à laquelle je veux prendre part. Je veux aussi leur exprimer ma gratitude car j'ai conscience de la chance que j'ai eue d'avoir deux directeurs de thèse aussi présents et disponibles pour m'aider et me conseiller durant ce doctorat.

Je tiens aussi à remercier Xavier Antoine et Timo Betcke d'avoir accepté d'être les rapporteurs de cette thèse, ainsi que Stéphanie Chaillat et Frédéric Hecht de faire partie du jury, j'en suis très honoré.

Une grande partie des expériences numériques de cette thèse n'auraient pas été possibles sans l'aide de Pierre-Henri Tournier que je remercie chaleureusement, et je pense qu'une bonne partie du couloir 15-25 peut témoigner que nous rigolions beaucoup en travaillant ensemble ! Merci aussi à Pierre Jolivet qui a joué un rôle essentiel, c'est grâce à son aide et à sa bibliothèque HPDDM que nous avons pu mettre en place l'architecture logicielle nécessaire à tous les calculs concernant les préconditionneurs DDM. Je souhaite aussi remercier Marcella Bonazzoli pour les six semaines de collaboration intenses au CEMRACS, Julien Salomon de m'avoir invité à DD25 et Gabriele Ciaramella de m'avoir invité à Konstanz. J'espère avoir l'occasion de collaborer de nouveau avec chacun d'entre vous dans un avenir très proche !

Durant ce doctorat, j'ai aussi eu l'opportunité de rencontrer de nombreuses personnes qui m'ont marqué par leur gentillesse et leurs conseils. Je me permets donc de remercier aussi Axel Modave, Victorita Dolean, Nicole Spillane, Ryadh Haferssas, Sonia Fliss et François Alouges. J'ai aussi une pensée pour les membres présents et passés de l'équipe Inria ALPINES, notamment Laura Grigori, Olivier Tissot, Hussam Al Daas, Igor Chollet et Thibault Cimic. Je remercie aussi Euan Spence et Ivan Graham de m'avoir fait confiance et m'avoir permis d'écrire cette thèse avec l'esprit plus tranquille.

Je remercie évidemment mes deux compères durant cette thèse, Jean et Jean-François. L'expérience de cette thèse aurait été bien plus triste sans toutes les discussions, sorties, escape games, pièces de théatre, séances de sport ou de jeux de société et tailles de poirier que j'ai pu faire avec vous ! Un grand merci aussi à Bertrand, de la musique à la geekitude, nous avons beaucoup partagé et j'espère que nous continuerons ainsi longtemps ! J'ai aussi une pensée pour Anne-Françoise et toutes nos discussions en salle café, moments de décompression nécessaires à toute bonne journée de travail.

Je ne peux pas oublier de remercier mes amis de longue date, Alexandre et Mazen, pour leur soutien et tous les bons moments passés ensemble, des parties de ping-pong à de nos jours. Je tiens aussi à remercier Thibaud pour toutes nos aventures depuis les balades en kayak, et

pour m'avoir proposé de rejoindre REP+, dont je salue aussi les autres membres, Clémentine et François. Il n'y a rien de telle que la musique pour se vider l'esprit pendant une thèse et j'espère que nous aurons de nouveau l'occasion de jouer ensemble. Je remercie mes amis des Ponts pour tous les voyages, les soirées et les parties faites ensemble, Laurent, Thomas, Pierre-Loïk, Auréliane, Maud et Pierre-Adrien, ainsi que Laure pour le chemin parcouru ensemble. Merci aussi aux amis plus récents pour les échanges qui sortent de l'ordinaire et les découvertes, Alice, Chiara et Cristina. J'aimerais aussi remercier Eugenia pour m'avoir accompagné et encouragé pendant la fin de cette thèse.

Évidemment, rien de tout cela ne serait possible sans ma famille, je souhaite donc rendre hommage à mes parents, ma sœur, ma tante et mes grands-parents qui ont toujours été là pour me soutenir.

Contents

List of Figures

CHAPTER 1

Introduction

Outline of the current chapter

1.1 Introduction : version française

1.1.1 Contexte de la thèse

Les modèles mathématiques sont particulièrement utiles pour décrire des problèmes issus de la physique et de l'ingénierie tels que l'acoustique, la mécanique ou l'électromagnétisme. Ils permettent d'exprimer des phénomènes physiques dans un langage unifié et rigoureux. En particulier, les équations aux dérivées partielles relient les grandeurs physiques et leurs éventuelles variations. La résolution de telles équations permet de mieux comprendre et prédire le comportement de ces grandeurs physiques à l'erreur du modèle mathématique près, qui ne peut que simplifier la complexité de la physique.

Mais les solutions d'équations aux dérivées partielles sont rarement connues explicitement. Heureusement, l'invention et l'évolution des ordinateurs depuis le siècle dernier ont permis aux scientifiques d'inventer des méthodes numériques pour les approcher. À partir de l'analyse des équations aux dérivées partielles et du comportement de leurs solutions, les mathématiciens du domaine de l'analyse numérique étudient l'erreur et l'efficacité de ces approximations.

La résolution numérique des équations aux dérivées partielles conduit à des systèmes linéaires à résoudre par ordinateur. La performance des calculateurs est alors cruciale. Ainsi, l'amélioration de leurs performances se traduit par des simulations numériques plus rapides et la résolution de problèmes plus importants. Jusqu'au milieu des années 2000, ces améliorations provenaient de l'augmentation de la fréquences des processeurs utilisés pour les simulations. À partir du milieu du XXe siècle, la loi de Moore stipule que le nombre de transistors par microprocesseur double tous les deux ans, tandis que la règle d'échelle de Dennard [44] indique que la puissance nécessaire au fonctionnement des transistors dans un volume donné reste constante, peu importe le nombre de transistors. La combinaison de ces deux lois (également appelée loi de Koomey [108]) signifie que la performance par watt augmente au même rythme que la loi de Moore. Mais à partir du milieu des années 2000, la taille des transistors est devenue si petite que les limitations physiques ont mis fin à la règle d'échelle de Dennard, principalement à cause des fuites de courant et de la dissipation thermique, et donc l'efficacité des processeurs monocœurs a commencé à stagner.

La fin de la règle d'échelle de Dennard ayant conduit à un changement dans les architectures des ordinateurs, le calcul parallèle est devenu le nouveau paradigme. Il est aujourd'hui largement utilisé pour le calcul haute performance dans les supercalculateurs, mais aussi dans les ordinateurs personnels et les téléphones portables où les processeurs multicœurs sont aujourd'hui courants. La figure 1.1[1] illustre ces différentes tendances.

Cette évolution de l'architecture des ordinateurs a également eu une conséquence dans le développement des méthodes numériques, où le parallélisme des algorithmes est aujourd'hui indispensable à leur efficacité sur supercalculateur. Dans le cas des algorithmes de résolution de systèmes linéaires, il existe deux classes de méthodes : les méthodes directes et les méthodes itératives. D'une part, les méthodes directes sont connues pour être robustes, mais leur nombre d'opérations et leur consommation mémoire peuvent être importantes pour les grands systèmes. D'autre part, les méthodes itératives ont une faible consommation mémoire parce qu'elles se composent principalement de produits matrice-vecteur avec le système linéaire original, et peuvent donc être plus faciles à paralléliser. Mais elles sont moins robustes et peuvent ne pas converger.

Notons $\mathbf{Ax} = \mathbf{f}$ le système linéaire considéré. Une méthode pour améliorer la robustesse des méthodes itératives est de multiplier le système par un autre opérateur $\mathbf{P}$ appelé *préconditionneur*, afin que $\mathbf{PAx} = \mathbf{Pf}$ soit plus facile à résoudre, et dont l'application n'est pas trop coûteuse.

1.1.2 Méthodes de décomposition de domaine et équations intégrales de frontière

Les méthodes de décomposition de domaine regroupent un large éventail de techniques dont le point commun est principalement une stratégie de « diviser pour régner », qui est naturellement parallèle et donc bien adaptée aux ordinateurs modernes. Dans cette thèse, nous nous intéresserons aux méthodes de Schwarz, une classe particulière de méthodes de décomposition de domaine. Elles consistent à diviser le domaine de calcul en plusieurs sous-domaines dans lesquels nous résolvons des problèmes locaux tout en échangeant des informations pertinentes entre eux. Deux points de vue existent sur ces méthodes, elles peuvent être considérées comme des algorithmes itératifs où une approximation de la solution de l'équation considérée est calculée à chaque itération, ou comme des préconditionneurs pour résoudre le système linéaire original par des méthodes itératives de type Krylov (gradient conjugué, GMRes, gradient biconjugué,…). Mais dans les deux cas, les problèmes locaux sont typiquement résolus par des méthodes directes, c'est pourquoi ces méthodes sont dites hybrides.

[1] Les données ont été recueillies par Karl Rupp pour son article « 42 Years of Microprocessor Trend Data » et publié dans https://github.com/karlrupp/microprocessor-trend-data sous une license Creative Commons Attribution 4.0 International Public License

Dans cette thèse, nous cherchons à adapter les méthodes de Schwarz aux équations intégrales de frontière qui sont des reformulations des équations aux dérivées partielles usuelles. En utilisant la fonction de Green de l'équation aux dérivées partielles considérée, des opérateurs intégraux peuvent être définis pour formuler une équation intégrale dont l'inconnue est liée à la frontière du problème initial. Les principaux avantages des équations intégrales de frontière sont que le domaine de calcul comporte une dimension de moins, et que la condition à l'infini est naturellement satisfaite. Cela signifie en particulier que les problèmes en domaine non-borné sont naturellement formulés à l'aide d'équations intégrales de frontière.

Mais les équations intégrales de frontière sont limitées aux problèmes dont la fonction de Green est explicitement connue, tels que les problèmes elliptiques à coefficients constants. Par ailleurs, les opérateurs intégraux sont non-locaux, rendant l'analyse mathématique plus difficile. En pratique, cela signifie aussi que les matrices obtenues par une méthode de Galerkin sont denses, contrairement aux matrices issues de la méthode des éléments finis qui sont creuses.

Dans le cas des équations intégrales de frontières, on pourrait penser à deux approches de décomposition de domaine, comme indiqué dans l'introduction de [80]. Nous pouvons faire une décomposition du domaine où la solution est recherchée, une *décomposition volumique*, puis formuler une équation intégrale de frontière dans chaque sous-domaine en les couplant. Ou nous pouvons d'abord établir une équation intégrale de frontière et ensuite appliquer une décomposition de domaine de la frontière, une *décomposition surfacique*.

1.1.3 Résumé et contributions

Le sujet principal de cette thèse est d'établir des liens entre les méthodes de décomposition de domaine et les équations intégrales de frontières en utilisant les différents points de vue décrits dans la section précédente. Outre les défis scientifiques auxquels nous avons dû faire face, une difficulté inhérente à ce sujet est d'ordre pédagogique. Les méthodes de décomposition de domaine et les équations intégrales de frontière correspondent à deux domaines de recherche distincts. C'est pourquoi nous prendrons le temps de présenter succinctement ces deux domaines en donnant des références importantes, afin que les lecteurs de l'une ou l'autre communauté puissent lire ce document. Ce dernier est structuré en six chapitres dont le **chapitre 1** est la présente introduction.

Dans le chapitre 2, nous présentons au lecteur les équations d'intégrale de frontière avec la plupart des résultats de base.

- Nous définissons les espaces de Sobolev fractionnaires, les espaces d'énergie des équations intégrales de frontière considérées.
- Les équations aux dérivées partielles elliptiques sont introduites avec les résultats concernant leur caractère bien posé.
- Les opérateurs intégraux de frontière associés, basés sur la fonction de Green, sont présentés.
- Nous faisons le lien entre les opérateurs intégraux de frontière et le projecteur Calderón, ainsi qu'avec l'opérateur Dirichlet-to-Neumann.
- Les équations intégrales de frontière sont présentées sous plusieurs formes et les résultats sur leur caractère bien posé sont présentés.

Dans le chapitre 3, nous donnons au lecteur une introduction très succincte aux méthodes de Schwarz. Nous présentons les deux points de vue, comme algorithmes itératifs et comme préconditionneurs.

- Nous donnons une présentation historique des méthodes de Schwarz avec les principaux algorithmes itératifs.
- La méthode de Schwarz optimale est présentée et son optimalité est illustrée.
- Après avoir présenté le cadre discret lié à la méthode des éléments finis, nous présentons les outils théoriques pour étudier les préconditionneurs de Schwarz, et notamment le lemme de l'espace fictif.
- Nous l'utilisons afin d'analyser le préconditionneur additif Schwarz sans espace grossier, pour justifier de la nécessité d'un espace grossier ajoutant un minimum de communication globale entre les sous-domaines.
- Nous présentons l'espace grossier GenEO (Generalized Eigenproblem in the Overlap), qui est basé sur des problèmes aux valeurs propres généralisés locaux.

Le chapitre 4 est consacré à notre travail sur la méthode multi-trace locale. Nous considérons une décomposition volumique, dans laquelle une équation intégrale de frontière est formulée dans chaque sous-domaine. Cela permet de représenter une équation avec des coefficients constants par morceaux dans chaque sous-domaine. Il existe plusieurs formulations pour ce type de problèmes : la formulation PMCHWT (Poggio-Miller-Chang-Harrington-Wu-Tsai) [131, 23, 163] aussi appelée principe de réaction de Rumsey, la méthode BETI (Boundary Element Tearing and Interconnecting) [110] et les formulations Multi-Trace (MTF) [38]. Dans [33], des résultats intéressants ont été présentés pour Dissipative Helmholtz et l'opérateur associé à local-MTF, une variante de MTF. Les auteurs ont examiné une discrétisation particulière de local-MTF liée à une méthode de Jacobi. Ils ont prouvé que pour deux et trois domaines, celle-ci converge en un nombre fini d'itérations pour des coefficients constants dans tout le domaine et avec un choix particulier de paramètre dans la méthode. Une équivalence avec la méthode optimale de Schwarz est également présentée pour deux sous-domaines dans une dimension, où tout peut être fait de manière analytique. Nous généralisons ces résultats dans ce chapitre.

- Les opérateurs intégraux dans un contexte multi-domaines sans points de jonction sont introduits, ce qui permet de présenter la formulation PMCHWT et local-MTF.
- Nous introduisons la méthode de Jacobi de [33] dans un cadre plus général (n sous-domaines sans points de jonction et équation elliptique générique) et nous prouvons qu'elle converge en un nombre fini d'itérations pour des coefficients constants dans tout le domaine avec un choix particulier de paramètre dans la méthode. Nous donnons explicitement le nombre d'itérations pour converger en fonction de la géométrie.
- Nous prouvons également l'équivalence entre le local-MTF et la méthode de Schwarz optimale pour ce choix particulier de paramètres et des coefficients constants dans le domaine.
- Des expériences numériques sont menées pour illustrer ces résultats théoriques et pour observer ce qui se passe lorsque les hypothèses ne sont plus satisfaites, par exemple des coefficients constants par sous-domaine au lieu de constants dans tout le domaine. Ainsi, nous montrons le potentiel et les limites de cette approche.

Dans le chapitre 5, nous présentons nos travaux sur les préconditionneurs de Schwarz pour la méthode des éléments finis de frontière. Les méthodes de Schwarz sont bien connues pour les équations aux dérivées partielles, et leur application à la méthode des éléments finis de frontière en tant que préconditionneurs a également été explorée (voir [80, 156, 88]) avec un

espace grossier basé sur un problème grossier. Les stratégies élaborées jusqu'à maintenant dans la litérature semblent être des adaptations du cadre présenté dans [153, Chapitre 3]. Cela signifie qu'un problème grossier est défini pour construire l'espace grossier, de telle sorte qu'il assure un conditionnement indépendant du nombre de sous-domaines. Mais le problème grossier pertinent dépend de l'opérateur, de la dimension, etc. À notre tour, nous adaptons l'espace grossier GenEO à des équations intégrales de frontière symétriques et définies positives. Cet espace grossier a l'avantage d'être plus algébrique car il est construit à partir de problèmes aux valeurs propres généralisés. À notre connaissance, c'est le premier espace grossier spectral et adaptatif pour les éléments finis de frontière.

- Nous présentons le cadre discret associé à la méthode des éléments finis de frontière et à la décomposition surfacique.
- Plusieurs inégalités liées à la localisation des normes fractionnaires de Sobolev sont présentées.
- En utilisant le cadre du lemme de l'espace fictif avec l'approche de l'espace grossier GenEO, nous introduisons de nouveaux espaces grossiers spectraux pour l'opérateur hypersingulier.
- Nous effectuons plusieurs tests numériques pour vérifier que la méthode passe à l'échelle en termes de nombre d'itérations.
- Des perspectives pour l'extension de la méthode sont données et des tests numériques avec des préconditionneurs de Schwarz pour l'opérateur faiblement singulier et l'équation d'Helmholtz sont présentés.

Dans le chapitre 6, nous présentons notre implémentation parallèle de matrices hiérarchiques pour pouvoir utiliser des préconditionneurs de Schwarz. Une particularité de la méthode des éléments finis de frontière par rapport à la méthode des éléments finis usuelle est que les systèmes linéaires à résoudre sont denses. Cela peut s'avérer rapidement très coûteux en terme de nombres d'opérations et de stockage pour les grands systèmes, ce qui rend les techniques de compression nécessaires. Plusieurs techniques existent, *panel clustering* [78, 79], *Fast Multipole Method* (FMM) [134, 70, 42, 42, 43], *matrices hiérarchiques* ($\mathcal{H}$ et $\mathcal{H}^2$-matrices) [74, 76, 77, 17, 12, 75] *Adaptive Cross Approximation* (ACA) [11, 13, 18, 12], *Sparse Cardinal Sine Decomposition* (SCSD) [4] et *wavelet compression techniques* [5]. Nous avons utilisé des matrices hiérarchiques car leur construction a l'avantage d'être algébrique et indépendante du noyau. Pour pouvoir réaliser des expériences numériques avec les préconditionneurs de Schwarz, nous avons développé Htool, une bibliothèque C++ pour les matrices hiérarchiques parallélisée avec MPI et OpenMP. Des illustrations numériques des différents concepts présentés dans ce chapitre sont présentées à l'aide de Htool.

- Plusieurs approximations classiques de rang faible sont présentées, à savoir la SVD tronquée, l'approximation adaptative en croix entièrement pivotée et l'approximation adaptative croisée partiellement pivotée.
- Nous présentons les différentes structures hiérarchiques nécessaires pour définir une matrice hiérarchique.
- Notre mise en œuvre de ces concepts est présentée. En particulier, nous montrons comment Htool est conçue pour faciliter les ajouts et nous présentons comment la bibliothèque est parallélisée pour le processus de construction des matrices hiérarchiques et l'assemblage de l'espace grossier GenEO.

- Enfin, nous donnons quelques perspectives de développement logiciel pour la bibliothèque.

Publications : Cette thèse a donné lieu aux publications suivantes :

- Pierre Marchand, Xavier Claeys, Pierre Jolivet, Frédéric Nataf, Pierre-Henri Tournier (2019). Boundary integral multi-trace formulations and Optimised Schwarz Methods. (Submitted to Numerische Mathematik).
- Xavier Claeys, Pierre Marchand (2018). Boundary integral multi-trace formulations and Optimised Schwarz Methods. (Submitted to Computers and Mathematics with Applications).

Développement logiciel : Cette thèse a conduit à l'implémentation de Htool disponible sur la page de l'auteur[2], une bibliothèque C++ header-only pour les matrices hiérarchiques parallélisée avec MPI et OpenMP. Toutes les expériences numériques seront disponibles prochainement sur la page web de l'auteur.

[2]Aussi disponible à l'adresse suivante : https://github.com/PierreMarchand20/htool

1.2 Introduction: english version

1.2.1 Context of the thesis

Mathematical models are particularly useful for describing problems in physics and engineering such as acoustics, mechanics or electromagnetics. They allow physical phenomena to be expressed with a unified and rigorous language. In particular, partial differential equations relate physical quantities and their possible variations. Solving such equations makes it possible to better understand and predict the behaviour of these physical quantities up to the error of the mathematical model, which can only simplify the complexity of the physics.

However solutions of partial differential equations are rarely explicitly known. Fortunately, the invention and evolution of computers since the last century has allowed scientists to invent numerical methods to approximate them. From the analysis of partial differential equations and the behaviour of their solutions, mathematicians in the field of numerical analysis study the error and efficiency of these approximations.

The numerical solution to partial differential equations leads to linear systems to be solved by computers, whose performance is then crucial. Thus, improving their performance results in faster numerical simulations and solving larger problems. Until the mid-2000s, these improvements were due to the increase in the frequency of the processors used for simulations. From the mid-20th century, Moore's law stated that the number of transistors on a chip would double every two years, while Dennard scaling [44] stated that the power needed to run transistors in a unit volume stays constant, regardless of the number of transistors. The combination of these two laws (also called Koomey's law [108]) meant that performance per watt increases at the same rate as Moore's law. But from the mid-2000s, the size of the transistors became so small that physical limitations broke Dennard scaling, mainly due to current leakage and heat dissipation, and so the efficiency of single-core processors began to stagnate.

The breakdown of Dennard scaling led to a change in computer architectures, parallel computing became the new paradigm. It is now widely used in high-performance computing on supercomputers, but also on personal computers and mobile phones where multicore processors are common now. Figure 1.1[3] illustrates these different trends.

This change in computer architectures also had a consequence in the development of numerical methods, where parallelism of the algorithms is now essential for them to be efficient on supercomputers. In the case of algorithms for solving linear systems, there exist two classes of methods: direct methods and iterative methods. On the one hand, direct methods are known to be robust but their number of operations and memory consumption can be overwhelming for large systems. On the other hand, iterative methods have low memory consumption because they consist mainly of matrix-vector products with the original linear system, and may therefore be easier to parallelize. But they are less robust and might not converge.

Denoting $\mathbf{Ax} = \mathbf{f}$ the considered linear system, a method to circumvent the lack of robustness of iterative methods is to multiply the system by another operator $\mathbf{P}$ called *preconditioner*, so that $\mathbf{PAx} = \mathbf{Pf}$ is easier to solve, and whose application is not too expensive.

1.2.2 Domain decomposition methods and Boundary Element Method

Domain decomposition method is a term that encompasses a wide range of approaches whose common point is mainly a "divide and conquer" strategy, which is naturally parallel and therefore well suited to modern computers. In this thesis, we will be interested in Schwarz methods,

[3]The data has been gathered by Karl Rupp for his article "42 Years of Microprocessor Trend Data" and published in https://github.com/karlrupp/microprocessor-trend-data under a Creative Commons Attribution 4.0 International Public License

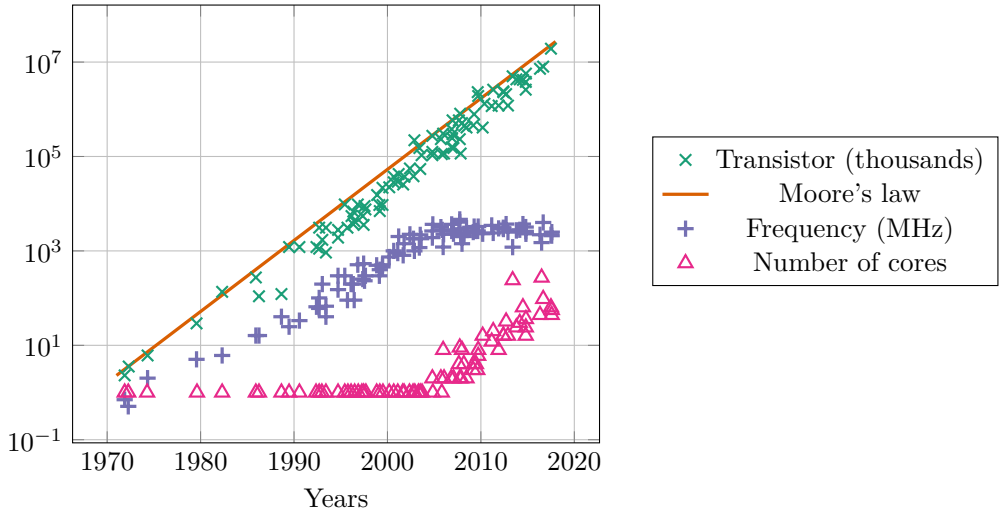


Figure 1.1 – 42 years of microprocessors trend data by Karl Rupp.

a particular class of domain decomposition methods. It consists in dividing the computational domain in many subdomains on which we solve local problems while exchanging relevant information between them. Two points of view exist on these methods, they can be seen as iterative algorithms where an approximated solution is computed at each iteration, or as preconditioners for solving the original linear system with iterative methods such as Krylov methods (conjugate gradient, GMRes, biconjugate grandient,...). But in both cases, local problems are typically solved using direct methods, hence they are called hybrid methods.

In this thesis, we seek to adapt Schwarz methods to boundary integral equations that consist in reformulations of usual partial differential equations. Using the Green function of the considered partial differential equation, integral operators can be defined to formulate an integral equation whose unknown is related to the boundary of the original problem. The main advantages of boundary integral equations are that the computational domain is one dimension smaller, and the condition at infinity is naturally satisfied. This means in particular that problems in unbounded domains are naturally formulated using boundary integral equations.

But they are restricted to problems whose Green function is explicitly known, such as elliptic problems with constant coefficients, and the integral operators are non-local making the mathematical analysis more difficult. In practice, it also means that matrices obtained by Galerkin methods are dense, contrary to matrices stemming from the finite element method which are sparse.

In the case of boundary integral equations, we could think of two approaches for domain decomposition, as pointed out in the introduction of [80]. We could make a decomposition of the original domain where the solution is sought, a *volume decomposition*, and then formulate a boundary integral equation in each subdomain with some ways of coupling them. Or we could first set up a boundary integral equation and then apply a domain decomposition of the boundary, a *surface decomposition*.

1.2.3 Summary and contributions

The main subject of this thesis is to make connections between domain decomposition methods and boundary integral equations using the different points of view described in the previous

section. In addition to the scientific challenges we had to face, an inherent difficulty in this subject is pedagogical. Domain decomposition methods and boundary integral equations correspond to two distinct fields of research. With this in mind, we will take the time to present succinctly both domains giving important references for each field, so that readers from one community or the other could still read this manuscript. The manuscript is structured in six chapters whose **Chapter 1** is the current introduction.

In Chapter 2, we introduce the reader to boundary integral equations with most of the basic results.

- We define fractional Sobolev spaces, the energy spaces of the considered boundary integral equations.
- Classically, we introduce usual elliptic partial differential equations and results about their well-posedness.
- The associated boundary integral operators based on their Green functions are presented.
- We make the connection between the boundary integral operators, and both Calderón projectors and Dirichlet-to-Neumann operator.
- Boundary integral equations in several forms are given, and results about their well-posedness are presented.

In Chapter 3, we give to the reader a very succinct introduction to Schwarz methods. We present both points of view, as iterative algorithms and as preconditioners.

- We give an historical presentation of Schwarz methods with the main iterative algorithms, alternating Schwarz method and parallel Schwarz method.
- The optimal Schwarz method is introduced and its optimality is illustrated.
- After introducing the discrete setting, we present the theoretical framework to study Schwarz preconditioners using the fictitious space lemma.
- We use it to analyse the additive Schwarz preconditioner without coarse space, to justify the need for a mechanism adding a minimum of global communication between the subdomains.
- We present the GenEO (Generalized Eigenproblem in the Overlap) coarse space, which is based on local eigenproblems.

Chapter 4 is devoted to our work on multi-trace method. We consider a volume decomposition in which, a boundary integral equation is formulated in each subdomain. It allows having an equation with piecewise constant coefficients in each subdomain. There exist several formulations for this type of problems: the PMCHWT (Poggio-Miller-Chang-Harrington-Wu-Tsai) formulation [131, 23, 163] also called Rumsey's reaction principle, the Boundary Element Tearing and Interconnecting (BETI) method [110] and Multi-Trace Formulations (MTF) [38]. In [33], interesting results have been presented for Dissipative Helmholtz and the operator associated with local-MTF, a variant of MTF. The authors considered a particular discretization of local-MTF related to a Jacobi method, and they proved that for two and three domains it converges in a finite number of iterations for constant coefficients in the whole domain, with a particular choice of parameter in the method. An explicit equivalence with the optimal Schwarz method is also presented for two subdomains in one dimension, where everything can be done analytically. We generalize these results in this chapter.

- Integral operators in a multi-subdomain setting without junction points are introduced, which allows presenting PMCHWT formulation and local-MTF.
- We introduce the Jacobi method from [33] in a more general setting (n subdomains without junction points and any elliptic equation) and we prove that it converges in a finite number of iterations for constant coefficients in the whole domain, with a particular choice of parameter in the method. We give explicitly the number of iterations to converge in function of the geometry.
- We also prove the equivalence between local-MTF and the optimal Schwarz method for this particular choice of parameter and constant coefficients.
- Numerical experiments are conducted to illustrate these theoretical results and to observe what happens when the hypotheses no longer hold, piecewise constant coefficients in each subdomain instead of constant in the whole domain for example. Thus, we show the potentialities and the limits of this approach.

In Chapter 5, we present our work on Schwarz preconditioners for the boundary element method. Schwarz methods are well-known for usual partial differential equations, and their application to boundary element method as preconditioners has also been explored (see [80, 156, 88]) with a coarse space based on a coarse problem. As far as we know, the strategies developed in the present literature are usually adaptations of the framework presented in [153, Chapter 3]. It means that a coarse problem is defined to build the coarse space such that it ensures a condition number independent of the number of subdomains. But the relevant coarse problem depends on the operator, the dimension, etc. In our turn, we adapt the GenEO coarse space to symmetric positive definite boundary integral equations. This coarse space has the advantage to be more algebraic, because it is built upon local generalized eigenproblems. As far as we know, it is the first adaptive spectral coarse space for the boundary element method.

- We introduce the discrete setting associated with the boundary element method and the surface domain decomposition.
- Several inequalities related to the localization of fractional Sobolev norms are presented.
- Using the framework of the fictitious space lemma with the approach of the GenEO coarse space, we introduce several new spectral coarse spaces for the hypersingular operator.
- We conduct several numerical tests to check the scalability of the method in terms of number of iterations.
- Perspectives to extend the method are given and numerical tests with Schwarz preconditioners for the weakly singular operator and Helmholtz are shown.

In Chapter 6, we present our parallel implementation of hierarchical matrices for using Schwarz preconditioners. A particularity of the boundary element method compared to the finite element method is that the linear systems to solve are dense. This can be quickly overwhelming in terms of storage and number of operations for large systems, making compression techniques mandatory. Several techniques exist, *panel clustering* [78, 79], *Fast Multipole Method* (FMM) [134, 70, 42, 43], *hierarchical matrices* ($\mathcal{H}$ and $\mathcal{H}^2$-matrices) [74, 76, 77, 17, 12, 75] *Adaptive Cross Approximation* (ACA) [11, 13, 18, 12], *Sparse Cardinal Sine Decomposition* (SCSD) [4] and *wavelet compression techniques* [5]. In our work, we used hierarchical matrices, which have the advantage to be algebraic and kernel independent. To be able to conduct numerical experiments with Schwarz preconditioners, we developed Htool, a C++ library for hierarchical matrices

parallelized with MPI and OpenMP. Numerical illustrations of the different concepts introduced in this chapter are provided using Htool.

- Several classical low-rank approximations are presented, namely truncated SVD, fully pivoted adaptive cross approximation and partially pivoted adaptive cross approximation.
- We present the different hierarchical structures to define a hierarchical matrix.
- Our implementation of these concepts are presented. In particular, we show how the library is designed to facilitate easy additions and we present how the library is parallelized for the building process of hierarchical matrices and the assembly of the GenEO coarse space.
- Finally, we give some software development perspectives for the library.

Publications: This thesis led to the following publications:

- Pierre Marchand, Xavier Claeys, Pierre Jolivet, Frédéric Nataf, Pierre-Henri Tournier (2019). Boundary integral multi-trace formulations and Optimised Schwarz Methods. (Submitted to Numerische Mathematik).
- Xavier Claeys, Pierre Marchand (2018). Boundary integral multi-trace formulations and Optimised Schwarz Methods. (Submitted to Computers and Mathematics with Applications).

Software development: This thesis led to the implementation of Htool[4], a C++ header-only template library for hierarchical matrices parallelized with MPI and OpenMP. All the numerical experiments should be available on the webpage of the author shortly.

[4] https://github.com/PierreMarchand20/htool

CHAPTER 2

Boundary integral equations

Outline of the current chapter

In this chapter, we introduce what are *Boundary integral equations* (BIE), which are one of the main focus in our work, and how they can be used to solve elliptic Boundary Value Problems (BVP). In Section 2.1, we recall what are Lipschitz domains and surfaces since it will be our main geometric setting. We also introduce relevant function spaces for BIE, that is to say, *fractional*

Sobolev spaces. Then, in Section 2.2 we define what we call Dirichlet and Neumann traces, and we present elliptic BVP and their properties. Finally, we introduce the associated Boundary Integral Operators (BIO) with their properties in Section 2.3 and how they can be used to define BIE in Section 2.4. We refer to [139, 132, 123] for more details.

2.1 Sobolev spaces

In this section, we define the function spaces that we will need later to introduce BIE. To do so, we first define the domain on which the function spaces will be defined. Then we introduce the relevant functional spaces, namely the *Sobolev-Slobodeckij* spaces H^s. We refer to [123, 139] for more details.

2.1.1 Lipschitz domains and surfaces

Let $\Omega \subset \mathbb{R}^d$ be an open set and $d \geq 2$ an integer, its boundary Γ is defined as follows:

$$\Gamma = \partial\Omega = \overline{\Omega} \cap (\mathbb{R}^d \setminus \Omega).$$

Then Ω is said to be a Lipschitz domain, if its boundary can be represented locally as the graph of a Lipschitz function in an appropriate orthonormal basis. More precisely, we have the following definitions:

Definition 2.1 (Lipschitz hypograph)**.** *Ω is said to be a* Lipschitz hypograph *if it can be represented as*

$$\Omega = \left\{ \mathbf{x} = (x_i)_{i=1}^d \in \mathbb{R}^d \,|\, x_d < \zeta(x') \text{ for all } x' = (x_i)_{i=1}^{d-1} \in \mathbb{R}^{d-1} \right\},$$

where $\zeta : \mathbb{R}^{d-1} \to \mathbb{R}$ is a Lipschitz function.

Definition 2.2 (Lipschitz domain, [123, Definition 3.28])**.** *Ω is said to be a* Lipschitz domain *if its boundary Γ is compact and if there exist finite families W_j and $\Omega_j \subset \mathbb{R}^d$ with $1 \leq j \leq J$ such that:*

(i) The family $\{W_j\}$ is a finite open cover of Γ, that is to say, each W_j is an open subset of $\mathbb{R}^d$ and $\Gamma \subseteq \cup_j W_j$.

(ii) Each Ω_j can be transformed to a Lipschitz hypograph by a rigid motion, i.e., by a rotation and a translation.

(iii) $W_j \cap \Omega = W_j \cap \Omega_j$ for each j.

Remarks 2.3.

- *Ω can be unbounded, even if Γ is bounded.*
- *If Ω is a Lipschitz hypograph, then*

$$\Gamma = \left\{ \mathbf{x} = (x_i)_{i=1}^d \in \mathbb{R}^d \,|\, x_d = \zeta(x') \text{ for all } x' = (x_i)_{i=1}^{d-1} \in \mathbb{R}^{d-1} \right\}$$

- *Polygons in $\mathbb{R}^2$ and polyhedrons in $\mathbb{R}^3$ are Lipschitz domains.*

2.1.2 Sobolev-Slobodeckij spaces

Let $\Omega \subseteq \mathbb{R}^d$ be a non-empty open set, we denote by $L^2(\Omega)$ the usual space of Lebesgue square integrable functions equipped with the standard norm $\|\cdot\|_{L^2(\Omega)}$. We also define the space of infinitely differentiable functions with compact support in Ω,

$$\mathcal{D}(\Omega) := \{u \in C^\infty(\Omega) \text{ with } \operatorname{supp}(u) \subseteq K \text{ for some } K, \text{ a compact subset of } \Omega\},$$

so that $\mathcal{D}^*(\Omega)$ is the space of Schwartz distributions. Let $\alpha = (\alpha_1, \dots, \alpha_d)$, $\beta = (\beta_1, \dots, \beta_d) \in \mathbb{N}^d$, be some d-tuples, and denote

$$\partial^\alpha u := \left(\frac{\partial}{\partial x_1}\right)^{\alpha_1} \dots \left(\frac{\partial}{\partial x_d}\right)^{\alpha_d} u \quad \text{and} \quad \mathbf{x}^\beta := \left(x_1^{\beta_1}, \dots, x_d^{\beta_d}\right),$$

we define *Sobolev* spaces as follows: for $0 \leq p < \infty$,

$$H^p(\Omega) := \left\{u \in L^2(\Omega) \,|\, \partial^\alpha u \in L^2(\Omega) \text{ for } |\alpha| \leq p\right\},$$

where $\partial^\alpha u$ is a distribution on Ω, and we define its norm

$$\|u\|^2_{H^p(\Omega)} := \sum_{|\alpha| \leq p} \int_\Omega |\partial^\alpha u(x)|^2 \, \mathrm{d}x.$$

For spaces of fractional order, we define the Slobodeckij semi-norm: for $0 < \mu < 1$

$$|u|^2_{\mu,\Omega} := \int_\Omega \int_\Omega \frac{|u(x) - u(y)|^2}{|x - y|^{d+2\mu}} \, \mathrm{d}x \, \mathrm{d}y,$$

so that, with $s = r + \mu > 0$ r an integer and $0 < \mu < 1$, we can define the following space, called *Sobolev-Slobodeckij* space $H^s(\Omega) := \{u \in H^r(\Omega) \,|\, |\partial^\alpha u|_{\mu,\Omega} < \infty \text{ for } |\alpha| = r\}$ equipped with the norm

$$\|u\|^2_{H^s(\Omega)} := \|u\|^2_{H^r(\Omega)} + \sum_{|\alpha| = r} |\partial^\alpha u|^2_{\mu,\Omega}.$$

Then, we define two subspaces of $H^s(\Omega)$,

$$\widetilde{H}^s(\Omega) := \overline{\mathcal{D}(\Omega)}^{H^s(\mathbb{R}^d)} \quad \text{and} \quad H^s_0(\Omega) := \overline{\mathcal{D}(\Omega)}^{H^s(\Omega)}.$$

We define $H^{-s}(\mathbb{R}^d)$ as the dual of $H^s(\mathbb{R}^d)$. Then, according to [123, Theorem 3.30], if Ω is Lipschitz, we have actually $H^s(\Omega)^* = \widetilde{H}^{-s}(\Omega)$ for $s \in \mathbb{R}$. Remark that the main difference from $H^s(\Omega)$ and $\widetilde{H}^s(\Omega)$ lies in traces at $\partial\Omega$. According to [123, Theorem 3.33 and 3.40], $H^s(\Omega) = \widetilde{H}^s(\Omega)$ for $-1/2 < s < 1/2$ since there is no notion of trace anymore. We associate the following norms to the negative order Sobolev spaces

$$\|u\|_{H^{-s}(\Omega)} := \sup_{v \in \widetilde{H}^s(\Omega) \setminus \{0\}} \frac{|\langle u, v\rangle_{H^{-s}(\Omega) \times \widetilde{H}^s(\Omega)}|}{\|v\|_{\widetilde{H}^s(\Omega)}}, \text{ for } u \in H^{-s}(\Omega)$$

$$\|w\|_{\widetilde{H}^{-s}(\Omega)} := \sup_{v \in H^s(\Omega) \setminus \{0\}} \frac{|\langle w, v\rangle_{\widetilde{H}^{-s}(\Omega) \times H^s(\Omega)}|}{\|v\|_{H^{-s}(\Omega)}}, \text{ for } w \in \widetilde{H}^s(\Omega).$$

Remark 2.4 (Notations). *According to [123, Theorem 3.33], if Ω is a Lipschitz domain and $s \geq 0$ then $\widetilde{H}^s(\Omega) = H_0^s(\Omega)$ provided $s \notin \{1/2, 3/2, 5/2, \ldots\}$. In particular, we have $H_0^1(\Omega) = \widetilde{H}^1(\Omega)$. In the literature, the notation $\widetilde{H}^{1/2}(\Omega)$ is used in the BIE community following [73] but it is more often denoted $H_{00}^{1/2}(\Omega)$ in the FEM community following [111].*

2.1.3 Sobolev spaces on the boundary

We extend the previous definitions to Sobolev spaces on boundaries $\Gamma = \partial\Omega$ and $\Gamma_0 \subsetneq \Gamma$, which corresponds to a Lipschitz curve in 2D and a Lipschitz surface in 3D. For the sake of completeness, we recall these definitions in this section.

Any Lipschitz domain Ω has a surface measure σ, and an outward unit normal ν that exists σ-almost everywhere on Γ. Using the notations from Definition 2.2 and [139, Theorem 2.7.1. (Rademacher)], we have that a lipschitz function on an open set of $\mathbb{R}^d$ is differentiable almost everywhere. In particular, we have

$$\|\nabla\zeta\|_{L^\infty(\mathbb{R}^{d-1})^{d-1}} \leq M$$

where M is a Lipschitz constant for ζ. Then, for a Lipschitz hypograph, we have

$$\mathrm{d}\sigma = \sqrt{1 + |\nabla\zeta(x')|^2}\,\mathrm{d}x' \quad \text{and} \quad \mathbf{n}(x) = \frac{(-\nabla\zeta(x'), 1)}{\sqrt{1 + |\nabla\zeta(x')|^2}},$$

for $\mathbf{x} \in \Gamma$. In particular, we deduce that $\mathbf{n} \in (L^\infty(\Gamma))^d$.

Let Ω be a Lipschitz domain, we define Sobolev-Slobodeckij spaces on $\Gamma = \partial\Omega$ exactly as for $(d-1)$-dimensional domains but using surface integral. As in [73, Section 1.3.3], we define the associated norm as

$$\|u\|_{H^s(\Gamma)}^2 := \|u\|_{L^2(\Gamma)}^2 + \int_\Gamma \int_\Gamma \frac{|u(\mathbf{x}) - u(\mathbf{y})|^2}{|\mathbf{x} - \mathbf{y}|^{d+2s}}\,\mathrm{d}\sigma(\mathbf{x})\,\mathrm{d}\sigma(\mathbf{y}),$$

for $0 < s < 1$, so that $H^s(\Gamma) := \{u \,|\, \|u\|_{H^s(\Gamma)} < \infty\}$. Since Ω is a Lipschitz domain, we could not define a similar space for $s > 1$. Similarly to $H^s(\mathbb{R}^d)$, we define $H^{-s}(\Gamma)$ as the dual of $H^s(\Gamma)$ with the norm

$$\|u\|_{H^{-s}(\Gamma)} = \sup_{v \in H^s(\Gamma), v \neq 0} \frac{|\langle u, v\rangle_{H^{-s}(\Gamma)\times H^s(\Gamma)}|}{\|v\|_{H^s(\Gamma)}}.$$

We also need to define Sobolev spaces on a part $\Gamma_0 \subsetneq \Gamma$ of the boundary, meaning that Γ_0 itself has a boundary. Let us denote

$$\begin{aligned}\mathcal{D}(\Gamma) &= \left\{u \,|\, u = U|_\Gamma \text{ for some } U \in \mathcal{D}(\mathbb{R}^d)\right\},\\ \mathcal{D}(\Gamma_0) &= \{\phi \in \mathcal{D}(\Gamma) \,|\, \mathrm{supp}(\phi) \subseteq \Gamma_0\}.\end{aligned}$$

Then, we define

$$\begin{aligned}H^s(\Gamma_0) &:= \{U|_{\Gamma_0} \,|\, U \in H^s(\Gamma)\},\\ \widetilde{H}^s(\Gamma_0) &:= \overline{\mathcal{D}(\Gamma_0)}^{H^s(\Gamma)}.\end{aligned}$$

Then, the norm of $\widetilde{H}^s(\Gamma_0)$ for $0 \leq s \leq 1$ is given by $\|u\|_{\widetilde{H}^s(\Gamma_0)} := \|\mathrm{E}_{\Gamma_0}(u)\|_{H^s(\Gamma)}$, where $\mathrm{E}_{\Gamma_0}(u)$ is the extension by zero of u.

Similarly to H^s-norms defined on $\Omega \subseteq \mathbb{R}^d$, we have the following properties for Sobolev spaces on Lipschitz open surfaces such as Γ_0

- If $\partial\Gamma = \emptyset$, then $H^s(\Gamma) = \widetilde{H}^s(\Gamma)$ for $|s| \leq 1$.
- According to [123, Theorem 3.33 and 3.40], $\widetilde{H}^s(\Gamma_0) = H^s(\Gamma_0)$ for $-1/2 < s < 1/2$.
- According to [123, Theorem 3.30], if $\Omega \subset \mathbb{R}^d$ is a Lipschitz domain, then $H^s(\Gamma_0)^* = \widetilde{H}^{-s}(\Gamma_0)$ and $\widetilde{H}^s(\Gamma_0)^* = H^{-s}(\Gamma_0)$ for $|s| \leq 1$.

We also need to define the operators of extension by zero and restriction by duality on these spaces: for $u \in H^{-1/2}(\Gamma)$

$$\langle u|_{\Gamma_0}, \psi\rangle_{H^{-1/2}(\Gamma_0)\times\widetilde{H}^{1/2}(\Gamma_0)} := \langle u, \mathrm{E}_{\Gamma_0}(\psi)\rangle_{H^{-1/2}(\Gamma)\times\widetilde{H}^{1/2}(\Gamma)} \quad \text{for } \psi \in \widetilde{H}^{1/2}(\Gamma_0),$$

and for $u \in \widetilde{H}^{-1/2}(\Gamma_0)$

$$\langle \mathrm{E}_{\Gamma_0}(u), \varphi\rangle_{\widetilde{H}^{-1/2}(\Gamma)\times H^{1/2}(\Gamma)} := \langle u, \varphi|_{\Gamma_0}\rangle_{\widetilde{H}^{-1/2}(\Gamma_0)\times H^{1/2}(\Gamma_0)} \quad \text{for } \varphi \in H^{1/2}(\Gamma).$$

For $0 < s < 1$ and $\Gamma \subseteq \partial\Omega$, we will use the dual norms:

$$\|u\|_{H^{-s}(\Gamma)} := \sup_{v\in\widetilde{H}^s(\Gamma)\setminus\{0\}} \frac{\langle u, v\rangle_{H^{-s}(\Gamma)\times\widetilde{H}^s(\Gamma)}}{\|v\|_{\widetilde{H}^s(\Gamma)}}, \text{ for } u \in H^{-s}(\Gamma)$$

$$\|w\|_{\widetilde{H}^{-s}(\Gamma)} := \sup_{v\in H^s(\Gamma)\setminus\{0\}} \frac{\langle w, v\rangle_{\widetilde{H}^{-s}(\Gamma)\times H^s(\Gamma)}}{\|v\|_{H^s(\Gamma)}}, \text{ for } w \in \widetilde{H}^{-s}(\Gamma).$$

2.1.4 Trace operators

Trace operators are a fundamental tool to build boundary integral operators. It is the generalisation of the notion of restriction for a function $u \in H^s(\Omega)$ on its boundary $\partial\Omega = \Gamma$ with $1/2 < s < 3/2$. The trace operator can also be defined for functions only locally in $H^s(\Omega)$,

$$H^s_{\mathrm{loc}}(\Omega) := \{u \in C^\infty_{\mathrm{comp}}(\Omega)^* \,|\, \varphi u \in H^s(\Omega), \forall \varphi \in C^\infty_{\mathrm{comp}}(\Omega)\},$$

where $C^\infty_{\mathrm{comp}}(\Omega) := \{u|_\Omega \,|\, u \in \mathcal{D}(\mathbb{R}^d)\}$.

Remarks 2.5. *We can do the same remarks as [139, Remark 2.6.2.].*

(i) The space $H^s_{\mathrm{loc}}(\Omega)$ contains no restriction on the growth of functions at infinity.

(ii) By choosing φ equal to one, we see that for bounded domain Ω, $H^s_{\mathrm{loc}}(\Omega)$ and $H^s(\Omega)$ coincide.

(iii) Let $\Omega \subset \mathbb{R}^d$ be a bounded domain. Then, the growth of functions from $H^s_{\mathrm{loc}}(\mathbb{R}^d \setminus \overline{\Omega})$ is not restricted towards infinity, and thus relevant function spaces need to be defined when introducing partial differential equations on unbounded domains. But notice that the growth of functions is restricted in every bounded neighborhood of $\partial\Omega$.

The following theorem gives the values of s for which the trace operator has a meaning and, it also shows that its range is given by fractional Sobolev spaces.

Theorem 2.6 (Trace theorem, [139, Theorem 2.6.8.])**.** *Let $\Omega^- \subset \mathbb{R}^d$ be a bounded Lipschitz domain with boundary $\Gamma = \partial\Omega^-$ and $\Omega^+ = \mathbb{R}^d \setminus \overline{\Omega^-}$,*

(i) For $1/2 < s < 3/2$, there exists a continuous, linear trace operator $\gamma_D : H^s_{\mathrm{loc}}(\mathbb{R}^d) \to H^{s-1/2}(\Gamma)$ with $\gamma_D\varphi = \varphi|_\Gamma$ for all $\varphi \in \mathcal{D}(\mathbb{R}^d)$.

(ii) For $1/2 < s < 3/2$ and $l \in \{-, +\}$, there exists one-sided, continuous, linear trace operators $\gamma^l_D : H^s_{\mathrm{loc}}(\Omega^l) \to H^{s-1/2}(\Gamma)$ with $\gamma^l_D\varphi = \varphi|_\Gamma$ for all $\varphi \in \mathcal{D}(\overline{\Omega^l})$ and $\gamma^+_D\varphi = \gamma^-_D\varphi = \gamma_D\varphi$ almost everywhere for all $u \in H^s_{\mathrm{loc}}(\mathbb{R}^d)$.

When it is relevant, we denote γ^Ω_D the Dirichlet trace taken from the *interior* of Ω.

2.2 Elliptic boundary value problems

We introduce here the model problems we want to solve for Ω, a bounded Lipschitz domain. We refer to [139, 132] for more details.

2.2.1 Model problem

Let us define a generic differential operator L as follows

$$\mathrm{L}(u) := -\operatorname{div}(\mathbf{A}\nabla u) + 2\mathbf{b}\cdot\nabla u + cu, \tag{2.1}$$

where $\mathbf{A} \in \mathbb{R}^{d\times d}$ is positive definite, $\mathbf{b} \in \mathbb{R}^d$ and $c \in \mathbb{R}$. This generic differential operator coincides with several well-known partial differential operators:

(i) With $\mathbf{A} = \mathbf{I}_{\mathrm{d}}$, $\mathbf{b} = \mathbf{0}$ and $c = 0$, we obtain the Laplace operator $\mathrm{L} = -\Delta$,

(ii) With $\mathbf{A} = \mathbf{I}_{\mathrm{d}}$, $\mathbf{b} = \mathbf{0}$ and $c = -k^2$, we obtain the Helmholtz operator $\mathrm{L} = -\Delta - k^2$ where $k > 0$ is usually called the *wave number*.

We denote the bilinear form associated with the differential operator defined in Equation (2.1)

$$a(u, v) := \int_\Omega \mathbf{A}\nabla u \cdot \overline{\nabla v} + 2(\mathbf{b}\cdot\nabla u)\overline{v} + cu\overline{v}\,\mathrm{d}\mathbf{x}.$$

We also need to define the *formal adjoint* of L

$$\mathrm{L}^*(v) := -\operatorname{div}(\mathbf{A}\nabla v) - 2\mathbf{b}\cdot\nabla v + cv,$$

2.2.2 Conormal derivative and Green's formulas

Elliptic boundary problems usually consists of a differential operator such as L defined in Equation (2.1) and associated boundary conditions. These boundary conditions usually gives the value of the trace or the normal derivative of the unknown function. We defined the trace operator in Section 2.1.4, and we now define the conormal derivative. For $u \in H^2(\Omega)$, we have $\mathbf{A}\nabla u \in H^1(\Omega)^d$ so that we can define the *conormal derivative*

$$\gamma^\Omega_N(u) := \mathbf{A}\mathbf{n}_\Omega \cdot \gamma^\Omega_D\nabla u \in L^2(\Gamma), \tag{2.2}$$

where $\mathbf{n}_\Omega$ is the outer normal relative to Ω. For $\mathbf{A} = \mathbf{I}_{\mathrm{d}}$, notice that γ^Ω_N is exactly the usual normal derivative. Then, using Gauss' theorem (cf. [139, Theorem 2.7.3.]), we obtain the following Green's first formulas [139, Theorem 2.7.4]

- for $u \in H^2(\Omega)$ and $v \in H^1(\Omega)$,

$$a(u,v) = (\mathrm{L}(u), v)_{L^2(\Omega)} + \int_\Gamma \gamma_N^\Omega(u)\overline{\gamma_D^\Omega(v)}\,\mathrm{d}\sigma(\mathbf{x}) \tag{2.3}$$

- for $u \in H^1(\Omega)$ and $v \in H^2(\Omega)$,

$$a(u,v) = (u, \mathrm{L}^*(v))_{L^2(\Omega)} + \int_\Gamma \gamma_D^\Omega(u)\overline{\widetilde{\gamma}_N^\Omega(v)}\,\mathrm{d}\sigma(\mathbf{x}) \tag{2.4}$$

where $\widetilde{\gamma}_N^\Omega(v) := \gamma_N^\Omega(v) + 2\mathbf{b}\cdot\mathbf{n}_\Omega\gamma_D^\Omega(v)$ is the *modified* conormal derivative. More generally, we can extend the definition of the conormal derivative to the space

$$H^1_{\mathrm{L}}(\Omega) := \{u \in H^1_{\mathrm{loc}}(\Omega) \mid \mathrm{L}(u) \in L^2(\Omega)\},$$

using the following lemma

Lemma 2.7 ([123, Lemma 4.3])**.** *Suppose that Ω is a Lipschitz domain. If $u \in H^1(\Omega)$ and $f \in \widetilde{H}^{-1}(\Omega)$ satisfy*

$$\mathrm{L}(u) = f \quad \text{on } \Omega,$$

then there exists $g \in H^{-1/2}(\Omega)$ such that

$$a(u,v) = \langle f, v\rangle_{\widetilde{H}^{-1}(\Omega)\times H^1(\Omega)} + \langle g, \gamma_D v\rangle_{H^{-1/2}(\Gamma)\times H^{1/2}(\Gamma)} \quad \text{for } v \in H^1(\Omega).$$

Furthermore, g is uniquely determined by u and f, and we have

$$\|g\|_{H^{-1/2}(\Gamma)} \le C(\|u\|_{H^1(\Omega)} + \|f\|_{\widetilde{H}^{-1}(\Omega)}).$$

We define the *weak conormal derivative* $\gamma_N^\Omega(u) := g$ for $u \in H^1_{\mathrm{L}}(\Omega)$ and g from Lemma 2.7. According to this lemma, $\gamma_N^\Omega : H^1_{\mathrm{L}}(\Omega) \to H^{-1/2}(\Gamma)$ is a bounded linear operator and it extends the relation given in Equation (2.3) for $u \in H^1_{\mathrm{L}}(\Omega)$. Noticing that $H^1_{\mathrm{L}}(\Omega) = H^1_{\mathrm{L}^*}(\Omega)$, we can extend similarly the definition of the modified conormal derivative to $\widetilde{\gamma_N}^\Omega : H^1_{\mathrm{L}}(\Omega) \to H^{-1/2}(\Gamma)$ so that it also extends the relation given Equation (2.4) for $v \in H^1_{\mathrm{L}}(\Omega)$. Remark that the definition of the conormal derivative only depends on the *principal part* of L i.e., $-\operatorname{div}(\mathbf{A}\nabla\cdot)$. Indeed, we have

$$\langle g, \gamma_D v\rangle_{H^{-1/2}(\Gamma)\times H^{1/2}(\Gamma)} = \langle f, v\rangle_{\widetilde{H}^{-1}(\Omega)\times H^1(\Omega)} - a(u,v) = \langle \mathrm{L}(u), v\rangle_{H^{-1/2}(\Gamma)\times H^{1/2}(\Gamma)} - a(u,v),$$

and the sum of the terms of lower order in the last expression is equal to zero.

According to Theorem 2.6 and the previous definition, the operator γ_D^Ω (resp. γ_N^Ω) continuously maps $H^1_{\mathrm{loc}}(\Omega)$ to $H^{1/2}(\partial\Omega)$ (resp. $H^1_{\mathrm{L}}(\Omega)$ to $H^{-1/2}(\partial\Omega)$). In these definitions, the trace is taken *from the interior* of Ω. We shall denote trace operators $\gamma_c^\Omega(\phi) := (\gamma_{D,c}^\Omega(\phi), \gamma_{N,c}^\Omega(\phi))$ defined as $\gamma^\Omega(\phi) = (\gamma_D^\Omega(\phi), \gamma_N^\Omega(\phi))$ except that the traces are taken from the exterior.

This allows us to define the jump and the mean value of these traces as follows:

$$[\gamma^\Omega] := \gamma^\Omega - \gamma_c^\Omega \quad \text{and} \quad \{\gamma^\Omega\} := \frac{\gamma^\Omega + \gamma_c^\Omega}{2}. \tag{2.5}$$

We will drop the subscript referring to Ω when there is no ambiguity, which will be mostly the

case in this chapter.

2.2.3 Boundary value problems

Let us suppose that the boundary of Ω is split into two disjoint parts $\partial\Omega = \overline{\Gamma}_D \cup \overline{\Gamma}_N$ with $\Gamma_D \cap \Gamma_N = \emptyset$. We can define the following boundary value problem (BVP):

$$\begin{cases} \mathrm{L}(u) = f & \text{in } \mathcal{D}^*(\Gamma) \\ \gamma_D(u) = g_D & \text{in } H^{1/2}(\Gamma) \\ \gamma_N(u) = g_N & \text{in } H^{-1/2}(\Gamma) \end{cases} \tag{2.6}$$

where $f \in \widetilde{H}^{-1}(\Omega)$, $g_D \in H^{1/2}(\Omega)$ is called *Dirichlet data* and $g_N \in H^{-1/2}(\Omega)$ is called *Neumann data*. If $\Gamma_D = \Gamma$, the BVP (2.6) is called *first* or *Dirichlet BVP*, and if $\Gamma_N = \Gamma$, *second* or *Neumann BVP*. To study the well-posedness of this BVP, we introduce the weak formulation of BVP (2.6): Find $u \in H^1(\Omega)$ with $\gamma_D(u) = g_D$ on Γ_D such that

$$a(u,v) = \langle f, v\rangle_{\widetilde{H}^{-1}(\Omega)\times H^1(\Omega)} + \langle g_N, \gamma_D(v)\rangle_{H^{-1/2}(\Gamma_N)\times\widetilde{H}^{1/2}(\Gamma_N)} \quad \forall v \in H^1_D(\Omega) \tag{2.7}$$

where

$$H^1_D(\Omega) := \left\{ v \in H^1(\Omega) \,|\, \gamma_D(v)|_{\Gamma_D} = 0 \right\}.$$

Notice that, since $v \in H^1_D(\Omega)$ in Equation (2.7), $\gamma_D(v)|_{\Gamma_N} \in \widetilde{H}^{1/2}(\Gamma_N)$ so that the second term in right-hand side of Equation (2.7) makes sense.

Remark 2.8. *Exterior problems can be formulated in the same way, but using functional spaces that take into account decay conditions at infinity. For example:*

- *For $c \geq 0$ and $d = 3$, $|u(\mathbf{x})| = \mathcal{O}(|\mathbf{x}|^{-1})$ for $|\mathbf{x}| \to \infty$,*
- *For the Helmholtz equation, we have the* Sommerfeld's radiation conditions

$$\left.\begin{aligned} |u(\mathbf{x})| &= \mathcal{O}(|\mathbf{x}|^{-\frac{d-1}{2}}) \\ \left|\frac{\partial u}{\partial r}(\mathbf{x}) - iku(\mathbf{x})\right| &= o(|\mathbf{x}|^{-\frac{d-1}{2}}) \end{aligned}\right\} \quad \text{for } |\mathbf{x}| \to \infty,$$

  where $r = |\mathbf{x}|$ and $\frac{\partial u}{\partial r} = \frac{\mathbf{x}}{|\mathbf{x}|}\cdot\nabla u$. To lighten the presentation, we do not introduce all the different functions spaces and we refer to [139, Section 2.9.2.4] for more details.

2.2.4 Well-posedness

We define the main hypotheses on the bilinear form a that will be used to prove the well-posedness of Equation (2.7).

Definition 2.9 (Strong ellipticity)**.** *Let $a : H \times H \to \mathbb{C}$ be a sesquilinear form. We say that a is strongly H-elliptic if*

$$\mathrm{Re}(\sigma a(u,u)) \geq c\|u\|^2_H \quad \forall u \in H,$$

where $c > 0$ and $\sigma \in \mathbb{C}$.

Definition 2.10 (Ellipticity). *Let $a : H \times H \to \mathbb{C}$ be a sesquilinear form. We say that a is H-elliptic if*

$$|a(u,u)| \geq c\|u\|_H^2 \quad \forall u \in H,$$

with $c > 0$.

Remark 2.11 (Terminology). *We followed the terminology from [102] and we make the same remark as [102, Remark 5.2.1]: in the literature H-ellipticity may refer to the stronger condition of strong H-ellipticity (see [139]). But remark that strong H-ellipticity implies H-ellipticity (cf. [139, Remark 2.1.50.3]).*

Definition 2.12 (Coercivity). *Let H and U be two Hilbert spaces such that $H \subset U \subset H^*$ with the continuous and dense embedding $H \subset U$. Let $a : H \times H \to \mathbb{C}$ be a sesquilinear form. We say that a is H-coercive if*

$$\mathrm{Re}(\sigma a(u,u)) \geq c\|u\|_H^2 - C_U\|u\|_U^2 \quad \forall u \in H,$$

where $c > 0$, $C_U > 0$ and $\sigma \in \mathbb{C}$.

Typically, we will have $U = L^2(\Gamma)$ and $H = H^1(\Gamma)$. Let us recall two important tools to prove well-posedness.

Theorem 2.13 (Lax-Milgram Theorem, [139, Lemma 2.1.51]). *Let H be a Hilbert space, $a : H \times H \to \mathbb{C}$ a sesquilinear form and $b : H \to \mathbb{C}$ a continuous bilinear form. If a is H-elliptic and continuous, i.e.*

$$|a(u,v)| \leq C\|u\|_H\|v\|_H \quad \forall u, v \in H,$$

then, there exists a unique $u \in H$ that solves the abstract variational problem: find $u \in H$ such that

$$a(u,v) = b(v), \quad \forall v \in H.$$

Theorem 2.14 (Fredholm's alternative, [139, Theorem 2.1.60]). *Let $H \subset U \subset H^*$ with a compact and dense embedding between H and U. Let the sesquilinear form $a(\cdot,\cdot) : H \times H \to \mathbb{C}$ be H-coercive and we denote $A : H \to H^*$ its associated operator defined by $\langle Au, v\rangle_{H^* \times H} := a(u,v)$.*

Then, we have for all $\lambda \in \mathbb{C}$ either $(A - \lambda I)^{-1} : H^* \to H$ is a bounded operator or λ is an eigenvalue of A. In the first case, the variational problem: Find $u \in H$ such that

$$a(u,v) - \lambda(u,v)_U = \langle f, v\rangle_{H^* \times H}, \quad \forall v \in H,$$

has a unique solution for all $f \in H^*$. In the second case, the eigenspace $E(\lambda) = \ker(A - \lambda I) \neq \{0\}$ is finite dimensional.

The following lemmas show that the bilinear form a defined in Equation (2.7) is continuous and $H^1(\Omega)$-coercive.

Lemma 2.15 ([139, Lemma 2.10.1]). *The sesquilinear form a defined in Equation (2.7) is coercive on $H^1(\Omega)$, i.e.*

$$\mathrm{Re}(a(u,u)) \geq C_1\|u\|_{H^1(\Omega)}^2 - C_2\|u\|_{L^2(\Omega)}^2, \quad \forall u \in H^1(\Omega).$$

Notice that it is also true for every subspace of $H^1(\Omega)$ such as $H^1_D(\Omega)$. Besides, under certain conditions, the bilinear form a defined Equation (2.7) is also H^1-elliptic.

Lemma 2.16 ([139, Corollary 2.10.2])**.** *The bilinear form a defined in Equation (2.7) is*

- *$H^1(\Omega)$-elliptic, if $a_{\min} c > |\mathbf{b}|^2$,*
- *$H^1_D(\Omega)$-elliptic, if $|\Gamma_D| > 0$ and $\mathbf{b} = \mathbf{0}$ and $c = 0$*
- *elliptic on $H^1(\Omega)/\mathbb{C}$, if $\mathbf{b} = \mathbf{0}$ and $c = 0$,*

where $a_{\min}$ is the smallest eigenvalue of $\mathbf{A}$.

Using the previous lemmas, we deduce the following theorem concerning the well-posedness of Equation (2.7).

Theorem 2.17 ([139, Theorems 2.10.4, 2.10.5 and 2.10.6])**.** *Let us consider the variational formulation given in Equation (2.7) with $g_D \in H^{1/2}(\Gamma)$, $g_N \in H^{-1/2}(\Gamma)$ and $f \in (H^1_D(\Omega))^*$. The Fredholm alternative is applicable: either, for every boundary conditions and right-hand side, Equation (2.7) admits a unique solution $u \in H^1_D(\Omega)$ that depends continuously on the right-hand side, or zero is an eigenvalue of the operator associated with $a(\cdot,\cdot)$ that corresponds to a finite-dimensional eigenspace.*

Furthermore, we have

- *if $a_{\min} c > |\mathbf{b}|^2$, the first case applies in the Fredholm's alternative,*
- *if $c = 0$, $|\mathbf{b}| = \mathbf{0}$ and $|\Gamma_D| > 0$, the first case applies in the Fredholm's alternative,*
- *if $c = 0$, $|\mathbf{b}| = \mathbf{0}$ and $|\Gamma_D| = 0$, the first case applies in the Fredholm's alternative, but with $H = H^1(\Omega)/\mathbb{C}$. In other words, the solution is unique up to a constant function.*

Proof. The fact that we can apply Fredholm's alternative is a direct application of Theorem 2.14 and Lemma 2.15. Then, using the fact that a can be elliptic under certain conditions according to Lemma 2.16 shows that a is in fact in the first case of the Fredholm's alternative. □

In particular, we deduce that the Laplace equation is well-posed as soon as $|\Gamma_D| > 0$ and the Helmholtz equation is well-posed as soon as k^2 is not an eigenvalue of the Laplacian operator. Again, these results can be adapted to exterior problems considering the relevant functions spaces, in particular weighted Sobolev spaces for the Laplace and Helmholtz equation to control the decay of the considered functions at infinity.

2.3 Boundary integral operators

The main focus of our work is the study of *Boundary Integral Equations* (BIE) and how to solve them numerically. We introduce here *Boundary Integral Operators* (BIO) on which BIE are built on. We refer to [139, 132] for more details.

2.3.1 Representation formula and surface potentials

We set $v := c + \|\mathbf{b}\|^2_{\mathbf{A}}$ with $\|\mathbf{x}\|^2_{\mathbf{A}} := \mathbf{x}^T\mathbf{A}\mathbf{x}$ for every $\mathbf{x} \in \mathbb{R}^d$, and $\lambda = \sqrt{v}$ if $v > 0$ and $\lambda = -i\sqrt{|v|}$ otherwise. To define the relevant BIO of the equation $\mathrm{L}(u)(\mathbf{x}) = 0$ for $\mathbf{x} \in \Omega \subset \mathbb{R}^d$, a Lipschitz

domain, we need the fundamental solution $\mathrm{L}(G_\mathrm{L})(\mathbf{x}) = \delta(x)$ in the sense of distributions, where $\delta(x)$ is the Dirac delta function:

$$G_\mathrm{L}(\mathbf{x}) = \begin{cases} \dfrac{e^{\langle \mathbf{b},\mathbf{x}\rangle_\mathbf{A}}}{2\pi\sqrt{\det \mathbf{A}}} \ln\left(\dfrac{1}{\|\mathbf{x}\|_\mathbf{A}}\right) & \text{for } d = 2 \text{ and } \lambda = 0, \\ \dfrac{e^{\langle \mathbf{b},\mathbf{x}\rangle_\mathbf{A}}}{4\sqrt{\det \mathbf{A}}} i H_0^{(1)}(i\lambda\|\mathbf{x}\|_\mathbf{A}) & \text{for } d = 2 \text{ and } \lambda \neq 0, \\ \dfrac{1}{4\pi\sqrt{\det \mathbf{A}}} \dfrac{e^{\langle \mathbf{b},\mathbf{x}\rangle_\mathbf{A} - \lambda\|\mathbf{x}\|_\mathbf{A}}}{\|\mathbf{x}\|_\mathbf{A}} & \text{for } d = 3, \end{cases} \tag{2.8}$$

where $H_0^{(1)}$ stands for the Hankel function of order zero and of first kind (see [130, Section 10.2]). Remark that the function G_L is singular for $\mathbf{x} = \mathbf{0}$ and analytic for $\mathbf{x} \neq \mathbf{0}$.

With the fundamental solution, we can define the two following operators:

- *Single Layer Potential*

$$\mathrm{SL}_\mathrm{L}(q)(\mathbf{x}) := \int_\Gamma G_\mathrm{L}(\mathbf{x} - \mathbf{y}) q(\mathbf{y}) \, \mathrm{d}\sigma(\mathbf{y}), \quad \forall \mathbf{x} \in \mathbb{R}^d \setminus \Gamma,$$

- *Double Layer Potential*[1]

$$\mathrm{DL}_\mathrm{L}(v)(\mathbf{x}) := \int_\Gamma (\widetilde{\gamma_N} G_\mathrm{L})(\mathbf{x} - \mathbf{y}) v(\mathbf{y}) \, \mathrm{d}\sigma(\mathbf{y}), \quad \forall \mathbf{x} \in \mathbb{R}^d \setminus \Gamma,$$

for all $(v, q) \in H^{1/2}(\Gamma) \times H^{-1/2}(\Gamma)$. These two operators have the very interesting property to produce solutions of $\mathrm{L}(u) = 0$ in $\mathbb{R}^d \setminus \Gamma$ according to the following theorem

Theorem 2.18 ([139, Theorem 3.1.1])**.** *Let $v \in L^1(\Gamma)$,*

- *We have $(\mathrm{L} \circ \mathrm{SL}_\mathrm{L})(u)(\mathbf{x}) = (\mathrm{L} \circ \mathrm{DL}_\mathrm{L})(u)(\mathbf{x}) = 0$ for every $\mathbf{x} \in \mathbb{R}^d \setminus \Gamma$*
- *The functions $\mathrm{SL}_\mathrm{L}(v)$ and $\mathrm{DL}_\mathrm{L}(v)$ are infinitely differentiable in $\mathbb{R}^d \setminus \Gamma$.*

Besides, the functions produced by these operators naturally satisfy the necessary conditions at infinity (see Remark 2.8). It means that to solve the BVP (2.6) with $f = 0$, we can use the *ansatz* $\mathrm{SL}_\mathrm{L}(q)$ and $\mathrm{DL}_\mathrm{L}(v)$. Thus, it remains to find a trace v such that these ansatz also satisfy the boundary conditions given on Γ.

More precisely, we have the *representation formula*

Theorem 2.19 (Representation formula or Green's thrid formula, [139, Theorem 3.1.6])**.** *Let Ω be a bounded Lipschitz domain. For any function $u \in H^1_\mathrm{L}(\mathbb{R}^d \setminus \Gamma)$ where*

$$H^1_\mathrm{L}(\mathbb{R}^d \setminus \Gamma) := \{u \in L^2(\mathbb{R}^d) \,|\, u|_\Omega \in H^1_\mathrm{L}(\Omega) \text{ and } u|_{\mathbb{R}^d \setminus \overline{\Omega}} \in H^1_\mathrm{L}(\mathbb{R}^d \setminus \overline{\Omega})\},$$

with compact support and $\mathrm{L}(u) = 0$ in Ω and $\mathbb{R}^d \setminus \overline{\Omega}$[2]

$$u = \mathrm{SL}_\mathrm{L}([\gamma_N(u)]) + \mathrm{DL}_\mathrm{L}([\gamma_D(u)]) := \mathrm{G}_\mathrm{L}([\gamma(u)]),$$

[1] Usually the kernel is denoted $\widetilde{\gamma_{N,y}} G(\mathbf{x} - \mathbf{y})$, where the subscript $\mathbf{y}$ indicates that the modified conormal derivative is applied to the $\mathbf{y}$-variable. But here, we already applied the conormal derivative so that our definition is the same as in [139, p102] multiplied by -1. This allows us to only have plus signs in the representation formula and afterward.

[2] Notice that we also do not have the same definition of the normal $\mathbf{n}$ given in Equation (2.2) and of the Dirichlet and Neumann jumps given in Equation (2.5) as in [139], so that the signs are different.

where $\mathrm{G_L} : H^{1/2}(\Gamma) \times H^{-1/2}(\Gamma) \to H^1_{\mathrm{L}}(\mathbb{R}^d \setminus \Gamma)$ *such that* $\mathrm{G_L}(v,q) = \mathrm{SL_L}(q) + \mathrm{DL_L}(v)$ *for every* $(v,q) \in H^{1/2}(\Gamma) \times H^{-1/2}(\Gamma)$.

This last theorem can be extended to functions with unbounded support in 3D for the Laplace equation [139, Theorem 3.1.12.], Helmholtz equation [139, Theorem 3.1.13.] and if $a_{\min}c > |\mathbf{b}|^2$ [139, Theorem 3.1.11]. In particular, we have the following corollary

Corollary 2.20. *Let* $u \in H^1_{\mathrm{loc}}(\Omega)$ *satisfy the equation* $\mathrm{L}(u) = 0$ *in* Ω. *If* Ω *is unbounded, we also make the additionally hypothesis that* $d = 3$, L *corresponds to the Laplace equation or Helmholtz equation, or* $a_{\min}c > |\mathbf{b}|^2$ *and* u *satisfy the appropriate conditions at infinity. Then we have the representation formula*

$$\mathrm{G_L}(\gamma(u))(\mathbf{x}) = \begin{cases} u(\mathbf{x}) & \text{for } \mathbf{x} \in \Omega, \\ 0 & \text{for } \mathbf{x} \in \mathbb{R}^d \setminus \overline{\Omega}. \end{cases}$$

Similarly, if $v \in H^1_{\mathrm{loc}}(\mathbb{R}^d \setminus \overline{\Omega})$ *satisfies* $\mathrm{L}(u) = 0$ *in* $\mathbb{R}^d \setminus \overline{\Omega}$ *(and the same type of hypothesis as previously if* Ω *is bounded), then we have*

$$\mathrm{G_L}(\gamma_c(v))(\mathbf{x}) = \begin{cases} -v(\mathbf{x}) & \text{for } \mathbf{x} \in \mathbb{R}^d \setminus \overline{\Omega}, \\ 0 & \text{for } \mathbf{x} \in \Omega. \end{cases}$$

Proof. If Ω is bounded, we use Theorem 2.19 and we extend u by zero in $\mathbb{R}^d \setminus \overline{\Omega}$.

If Ω is unbounded, Theorem 2.19 can be extended to functions with unbounded support in 3D for Laplace equation, Helmholtz equation and $a_{\min}c > |\mathbf{b}|^2$ according to [139, Theorem 3.1.11, 3.1.12 and 3.1.13]. In these cases, we do as previously, extending u by zero in $\mathbb{R}^d \setminus \overline{\Omega}$. The same reasoning can be applied to the case where $v \in H^1_{\mathrm{loc}}(\mathbb{R}^d \setminus \overline{\Omega})$. □

Theorem 2.21 (Mapping properties, [41, Theorem 1] or [139, Theorem 3.1.16])**.** *Let* Ω *be a Lipschitz domain. The single and double layer potentials are continuous with the following mappings:*

$$\mathrm{SL_L} : H^{-1/2}(\Gamma) \to H^1_{\mathrm{loc}}(\mathbb{R}^d), \quad \text{and} \quad \mathrm{DL_L} : H^{1/2}(\Gamma) \to H^1_{\mathrm{L}}(\mathbb{R}^d \setminus \Gamma).$$

In particular, $\mathrm{G_L} : H^{1/2}(\Gamma) \times H^{-1/2}(\Gamma) \to H^1_{\mathrm{L}}(\mathbb{R}^d \setminus \Gamma)$ is continuous. We also have the following property on how the jumps of these potentials occur across the interface Γ

Theorem 2.22 (Jump relations, [139, p. 3.3.1.])**.** *Let* Ω *be a Lipschitz domain. We have for every* $q \in H^{-1/2}(\Gamma)$ *and* $v \in H^{1/2}(\Gamma)$

$$\begin{aligned} &[(\gamma_D \circ \mathrm{SL_L})(q)] = 0, \quad [(\gamma_D \circ \mathrm{DL_L})(v)] = v, \quad \text{in } H^{1/2}(\Gamma), \\ &[(\gamma_N \circ \mathrm{SL_L})(q)] = q, \quad [(\gamma_N \circ \mathrm{DL_L})(v)] = 0, \quad \text{in } H^{-1/2}(\Gamma). \end{aligned}$$

These relations can be rewritten in a more compact form as $[\gamma] \circ \mathrm{G_L} = \mathrm{I_d}$ *on* $H^{1/2}(\Gamma) \times H^{-1/2}(\Gamma)$.

In other words, the single layer potential $\mathrm{SL_L}$ (resp. the double layer potential $\mathrm{DL_L}$) applied to $q \in H^{-1/2}(\Gamma)$ (resp. $v \in H^{1/2}(\Gamma)$) produces a solution of the equation $\mathrm{L}(u) = 0$ in $\mathbb{R}^d \setminus \Gamma$ and the appropriate conditions at infinity, with a continuous Dirichlet trace and a jump equal to q for his Neumann trace (resp. with a continuous Neumann trace and a jump equal to v for his Dirichlet trace).

2.3.2 Definitions and their properties

We define the following *Boundary Integral Operators* (BIO):

- *Weakly singular operator* or *single layer potential*

$$\mathrm{V_L}(q) := (\gamma_D \circ \mathrm{SL_L})(q), \quad \forall q \in H^{-1/2}(\Gamma),$$

- *Double layer potential*

$$\left(\frac{1}{2}\,\mathrm{I_d} + \mathrm{K_L}\right)(v) := (\gamma_D \circ \mathrm{DL_L})(v), \quad \forall v \in H^{1/2}(\Gamma),$$

- *Adjoint double layer potential*

$$\left(\frac{1}{2}\,\mathrm{I_d} + \mathrm{K_L'}\right)(q) := (\gamma_N \circ \mathrm{SL_L})(q), \quad \forall q \in H^{-1/2}(\Gamma),$$

- *Hypersingular integral operator*

$$\mathrm{W_L}(v) := (\gamma_N \circ \mathrm{DL_L})(v), \quad \forall v \in H^{1/2}(\Gamma),$$

  where $\mathrm{I_d}$ denotes the identity operator.

We give a few interesting properties about these BIO. First, they are all linear and continuous operators according to the following theorem

Theorem 2.23 (Mapping properties, [139, Theorem 3.1.16])**.** *Let Ω be a Lipschitz domain, the BIO defined previously are continuous for the following mappings:*

- $\mathrm{V_L} : H^{-1/2}(\Gamma) \to H^{1/2}(\Gamma)$,
- $\frac{1}{2}\,\mathrm{I_d} + \mathrm{K_L} : H^{1/2}(\Gamma) \to H^{1/2}(\Gamma)$,
- $\frac{1}{2}\,\mathrm{I_d} + \mathrm{K_L'} : H^{-1/2}(\Gamma) \to H^{-1/2}(\Gamma)$,
- $\mathrm{W_L} : H^{1/2}(\Gamma) \to H^{-1/2}(\Gamma)$.

Proof. It is a direct application of Theorem 2.21 with the continuous properties of γ_D and γ_N, see Theorem 2.6 and Section 2.2.2. □

Then, we have an explicit expression of these operators for the differential operator L defined in Equation (2.1) (and others actually, elasticity problems for example). It is because we have these explicit expressions that we can discretize numerically the BIE that we will derive later.

- For $q \in L^\infty(\Gamma)$, according to [139, p. 3.3.5.], we have

$$\mathrm{V_L}(q) = \int_\Gamma G_\mathrm{L}(\mathbf{x} - \mathbf{y}) q(\mathbf{y})\, \mathrm{d}\sigma(\mathbf{y}), \quad \forall \mathbf{x} \in \mathbb{R}^d$$

- For $v, q \in C^1_{\mathrm{pw}}$, and if we assume that Γ is C^2_{pw}, according to [139, Corollaries 3.3.14. and 3.3.15.], we have

$$\left(\frac{\mathrm{I_d}}{2} + \mathrm{K_L}\right)(v)(\mathbf{x}) = \frac{v(\mathbf{x})}{2} + \int_\Gamma (\widetilde{\gamma_N} \circ G_\mathrm{L})(\mathbf{x}-\mathbf{y}) v(\mathbf{y}) d\sigma(\mathbf{y}),$$
$$\left(\frac{\mathrm{I_d}}{2} + \mathrm{K'_L}\right)(q)(\mathbf{x}) = \frac{q(\mathbf{x})}{2} + \int_\Gamma (\gamma_N \circ G_\mathrm{L})(\mathbf{x}-\mathbf{y}) q(\mathbf{y}) d\sigma(\mathbf{y}),$$

  where the second term in each expression is an improper integral. In practice, they are evaluated using the Cauchy principal value.

- There exist several representations of the hypersingular operator W. The following is obtained by means of integration by parts: for every $u, v \in H^{1/2}(\Gamma)$

$$\langle \mathrm{W_L}\, u, v\rangle_{H^{-1/2}(\Gamma)\times H^{1/2}(\Gamma)} = \int_{\Gamma\times\Gamma} G_\mathrm{L}(\mathbf{x}-\mathbf{y})(\mathrm{curl}_{\Gamma,\mathbf{A},\mathbf{0}}(\overline{v})(\mathbf{x}) \cdot \mathrm{curl}_{\Gamma,\mathbf{A},\mathbf{0}}(u)(\mathbf{y}))\, \mathrm{d}\sigma(\mathbf{x})\, \mathrm{d}\sigma(\mathbf{y})$$
$$+ c \int_{\Gamma\times\Gamma} G_\mathrm{L}(\mathbf{x}-\mathbf{y}) \overline{v}(\mathbf{x}) u(\mathbf{y}) (\mathbf{A}^{1/2}\mathbf{n}(\mathbf{x}) \cdot \mathbf{A}^{1/2}\mathbf{n}(\mathbf{y}))\, \mathrm{d}\sigma(\mathbf{x})\, \mathrm{d}\sigma(\mathbf{y}),$$

  where $\mathrm{curl}_{\Gamma,\mathbf{A},\mathbf{f}}(u) := (\mathbf{A}^{1/2}\nabla u^* + u\mathbf{A}^{1/2}\mathbf{f}) \times \mathbf{A}^{1/2}\mathbf{n}$ for $u \in H^{1/2}(\Gamma)$ and u^* its extension in Ω.

Remark that for $\mathbf{A} = \mathbf{I}$ and $\mathbf{b} = 0$, $\mathrm{curl}_{\Gamma,\mathbf{I},0}$ is the usual surface curl operator of a tangent vector.

2.3.3 Calderón projector

In this section, we derive some useful relations between the BIO introduced previously. The idea comes from the following remark: using Corollary 2.20, we see that for $u \in H^1_{\mathrm{loc}}(\Omega)$ such that $\mathrm{L}(u) = 0$ (and u satisfies the appropriate condition at infinity), we have $\gamma \circ \mathrm{G_L}(\gamma(u)) = \gamma(u)$. Taking $u = \mathrm{G_L}((v,q))$ with $(v,q) \in H^{1/2}(\Gamma) \times H^{-1/2}(\Gamma)$, we see that $\gamma \circ \mathrm{G_L}$ is a projector on $H^{1/2}(\Gamma) \times H^{-1/2}(\Gamma)$. In fact, we have the following result

Theorem 2.24 (Calderón projector, [139, Proposition 3.6.2.])**.** *We define the Cauchy data set* $C_\mathrm{L}(\Omega)$ *as:*

$$C_\mathrm{L}(\Omega) := \{\gamma(u) \mid \mathrm{L}(u) = 0 \text{ in } \Omega \text{ and } u \text{ satisfies the appropriate conditions at infinity if } \Omega \text{ is unbounded}\}.$$

Then $\mathrm{P_L} := \gamma \circ \mathrm{G_L} : H^{1/2}(\Gamma) \times H^{-1/2}(\Gamma) \to H^{1/2}(\Gamma) \times H^{-1/2}(\Gamma)$ *is a continuous projector, so called Calderón projector whose range coincides with* $C_\mathrm{L}(\Omega)$*, i.e. for any* $(v,q) \in H^{1/2}(\Gamma) \times H^{-1/2}(\Gamma)$ *we have*

$$(v,q) \in C_\mathrm{L}(\Omega) \iff (\gamma \circ \mathrm{G_L})((v,q)) = (v,q).$$

By definition, we can also write the Calderón projector as follows: for every $(v,q) \in H^{1/2}(\Gamma) \times H^{-1/2}(\Gamma)$,

$$\mathrm{P_L}((v,q)) = \gamma \circ \mathrm{G_L}((v,q)) = \begin{pmatrix} \frac{\mathrm{I_d}}{2} + \mathrm{K_L} & \mathrm{V_L} \\ \mathrm{W_L} & \frac{\mathrm{I_d}}{2} + \mathrm{K'_L} \end{pmatrix} \begin{pmatrix} v \\ q \end{pmatrix} = \left(\frac{\mathrm{I_d} + \mathrm{A_L}}{2}\right)(v,q).$$

Using the jump relations from Theorem 2.22, $\mathrm{A_L}$ can also be rewritten as follows:

$$\mathrm{A_L} = 2\{\gamma\} \circ \mathrm{G_L}$$

such that clearly $\mathrm{A}_{\mathrm{L}}^2 = \mathrm{I_d}$. In particular, we have the following corollary:

Corollary 2.25 (Calderón identities, [139, Proposition 3.6.4.])**.** *We have the following relations, called* Calderón identities

$$\mathrm{K_L}\,\mathrm{V_L} = -\,\mathrm{V_L}\,\mathrm{K_L'} \;\; \textit{on } H^{-1/2}(\Gamma), \quad \mathrm{W_L}\,\mathrm{K_L} = -\,\mathrm{K_L'}\,\mathrm{W_L} \;\; \textit{on } H^{1/2}(\Gamma), \tag{2.9a}$$

and

$$\mathrm{V_L}\,\mathrm{W_L} = \frac{\mathrm{I_d}}{4} - \mathrm{K_L^2} \;\; \textit{on } H^{1/2}(\Gamma), \quad \mathrm{W_L}\,\mathrm{V_L} = \frac{\mathrm{I_d}}{4} - \mathrm{K_L'^2} \;\; \textit{on } H^{-1/2}(\Gamma). \tag{2.9b}$$

Proof. According to Theorem 2.24, we have $\mathrm{P_L^2} = \mathrm{P_L}$ on $C_{\mathrm{L}}(\Omega)$, which can be rewritten

$$\begin{pmatrix} \frac{\mathrm{I_d}}{4} + \mathrm{K_L} + \mathrm{K_L^2} + \mathrm{V_L}\,\mathrm{W_L} & \mathrm{V_L} + \mathrm{K_L}\,\mathrm{V_L} + \mathrm{V_L}\,\mathrm{K_L'} \\ \mathrm{W_L} + \mathrm{W_L}\,\mathrm{K_L} + \mathrm{K_L'}\,\mathrm{W_L} & \mathrm{W_L}\,\mathrm{V_L} + \frac{\mathrm{I_d}}{4} + \mathrm{K_L'} + \mathrm{K_L'^2} \end{pmatrix} = \begin{pmatrix} \frac{\mathrm{I_d}}{2} + \mathrm{K_L} & \mathrm{V_L} \\ \mathrm{W_L} & \frac{\mathrm{I_d}}{2} + \mathrm{K_L'} \end{pmatrix}.$$

□

Remark 2.26. *The operators* $\mathrm{V_L}$ *and* $\mathrm{W_L}$ *have the order* 1 *and* -1 *respectively. From Equation* (2.9b)*, we see that the product of one with the other defines a operator of order* 0*. This property can be used to define efficient preconditioners [146, 26, 94], usually called* Calderón preconditioners*.*

2.3.4 Dirichlet-to-Neumann operator

In this section, we describe an interesting relation between the BIO introduced previously and the *Dirichlet-to-Neumann* operator (also called *Steklov-Poincaré* operator) defined as follows: Let $g_D \in H^{1/2}(\Gamma)$, we consider a Dirichlet problem

$$\begin{cases} \mathrm{L}(u) = 0 & \text{in } \Omega, \\ \gamma_D(u) = g_D & \text{in } \Gamma. \end{cases}$$

Assume this problem is well-posed (see Section 2.2.4 to see conditions on the coefficients of the differential operator L to satisfy this assumption), the mapping $g_D \mapsto \gamma_N(u)$ defines the Dirichlet-to-Neumann operator denoted $\mathrm{DtN_L}$. This operator is clearly continuous from $H^{1/2}(\Gamma)$ to $H^{-1/2}(\Gamma)$. Besides, it can be expressed using BIO as proved in the next lemma.

Theorem 2.27. *The Dirichlet-to-Neumann operator can be written explicitly as*

$$\mathrm{DtN_L} = (\mathrm{V_L})^{-1}\left(\frac{\mathrm{I}_d}{2} - \mathrm{K_L}\right) = \left(\frac{\mathrm{I}_d}{2} - \mathrm{K_L'}\right)^{-1} \mathrm{W_L}\,.$$

Proof. According to Theorem 2.24, for every $(v, q) \in C_{\mathrm{L}}(\Omega)$, we have

$$\mathrm{P}_{\mathrm{L}}((v,q)) = (v,q) \Leftrightarrow \begin{cases} \left(\frac{\mathrm{I_d}}{2} + \mathrm{K_L}\right) v + \mathrm{V_L}\, q = v \\ \mathrm{W_L}\, v + \left(\frac{\mathrm{I}_d}{2} + \mathrm{K}'_{\mathrm{L}}\right) q = q \end{cases}$$
$$\Leftrightarrow \begin{cases} (\mathrm{V_L})^{-1} \left(\frac{\mathrm{I}_d}{2} - \mathrm{K_L}\right) v = q \\ \left(\frac{\mathrm{I}_d}{2} - \mathrm{K}'_{\mathrm{L}}\right)^{-1} \mathrm{W_L}\, v = q. \end{cases}$$

Since $(v, q) \in C_{\mathrm{L}}(\Omega)$, we have in particular $\mathrm{DtN_L}(v) = q$ so that we deduce the expected relations. □

2.4 Integral equations for elliptic boundary value problems

We have now everything to define *Boundary Integral Equations* (BIE). There are several ways to formulate BIE related to the BVP (2.6). One way is to use the single or double layer potential as *ansatz*, and another possibility is to use the Calderón projector. The former give rise to the *indirect method*, while the latter is used to formulate the *direct method*.

2.4.1 The indirect method

As we have seen previously, for every $v \in H^{1/2}(\Gamma)$ and $q \in H^{-1/2}(\Gamma)$, $\mathrm{SL_L}(q)$ and $\mathrm{DL_L}(v)$ satisfy $\mathrm{L}(u) = 0$ with the appropriate conditions at infinity if Ω is unbounded. So that it remains to find v or q such that $\mathrm{SL_L}(q)$ and $\mathrm{DL_L}(v)$ also satisfy the boundary conditions given on Γ. In the case of a Dirichlet problem with $g_D \in H^{1/2}(\Gamma)$, we then can look for $q \in H^{-1/2}(\Gamma)$ such that $\gamma_D \circ \mathrm{SL_L}(q) = \mathrm{V_L}(q) = g_D$, or for $v \in H^{1/2}(\Gamma)$ such that $\gamma_D \circ \mathrm{DL_L}(v) = (\frac{1}{2}\,\mathrm{I_d} + \mathrm{K_L})(v) = g_D$. Then, we can multiply by a test function, which gives the following variational formulations:

- For Dirichlet problems, taking the Dirichlet trace of the single layer potential: Find $q \in H^{-1/2}(\Gamma)$ such that

$$\langle \mathrm{V_L}(q), r\rangle_{H^{1/2}(\Gamma)\times H^{-1/2}(\Gamma)} = \langle g_D, r\rangle_{H^{1/2}(\Gamma)\times H^{-1/2}(\Gamma)}, \quad \forall r \in H^{-1/2}(\Gamma). \tag{2.10}$$

  Or, taking the Dirichlet trace of the double layer potential: Find $v \in H^{1/2}(\Gamma)$ such that

$$\left\langle \left(\frac{1}{2}\,\mathrm{I_d} + \mathrm{K_L}\right)(v), r \right\rangle_{H^{1/2}(\Gamma)\times H^{-1/2}(\Gamma)} = \langle g_D, r\rangle_{H^{1/2}(\Gamma)\times H^{-1/2}(\Gamma)}, \quad \forall r \in H^{-1/2}(\Gamma). \tag{2.11}$$

- For Neumann problems, taking the Neumann trace of the double layer potential: Find $v \in H^{1/2}(\Gamma)$ such that

$$\langle \mathrm{W_L}(v), w\rangle_{H^{-1/2}(\Gamma)\times H^{1/2}(\Gamma)} = \langle g_N, w\rangle_{H^{-1/2}(\Gamma)\times H^{1/2}(\Gamma)}, \quad \forall w \in H^{1/2}(\Gamma). \tag{2.12}$$

Or, taking the Neumann trace of the single layer potential: Find $q \in H^{-1/2}(\Gamma)$ such that

$$\left\langle \left(\frac{1}{2}\,\mathrm{I_d} + \mathrm{K_L'} \right)(v), w \right\rangle_{H^{-1/2}(\Gamma)\times H^{1/2}(\Gamma)} = \langle g_N, w\rangle_{H^{-1/2}(\Gamma)\times H^{1/2}(\Gamma)}, \quad \forall w \in H^{1/2}(\Gamma). \tag{2.13}$$

2.4.2 The direct method

This method is based on the Calderón projector and Theorem 2.24. For Dirichlet problems, we have that $q \in H^{-1/2}(\Gamma)$ is solution if and only if (g_D, q) is a fixed point of the Calderón projector

$$\mathrm{P_L}((g_D, q)) = \begin{bmatrix} \frac{\mathrm{I}_d}{2} + \mathrm{K_L} & \mathrm{V_L} \\ \mathrm{W_L} & \frac{\mathrm{I_d}}{2} + \mathrm{K_L'} \end{bmatrix} \begin{bmatrix} g_D \\ q \end{bmatrix} = \begin{bmatrix} g_D \\ q \end{bmatrix}.$$

Taking the first row of the previous relation, we obtain the following variational formulation: find $q \in H^{-1/2}(\Gamma)$ such that

$$\langle \mathrm{V_L}(q), r\rangle_{H^{1/2}(\Gamma)\times H^{-1/2}(\Gamma)} = \frac{1}{2}\langle g_D, r\rangle_{H^{1/2}(\Gamma)\times H^{-1/2}(\Gamma)} - \langle \mathrm{K_L}(g_D), r\rangle_{H^{1/2}(\Gamma)\times H^{-1/2}(\Gamma)}, \quad \forall r \in H^{-1/2}(\Gamma). \tag{2.14}$$

Taking the second row, gives the following variational formulation: find $q \in H^{-1/2}(\Gamma)$ such that

$$\frac{1}{2}\langle q, w\rangle_{H^{-1/2}(\Gamma)\times H^{1/2}(\Gamma)} - \langle \mathrm{K_L'}(q), w\rangle_{H^{-1/2}(\Gamma)\times H^{1/2}(\Gamma)} = \langle \mathrm{W_L}(g_D), w\rangle_{H^{-1/2}(\Gamma)\times H^{1/2}(\Gamma)}, \quad \forall w \in H^{1/2}(\Gamma). \tag{2.15}$$

Similarly, for Neumann problem, we can obtain the following formulations. Find $v \in H^{1/2}(\Gamma)$ such that

$$\langle \mathrm{W_L}(v), w\rangle_{H^{-1/2}(\Gamma)\times H^{1/2}(\Gamma)} = \frac{1}{2}\langle g_N, w\rangle_{H^{-1/2}(\Gamma)\times H^{1/2}(\Gamma)} - \langle \mathrm{K_L'}(g_N), w\rangle_{H^{-1/2}(\Gamma)\times H^{1/2}(\Gamma)}, \quad \forall w \in H^{1/2}(\Gamma). \tag{2.16}$$

Find $v \in H^{1/2}(\Gamma)$ such that

$$\frac{1}{2}\langle v, r\rangle_{H^{1/2}(\Gamma)\times H^{-1/2}(\Gamma)} - \langle \mathrm{K_L}(v), r\rangle_{H^{1/2}(\Gamma)\times H^{-1/2}(\Gamma)} = \langle \mathrm{V_L}(g_N), r\rangle_{H^{1/2}(\Gamma)\times H^{-1/2}(\Gamma)}, \quad \forall r \in H^{-1/2}(\Gamma). \tag{2.17}$$

Remarks 2.28 (Comparaison between Direct and Indirect methods)**.** *As in [139, Section 3.4.3], we can formulate the following remarks:*

- *The right-hand side for the direct formulation is defined by an integral operator, while the right-hand side of the indirect formulation is just a $L^2(\Gamma)$ inner product between the boundary data and the test functions.*
- *Solving the direct formulation gives directly the trace of the solution. The indirect formulation only gives a boundary function that needs to be evaluated by means of potentials.*

- *To compute the solution in Ω, one needs to use only one potential with the indirect formulation by definition of the ansatz. But two potentials are needed for the direct formulation (see Corollary 2.20).*

According to the previous remarks, a rule of thumb for choosing between these two methods is to use the direct formulation when the unknown boundary data is needed, or the indirect formulation if the unknown solution needs to be computed in many points of Ω.

2.4.3 Well-posedness

As in Section 2.2.4, we can use Lax-Milgram theorem (Theorem 2.13) or Fredholm's theory (Theorem 2.14), depending on the properties of the bilinear forms introduced in the two last sections.

We can differentiate two types of formulation among those introduced previously: those that use $\mathrm{V_L}$ and $\mathrm{W_L}$, called *formulations of first kind*, and those that use $\mathrm{K_L}$ and $\mathrm{K_L'}$, called *formulations of second kind*.

In the case of formulations of first kind, the same approach as in Section 2.2.4 can be used. Cases where the previous bilinear forms are elliptic usually correspond to cases where the bilinear form a from (2.7) is also elliptic. Let us denote $a_{\mathrm{V_L}}(q,r) := \langle \mathrm{V_L}(q), r\rangle_{H^{1/2}(\Gamma)\times H^{-1/2}(\Gamma)}$ and $a_{\mathrm{W_L}}(q,r) := \langle \mathrm{W_L}(v), w\rangle_{H^{-1/2}(\Gamma)\times H^{1/2}(\Gamma)}$ for $v, w \in H^{1/2}(\Gamma)$ and $q, r \in H^{-1/2}(\Gamma)$, we have the following lemma (see [139, Theorem 3.5.3. and 3.5.4] and [125]):

Lemma 2.29 (Ellipticity of $a_{\mathrm{V_L}}$ and $a_{\mathrm{W_L}}$)**.** *The bilinear form $a_{\mathrm{V_L}}$ and $a_{\mathrm{W_L}}$ are respectively*

- *$H^{-1/2}(\Gamma)$ and $H^{1/2}(\Gamma)$ – elliptic if $a_{\min} c > |\mathbf{b}|^2$,*
- *$H^{-1/2}(\Gamma)$ and $H^{1/2}(\Gamma)/\mathbb{C}$ – elliptic if $\mathrm{L} = -\Delta$ and $d = 3$,*
- *$H^{-1/2}(\Gamma)$ and $H^{1/2}(\Gamma)/\mathbb{C}$ – elliptic if $\mathrm{L} = -\Delta$, $d = 2$, but the fundamental solution used in the weakly singular operator $\mathrm{V}_{-\Delta}$ has to be modified as follows:*

$$G_{-\Delta}(\mathbf{x}) = \frac{1}{2\pi} \ln\left(\frac{\alpha}{\|\mathbf{x}\|}\right),$$

with $\alpha > \alpha_\Gamma > 0$ where α_Γ is the logarithmic capacity of Γ.

Using Lax-Milgram Theorem 2.13, we deduce that the formulations of first kind are well-posed when L satisfies the conditions given in the previous lemma. More generally, we have the following theorem

Theorem 2.30 ([41, Theorem 2])**.** *The bilinear form $a_{\mathrm{V_L}}$ and $a_{\mathrm{W_L}}$ satisfy a Gårding inequality. More specifically, there exists a compact operator $T_{\mathrm{V_L}} : H^{-1/2}(\Gamma) \to H^{1/2}(\Gamma)$ and $T_{\mathrm{W_L}} : H^{1/2}(\Gamma) \to H^{-1/2}(\Gamma)$ such that*

$$\begin{aligned}
\langle (\mathrm{V_L} + T_{\mathrm{V_L}})(q), q\rangle_{H^{1/2}(\Gamma)\times H^{-1/2}(\Gamma)} &\geq c_{\mathrm{V_L}} \|q\|^2_{H^{-1/2}(\Gamma)}, \quad \forall q \in H^{-1/2}(\Gamma),\\
\langle (\mathrm{W_L} + T_{\mathrm{W_L}})(v), v\rangle_{H^{-1/2}(\Gamma)\times H^{1/2}(\Gamma)} &\geq c_{\mathrm{W_L}} \|w\|^2_{H^{1/2}(\Gamma)}, \quad \forall w \in H^{1/2}(\Gamma).
\end{aligned}$$

A more general statement of Fredholm's alternative [102, Theorem 5.3.10] shows that if a bilinear form satisfies a Gårding inequality and if it is injective, then it is well-posed. In the case of the Helmholtz equation, we have the following result.

Theorem 2.31 ([139, Theorem 3.9.1. and Exercise 3.9.5.])**.** *We consider the Helmholtz equation:* $\mathbf{A} = \mathbf{I}_{\mathrm{d}}$, $\mathbf{b} = \mathbf{0}$ *and* $c = -k^2$ *with* $k > 0$*. Then, the weakly singular operator* $\mathrm{V_L}$ *is injective if and only if*

- Ω *is bounded and* k^2 *is not an eigenvalue for the Laplace operator in* Ω *with Dirichlet conditions,*
- *or* $\mathbb{R}^d \setminus \Omega$ *is bounded and* k^2 *is not an eigenvalue for the Laplace operator in* $\mathbb{R}^d \setminus \Omega$ *with Dirichlet conditions.*

Similarly, the hypersingular operator $\mathrm{W_L}$ *for the Helmholtz equation is injective if and only if*

- Ω *is bounded and* k^2 *is not an eigenvalue for the Laplace operator in* Ω *with Neumann conditions,*
- *or* $\mathbb{R}^d \setminus \Omega$ *is bounded and* k^2 *is not an eigenvalue for the Laplace operator in* $\mathbb{R}^d \setminus \Omega$ *with Neumann conditions.*

Using the previous theorem and Theorem 2.30, we deduce that the formulations of first kind for the Helmholtz equation are well-posed. Surprisingly, exterior problems for the Helmholtz equation are well-posed for every $k > 0$ (see [139, Section 3.9.3.]), but their standard BIE cannot be solved for the resonant frequencies of the associated interior problem. This issue led to the introduction of modified BIE, see [139, Section 3.9.4.].

2.4.4 Screen problems

So far, we considered BIE on closed surfaces, but we are also interested in screen problems, i.e., for $\Gamma_0 \subset \Gamma = \partial\Omega$ such that $\partial\Gamma_0 \neq \emptyset$, $g_D \in H^{1/2}(\Gamma_0)$ and $g_N \in H^{-1/2}(\Gamma_0)$, we look for $u \in H^1_{\mathrm{loc}}(\mathbb{R}^d \setminus \Gamma_0)$ such that

$$\begin{cases} \mathrm{L}(u) = 0, & \text{in } \mathbb{R}^d \setminus \Gamma_0 \\ \gamma_D(u) = g_D, & \text{on } \Gamma_0 \end{cases} \quad \text{or} \quad \begin{cases} \mathrm{L}(u) = 0, & \text{in } \mathbb{R}^d \setminus \Gamma_0 \\ \gamma_N(u) = g_N, & \text{on } \Gamma_0 \end{cases}$$

with the appropriate conditions at infinity. We define the localization of the BIO on Γ_0 as follows: for every $v \in H^{1/2}(\Gamma)$, $q \in H^{-1/2}(\Gamma)$ such that $\mathrm{supp}(v) \subset \overline{\Gamma}_0$ and $\mathrm{supp}(q) \subset \overline{\Gamma}_0$, i.e., $v|_{\Gamma_0} \in \widetilde{H}^{1/2}(\Gamma_0)$ and $q|_{\Gamma_0} \in \widetilde{H}^{-1/2}(\Gamma_0)$

$$\begin{aligned} \mathrm{V}_{\mathrm{L},\Gamma_0}(q|_{\Gamma_0}) &:= \mathrm{V_L}(q)|_{\Gamma_0}, \quad & \mathrm{K}_{\mathrm{L},\Gamma_0}(v|_{\Gamma_0}) &:= \mathrm{K_L}(v)|_{\Gamma_0}, \\ \mathrm{W}_{\mathrm{L},\Gamma_0}(v|_{\Gamma_0}) &:= \mathrm{W_L}(v)|_{\Gamma_0}, \quad & \mathrm{K}'_{\mathrm{L},\Gamma_0}(q|_{\Gamma_0}) &:= \mathrm{K}'_{\mathrm{L}}(q)|_{\Gamma_0}. \end{aligned}$$

Following the same approach as in Section 2.4.1, these problems can be reformulated as BIE as follows

- Find $q \in \widetilde{H}^{-1/2}(\Gamma_0)$ such that

$$\langle \mathrm{V}_{\mathrm{L},\Gamma_0}(q), r\rangle_{H^{1/2}(\Gamma_0)\times\widetilde{H}^{-1/2}(\Gamma_0)} = \langle g_D, r\rangle_{H^{1/2}(\Gamma_0)\times\widetilde{H}^{-1/2}(\Gamma_0)}, \quad \forall r \in \widetilde{H}^{-1/2}(\Gamma_0). \tag{2.18}$$

- Find $v \in \widetilde{H}^{1/2}(\Gamma_0)$ such that

$$\langle \mathrm{W}_{\mathrm{L},\Gamma_0}(v), w\rangle_{H^{-1/2}(\Gamma_0)\times\widetilde{H}^{1/2}(\Gamma_0)} = \langle g_N, w\rangle_{H^{-1/2}(\Gamma_0)\times\widetilde{H}^{1/2}(\Gamma_0)}, \quad \forall w \in \widetilde{H}^{1/2}(\Gamma_0). \tag{2.19}$$

Notice that since Γ_0 has a boundary, the energy spaces become $\widetilde{H}^{1/2}(\Gamma_0)$ and $\widetilde{H}^{-1/2}(\Gamma_0)$, which are respectively subspaces of $H^{1/2}(\Gamma_0)$ and $H^{-1/2}(\Gamma_0)$ containing functions whose extension by zero on Γ are respectively in $H^{1/2}(\Gamma)$ and $H^{-1/2}(\Gamma)$. Thus, we can recover properties about well-posedness using the BIE on Γ just by extending by zero the considered functions (see Theorem [139, Theorem 3.5.9.]). We also refer to [148] for more details.

2.4.5 Galerkin discretization

The discretization of BIE is a large field of research and we only mention the main ideas and its specificities. There exist several ways to discretize them but we will focus here on the *Boundary Element Method* (BEM), which is similar to the *Finite Element Method* (FEM) in the sense that the solutions of the BIE introduced in Sections 2.4.1 and 2.4.2 will be approximated by a Galerkin method using the weak formulation where the subspaces of finite dimension are finite element spaces defined on the boundary.

For example, the approximation of Equation (2.10) can be formulated as follows: find $q_h \in \mathcal{V}_h \subset H^{-1/2}(\Gamma)$ such that

$$\langle \mathrm{V_L}(q_h), r_h \rangle_{H^{1/2}(\Gamma)\times H^{-1/2}(\Gamma)} = \langle g_D, r_h \rangle_{H^{1/2}(\Gamma)\times H^{-1/2}(\Gamma)}, \quad \forall r \in \mathcal{V}_h, \tag{2.20}$$

where $\mathcal{V}_h$ can be the space of piecewise constant functions on Γ with h the characteristic size of its associated mesh for example. The numerical analysis to prove the well-posedness and the convergence of the approximation is not too different from FEM, so we refer to [139, Chapter 4] for more details.

In practice, we need to compute the coefficients $a_{\mathrm{V_L}}(\varphi_j, \varphi_i)$ where φ_j and φ_i are two elements of the finite element basis $\{\varphi_l\}$ associated with $\mathcal{V}_h$. That is why we need the explicit expressions introduced in Section 2.3.2, for example:

$$a_{\mathrm{V_L}}(\varphi_j, \varphi_i) = \int_{\Gamma\times\Gamma} G_{\mathrm{L}}(\mathbf{x}-\mathbf{y})\varphi_i(\mathbf{x})\varphi_j(\mathbf{y})\,\mathrm{d}\sigma(\mathbf{x})\,\mathrm{d}\sigma(\mathbf{y}).$$

There are two interesting differences compared to usual FEM. First, this is a double integral with a non-local kernel so that all the coefficients are non-null a priori, so that the matrix is dense. Then, when the support of φ_j and φ_i are close to each other, or intersect with each other, the integral is quasi-singular so that extra care must be taken with quadrature rules. Several techniques exist to approximate this type of double integral [139, Chapter 5].

The discrete Equation (2.20) leads to the following linear system

$$\mathbf{A}_{\mathrm{V_L}}\mathbf{x} = \mathbf{b},$$

where $(\mathbf{A}_{\mathrm{V_L}})_{i,j} = a_{V_{\mathrm{L}}}(\varphi_j, \varphi_i)$, $\mathbf{b}_j = \langle g_D, \varphi_j \rangle_{H^{1/2}(\Gamma)\times H^{-1/2}(\Gamma)}$ and $q_h = \sum_i \mathbf{x}_i\varphi_i$. Because of the non-locality of the bilinear form $a_{\mathrm{V_L}}$, BEM matrices like $\mathbf{A}_{\mathrm{V_L}}$ are dense, which precludes direct methods for the inversion as Gauss–Jordan elimination, whose complexity grows cubically with respect to N, the number of degrees of freedom. Besides, the storage and the numerical work to assemble the matrices are both quadratic. Similarly, iterative solvers should be quite expensive because of the matrix-vector product that is also quadratic. Solutions to these three issues can be found with methods like *panel clustering* [78, 79], *Fast Multipole Method* (FMM) [134, 70, 42, 43], *hierarchical matrices* ($\mathcal{H}$ and $\mathcal{H}^2$-matrices) [74, 76, 77, 17, 12, 75] *Adaptive Cross Approximation* (ACA) [11, 13, 18, 12], *Sparse Cardinal Sine Decomposition* (SCSD) [4] and *wavelet compression techniques* [5]. They all give an approximate representation of the matrix that allows storage and matrix-vector product with almost linear complexity. In our numerical tests, we usually use

hierarchical matrices with the C++ library Htool[3] that we present in Chapter 6.

These techniques have the advantage of lowering the storage cost and make it easier to apply the operator to a vector, but the underlying structure is not well suited for direct solvers such as exact LU decomposition. Nevertheless, direct solvers to be used in conjunction with a compression procedure have been developed, for example, hierarchical LU decomposition [75] and a fast direct solver as described in [121, 71]. Another approach relies on iterative solvers, such as Conjugate Gradient (CG) [83] for symmetric positive definite matrices or Generalized Minimal Residual (GMRes) [137]. These methods have the advantage of relying only on matrix-vector products, they are non-intrusive in the sense that they are independent of the compression method used. However, the number of iterations needed strongly depends on the spectral properties of the matrix of the system. In the case of the single layer for the Laplace problem, the condition number $\kappa(\mathbf{A}_{\mathrm{V_L}}) := \|\mathbf{A}_{\mathrm{V_L}}\|_2 \|\mathbf{A}_{\mathrm{V_L}}^{-1}\|_2$ typically deteriorates like $O(h^{-1})$ [139, Section 4.5], and we have the same behavior for the hypersingular operator. This means that the number of iterations when using CG will increase when refining the mesh size as the square root of h^{-1}.

To circumvent this problem, one has to find a relevant preconditioner, i.e., a matrix $\mathbf{P}$ such that $\mathbf{P}\mathbf{A}_h$ has better spectral properties (typically, a condition number bounded independently of the meshsize) and that is not too costly in a matrix-vector multiplication, because instead of solving the previous linear system, we will solve

$$\mathbf{P}\mathbf{A}_h\mathbf{u}_h = \mathbf{P}\mathbf{b}_h.$$

This approach is called *left preconditioning* but another popular approach is *right preconditioning* which consists in solving $\mathbf{A}_h\mathbf{P}\mathbf{x}_h = \mathbf{b}_h$ with $\mathbf{x}_h = \mathbf{P}^{-1}\mathbf{u}_h$. These two approaches are closely related since their Krylov space only differs by a multiplication by $\mathbf{P}$, but the minimized residuals also differ because the latter is the residual of non-preconditioned system $\|\mathbf{b}_h - \mathbf{A}_h\mathbf{u}_h\|$ while the former minimizes $\|\mathbf{M}^{-1}(\mathbf{b}_h - \mathbf{A}_h\mathbf{u}_h\|)$ (see [138, Section 9.3.4] for more details).

Various preconditioning strategies have been proposed for BEM matrices. One of the most popular one is *Calderón preconditioning* [146], which is the application of *operator preconditioning* [94] based on Calderón identities from Equation (2.9a) that shows that certain products of boundary integral operators yield a compact perturbation of the identity. An alternative consists in building a sparse approximation of the inverse to precondition our problem, which is usually called *Sparse Approximate Inverse Preconditioner* (SPAI) [3]. Finally, another approach, proposed in [86, 156] for example, relies on adapting well-known preconditioners stemming from Domain Decomposition Methods (DDM) to BEM matrices.

[3] https://github.com/PierreMarchand20/htool

CHAPTER 3

Domain Decomposition Methods

Outline of the current chapter

We actually focus on a particular type of *Domain Decomposition Methods* (DDM), namely *Schwarz methods*. Schwarz methods is one of the first class of domain decomposition methods, and they have this particularity of having two related formulations. Historically, they have been introduced as iterative algorithms to approximate a solution solving local problems and exchanging informations between neighbouring subdomains. But after discretization, these iterative algorithms can be rewritten as stationary methods and define a specific preconditioner. Thus, one can be interested in studying the efficiency of such preconditioners applied in *Krylov methods*, that are more efficient than stationary methods. In this case, the speed of convergence is often related to the condition number of the linear system, that is to say, the ratio between the largest and lowest eigenvalues. In particular, we want the condition number to be bounded independently of the meshsize and the number of subdomains so that refining or increasing the number of subdomains do not change the number of iterations of the linear solver. One usually uses the *fictitious space lemma* to study the condition number of the preconditioned linear system. We refer to [57] for a historical review on Schwarz methods.

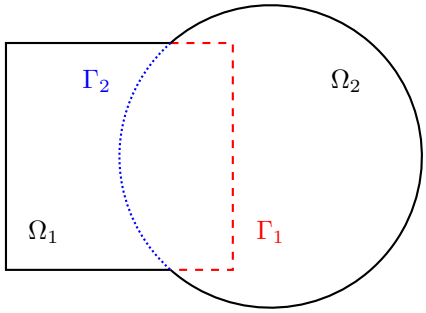


Figure 3.1 – Example of domain decomposition: the domain Ω consists of the union of a rectangle Ω_1 and a disk Ω_2 with an overlap.

Another interesting fact about Schwarz methods is that, since they mainly consist in solving local problems and exchanging informations between adjacent subdomains, there is a lack of global communication that prevent the method to be scalable in a *High-Performance Computing* (HPC) context. Meaning that increasing the number of subdomains will not be as efficient as expected. For example, suppose that we associate a subdomain to a processor, an engineer solving a problem will expect to solve his problem twice faster if he doubles the number of processors. But this will be unlikely because the lack of global communication will affect the efficiency of the method. See [45, Chapter 4] for numerical evidences.

To fix this issue, a *coarse space* is usually added so that a minimum of global communication is present in the method. The first coarse spaces were usually based on solving a coarse problem on a coarse grid, but recent progress in DDM was achieved and a coarse space construction based on Generalized Eigenproblems in the Overlap (which we will call *GenEO coarse space*) has been proposed. The advantage of the latter is a more "blackbox" approach to build robust preconditioners. This coarse space has been first introduced in [142], see also [45, Chapter 7] for another version. We refer to [161, 118] for a more general presentation on coarse spaces for DDM.

In this chapter, we recall the main ideas of *Domain Decomposition Methods* (DDM), more precisely of Schwarz methods at the continuous level in Section 3.1 and *GenEO* coarse space in Section 3.3, after introducing all the discrete setting in Section 3.2. We refer to [153, 45] for more details. Note that we are only considering finite element discretization, as it is commonly done in the DDM literature.

3.1 Schwarz methods at the continuous level

3.1.1 Origins

The origin of domain decomposition methods stems from the work of Schwarz in 1870 (see [57] for a historical review of Schwarz methods). The goal was to establish existence of a solution for Poisson problem with non-smooth boundaries such as:

$$\begin{cases} -\Delta u = f \text{ in } \Omega, \\ \quad u = 0 \text{ on } \partial\Omega, \end{cases} \tag{3.1}$$

where f is regular enough and $\Omega = \Omega_1 \cup \Omega_2$ (see Section 3.1.1). Without nowadays modern theoretical tools, such as Sobolev spaces and Lax-Milgram theorem, the proof for the well-posedness of Equation (3.1) was limited to simple geometries (circle, disc…) using Fourier transform. With this in mind, Schwarz suggested a constructive argument, which relies on a decomposition of the domain in simple geometries, to prove the well-posedness for more general geometries. The idea is to solve the problem alternatively on each of these subdomains, using transmission conditions coming from the solution computed in the neighboring subdomains. More precisely, he introduced the following iterative algorithm: given an initial guess u^0, which vanishes on $\partial\Omega$, u^{m+1} is computed with the two following sequential steps:

$$\begin{cases} -\Delta u_1^{m+1} = f & \text{in } \Omega_1, \\ u_1^{m+1} = 0 & \text{on } \partial\Omega_1 \cap \partial\Omega, \\ u_1^{m+1} = u_2^m & \text{on } \Gamma_1, \end{cases} \quad \begin{cases} -\Delta u_2^{m+1} = f & \text{in } \Omega_2, \\ u_2^{m+1} = 0 & \text{on } \partial\Omega_2 \cap \partial\Omega, \\ u_2^{m+1} = u_1^{m+1} & \text{on } \Gamma_2, \end{cases} \tag{3.2}$$

where $\Gamma_i = \partial\Omega_i \cap \overline{\Omega}_j$ for $i \neq j$ and $i, j \in \{1, 2\}$. Schwarz proved the convergence of this algorithm, called *Alternating Schwarz method*, and thus, the well-posedness of the Poisson problem in more general geometries. This algorithm was later studied in a more general framework in [112, 113].

One of the major drawbacks of this method is its sequential nature. A small modification of (3.2), which was introduced in [112], gives the *Parallel Schwarz method*:

$$\begin{cases} -\Delta u_1^{m+1} = f & \text{in } \Omega_1, \\ u_1^{m+1} = 0 & \text{on } \partial\Omega_1 \cap \partial\Omega, \\ u_1^{m+1} = u_2^m & \text{on } \partial\Omega_1 \cap \overline{\Omega}_2, \end{cases} \quad \begin{cases} -\Delta u_2^{m+1} = f & \text{in } \Omega_2 \\ u_2^{m+1} = 0 & \text{on } \partial\Omega_2 \cap \partial\Omega, \\ u_2^{m+1} = u_1^m & \text{on } \partial\Omega_2 \cap \overline{\Omega_1}. \end{cases} \tag{3.3}$$

3.1.2 Optimal Schwarz Method (OSM)

The main idea of Schwarz methods is to solve subproblems associated with a decomposition of geometric domain and to exchange some information between the neighboring subdomains. We saw in the previous section an example where there is an overlap between the subdomains and the information exchanged is a Dirichlet trace. But there exist a lot of variants, one of the most well-known is the P.L. Lions algorithm introduced in [113], where the exchanged information consists in Robin (or Fourier) data, which allows a convergence even without overlap.

More generally, one can wonder what is the best information to exchange, meaning the information that allows the fastest convergence. Let us denote a general elliptic problem as follows

$$\begin{cases} \mathrm{L}(u) = f \text{ in } \Omega, \\ u = g \text{ on } \partial\Omega. \end{cases}$$

We decompose Ω into two domains Ω_1 and Ω_2 as described in Section 3.1.1. The domain decomposition can be with and without overlap for OSM. We suppose the problem regular enough so that $u_i := u|_{\Omega_i}$, $i = 1, 2$ is continuous and has continuous normal derivatives across the interface Γ_i. Notice that Ω_1 and Ω_2 can overlap or not. A generic Schwarz method would

be as follows:

$$\begin{cases} \mathrm{L}(u_1^{m+1}) = f \text{ in } \Omega_1, \\ u_1^{m+1} = g \text{ on } \partial\Omega_1 \cap \partial\Omega, \\ \left.\begin{array}{r} \mu_1 \nabla u_1^{m+1} \cdot \mathbf{n}_1 + \mathcal{B}_1(u_1^{m+1}) \\ = -\mu_1 \nabla u_2^m \cdot \mathbf{n}_2 + \mathcal{B}_1(u_2^m) \end{array}\right] \text{ on } \Gamma_1, \end{cases} \qquad \begin{cases} \mathrm{L}(u_2^{m+1}) = f \text{ in } \Omega_2, \\ u_2^{m+1} = g \text{ on } \partial\Omega_2 \cap \partial\Omega, \\ \left.\begin{array}{r} \mu_2 \nabla u_2^{m+1} \cdot \mathbf{n}_2 + \mathcal{B}_2(u_2^{m+1}) \\ = -\mu_2 \nabla u_1^m \cdot \mathbf{n}_1 + \mathcal{B}_2(u_1^m) \end{array}\right] \text{ on } \Gamma_2, \end{cases}$$

with μ_1 and μ_2 real valued functions, $\mathcal{B}_1$ and $\mathcal{B}_2$ operators acting along Γ_1 and Γ_2 and u_1^0 and u_2^0 arbitrary initial guesses. Notice that we recover the original parallel Schwarz algorithm with $\mu_1 = \mu_2 = 0$ and $\mathcal{B}_1 = \mathcal{B}_2 = \mathrm{I_d}$. As for the P.L. Lions algorithm using Robin interface conditions, we need to set $\mu_1 = \mu_2 = 1$ and $\mathcal{B}_1 = \mathcal{B}_2 = \alpha\,\mathrm{I_d}$ with $\alpha \in \mathbb{R}$. The goal is to look for the operators $\mathcal{B}_i$ such that we have convergence in a minimal number of iterations. By linearity, the errors $e_i^m = u - u_i^m$ satisfy the same equations as u_i but with $f = 0$ and $g = 0$. To simplify, let us take $\mu_1 = \mu_2 = 1$.

Obviously, we cannot have convergence in one iteration since the initial guess is arbitrary. To have a convergence in two iterations, we need $e_1^2 = 0$ which requires

$$-\mu_2 \nabla e_2^1 \cdot \mathbf{n}_2 + \mathcal{B}_1(e_2^1) = 0 \qquad \text{on } \Gamma_1. \tag{3.4}$$

It should be noticed that e_2^1 is solution in Ω_2. Then, the idea is to introduce the Dirichlet-to-Neumann map:

$$\begin{aligned} \mathrm{DtN}^2 : H^{1/2}(\Gamma_1) &\to H^{-1/2}(\Gamma_1) \\ v &\mapsto \nabla w \cdot \mathbf{n}_{2|\partial\Omega_1 \cap \overline{\Omega}_2} \end{aligned}$$

where w satisfies the following problem:

$$\begin{cases} \mathrm{L}(w) = 0 \text{ in } \Omega_2 \setminus \overline{\Omega}_1, \\ w = 0 \text{ on } \partial\Omega_2 \cap \partial\Omega, \\ w = v \text{ on } \Gamma_1. \end{cases}$$

We observe that if $\mathcal{B}_1 = \mathrm{DtN}^2 \circ \gamma_D$, we recover Equation (3.4). Thus, Dirichlet-to-Neumann operators give the optimal interface condition and we obtain convergence in two iterations. It should be noticed that in practice, Dirichlet-to-Neumann operators are not partial differential operators which makes them difficult to implement. But this remark is the source of various optimized Schwarz methods that consist in approximating the Dirichlet-to-Neumann operator by partial differential operators (see [45, Section 2.5] for more details).

The result above is generalized for a domain decomposition with n vertical strips or n concentric rings in [128, Section 2] as follows: Let $\{\Omega_j\}_{j=0}^m$ denote the vertical strips or concentric rings without overlap, $\Gamma_{j,k} = \overline{\Omega}_j \cap \overline{\Omega}_k$

$$\begin{cases} \mathrm{L}(u_j^{m+1}) = f & \text{in } \Omega_j, \\ u_j^{m+1} = g & \text{on } \partial\Omega \cap \partial\Omega_j, \\ \left.\begin{array}{r} \nabla u_j^{m+1} \cdot \mathbf{n}_j + \widetilde{\mathrm{DtN}}_{kk}^j(u_j^{m+1}) \\ = -\nabla u_k^m \cdot \mathbf{n}_k + \widetilde{\mathrm{DtN}}_{kk}^j(u_k^m) \end{array}\right] & \text{on } \Gamma_{j,k},\ \forall k = 0, \ldots, n,\ k \neq j, \end{cases} \tag{3.5}$$

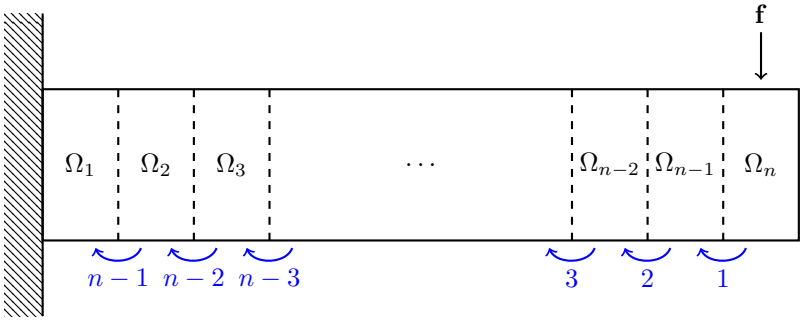


Figure 3.2 – Slow spread of the information about the presence of an external force for a domain decomposition method without a second level. Number of the iteration in blue.

where $\widetilde{\mathrm{DtN}}^j$ is the Dirichlet-to-Neumann operator associated with $\Omega \setminus \overline{\Omega_j}$, $\widetilde{\mathrm{DtN}}^j_{kl} = \mathrm{R}^j_{k,N} \circ \widetilde{\mathrm{DtN}}^j \circ (\mathrm{R}^j_{l,D})^*$ where $\mathrm{R}^j_{k,N} : H^{-1/2}(\Gamma_j) \to H^{-1/2}(\Gamma_{j,k})$ and $(\mathrm{R}^j_{l,D})^* : H^{1/2}(\Gamma_{j,l}) \to H^{1/2}(\Gamma_j)$ are the restriction and extension operator applied to, respectively, a Neumann trace and a Dirichlet trace.

In that case, the convergence is achieved in n iterations. To understand why we need at least n iterations, one can use the following argument: for n vertical strips as in Figure 3.2, let us assume that the support of the right-hand side $\mathbf{f}$ is contained in the domain on the far right, then the domain on the far left needs at least $n-1$ iterations just to be aware of the presence of a right-hand side (see Section 3.1.2). To conclude, we have convergence in a finite number of steps and the result is sharp.

3.2 Discrete setting

Now that we introduced the continuous point of view of the usual domain decomposition methods, we will present the discrete point of view. But first, we introduce in this section all the notions needed for discrete approximation of the solution to an elliptic problem such as Equation (3.1) with $\Omega \subset \mathbb{R}^d$ a Lipschitz domain and $d = 2$ or 3, we refer to [53] for more details.

3.2.1 Meshes

For numerical solution of our elliptic problem, we first need to introduce a triangulation of the domain Ω.

Definition 3.1 (Mesh, [56, Definition 8]). *Let Ω be a domain in $\mathbb{R}^d$. A mesh is a finite collection of disjoint non-degenerate simplices $\mathcal{T} = \{K_1, \ldots, K_{N_{\mathrm{el}}}\}$ such that*

- $\overline{\Omega} = \bigcup_{i=1}^{N_{\mathrm{el}}} \overline{K}_i$,
- *the intersection $\overline{K}_i \cap \overline{K}_j$ is either empty, a common point or a common edge of both K_i and K_j for every $1 \leq i, j \leq N_{\mathrm{el}}$ and $i \neq j$.*

Definition 3.2 (Meshsize, [53, Definition 1.13]). *For all $K \in \mathcal{T}$, h_K denotes the* diameter *of K*

$$\forall K \in \mathcal{T}, \quad h_K = \operatorname{diam} K = \max_{x,y \in K} |x - y|.$$

Then, we define the meshsize *of* $\mathcal{T}$ *as*

$$h := \max_{K \in \mathcal{T}} h_K.$$

We use the notation $\mathcal{T}_h$ for a mesh $\mathcal{T}$ of meshsize h.

Definition 3.3 (Shape regularity, [53, Definition 1.13])**.** *A family of meshes* $\{\mathcal{T}_h\}_{h>0}$ *is said to be* shape-regular *if there is* σ_0 *such that*

$$\forall h, \forall K \in \mathcal{T}_h, \quad \sigma_K = \frac{h_K}{\rho_K} \leq \sigma_0,$$

where ρ_K *is the radius of the largest ball that can be inscribed in* K*.*

Usually, the mesh $\mathcal{T}_h$ is generated using a *reference cell* K_{ref} and an affine geometric transformation mapping K_{ref} to the actual mesh cells. We denote this mapping F_K, so that $K = F_K(K_{\text{ref}})$.

3.2.2 Galerkin approximation

In the following, we will denote $\mathcal{V}_h$ a generic space of finite element functions associated with $\mathcal{T}_h$, N its dimension and $(\varphi_j)_{j=1}^N$ the finite element basis of $\mathcal{V}_h$. We need to write the variational formulation of Equation (3.1): find $u \in H_0^1(\Omega)$ such that

$$a(u, v) := \int_\Omega \nabla u(x) \cdot \nabla v(x) \, \mathrm{d}x = \int_\Omega f(x) v(x) \, \mathrm{d}x, \quad \forall v \in H_0^1(\Omega). \tag{3.6}$$

The Galerkin approximation of Equation (3.6) consists in looking for the solution in $\mathcal{V}_h \subset H_0^1(\Omega)$: find $u_h \in \mathcal{V}_h$ such that

$$a(v_h, w_h) = (f, w_h), \quad \forall w_h \in \mathcal{V}_h. \tag{3.7}$$

Then, from Equation (3.7), we obtain the following linear system

$$\mathbf{A}_h \mathbf{v}_h = \mathbf{b}_h, \tag{3.8}$$

where $(\mathbf{A}_h)_{i,j} = a(\varphi_j, \varphi_i)$, $(\mathbf{b}_h)_j = \int_\Omega f \varphi_j$ for $1 \leq i, j \leq N$ and $\mathbf{v}_h$ is the vector of coefficients corresponding to the unknown finite element function v_h. Notice that, since the bilinear form a is symmetric and positive definite, so is $\mathbf{A}_h$.

3.2.3 Domain decomposition

We now define the framework particular to DDM. We first partition Ω into a set of non-overlapping subdomains $(\Omega'_p)_{p=1}^n$ resolved by $\mathcal{T}_h$, using for example a graph partitioner such as METIS [105] or SCOTCH [25]. Then, each domain Ω'_p is extended to a domain Ω_p by adding one or several layers of mesh elements, such that $(\Omega_p)_{p=1}^n$ corresponds to an overlapping decomposition of Ω. This decomposition induces a natural decomposition of $\mathcal{V}_h$:

$$\widetilde{\mathcal{V}}_{h,p} := \operatorname{Span}(\varphi_j \,|\, \operatorname{supp}(\varphi_j) \cap \Omega_p \neq \emptyset \text{ and } 1 \leq j \leq N).$$

We also denote $N_p = \dim(\widetilde{\mathcal{V}}_{h,p})$. Notice that the $\widetilde{\mathcal{V}}_{h,p}$ corresponds to the set of functions whose support intersects Ω_p. It also means that their support is not necessarily contained in Ω_p, that

is to say $\Omega_p \subsetneq \cup_{\varphi\in\widetilde{\mathcal{V}}_{h,p}} \mathrm{supp}(\varphi)$ (see Figure 3.3). That is why, we introduce $\widetilde{\Omega}_p$ as the subdomain containing the support of all functions in $\widetilde{\mathcal{V}}_{h,p}$ and

$$\mathcal{V}_{h,p} := \mathrm{Span}(\varphi|_{\widetilde{\Omega}_p},\, \varphi \in \widetilde{\mathcal{V}}_{h,p}).$$

Besides, we define the set of degrees of freedom associated with $\mathcal{V}_{h,p}$: $\mathrm{dof}_{h,p} := \{j \,|\, \mathrm{supp}(\varphi_j) \cap \Omega_p \neq \emptyset\}$, and we choose an arbitrary local numbering of the degrees of freedom defined by $\sigma_p : \{1, 2, \ldots, N_p\} \to \mathrm{dof}_{h,p}$. Then, we can define the extension by zero of a function $v \in \mathcal{V}_{h,p}$ to Ω that lies in $\mathcal{V}_h$. The corresponding operator is denoted by

$$\mathrm{R}_p^T : \mathcal{V}_{h,p} \to \mathcal{V}_h.$$
$$\mathrm{R}_p^T(u_h^p)(\mathbf{x}) = \begin{cases} u_h^p(\mathbf{x}) & \text{if } \mathbf{x} \in \widetilde{\Omega}_p, \\ 0 & \text{otherwise.} \end{cases}$$

We also denote by $\mathbf{R}_p^T \in \mathbb{R}^{N\times N_p}$ a matrix representation of R_p^T defined as follows

$$(\mathbf{R}_p^T)_{j,k} = \begin{cases} 1 & \text{if } j = \sigma_p(k), \\ 0 & \text{otherwise.} \end{cases}$$

Thus, we can define the algebraic restriction $\mathbf{R}_p \in \mathbb{R}^{N_p\times N}$, the dual of $\mathbf{R}_p^T$ using the usual euclidean scalar product and its continuous counterpart $\mathrm{R}_p : \mathcal{V}_h \to \mathcal{V}_{h,p}$

$$\mathrm{R}_p : \mathcal{V}_h \to \mathcal{V}_{h,p}.$$
$$\sum_{j=1}^{N} u_{h,j}\varphi_j \mapsto \sum_{j=1}^{N_p} u_{h,\sigma_p(j)}\varphi_{\sigma_p(j)}|_{\widetilde{\Omega}_j}$$

Note that R_p is *not* a restriction in the natural sense. It restricts the set of shape functions used to represent an element of the discrete space $\mathcal{V}_{h,p}$ (see remark thereafter). Notice also that R_p^T is not the dual of R_p for the L^2 scalar product, they are respectively the equivalent of $\mathbf{R}_p^T$ and $\mathbf{R}_p$ for finite element functions, and $\mathbf{R}_p^T$ is the transpose of $\mathbf{R}_p$.

Remarks 3.4.

- *Let us take an example and consider the case where* Γ *is the interval between* 0 *and* 10 *on the x-axis discretized using* $\mathbb{P}_1$ *Lagrange elements* $\mathcal{V}_h = \mathrm{Span}(\varphi_j \,|\, 0 \leq j \leq 10)$. *Defining* $\mathrm{dof}_{h,1} := \{0, 1, 2, 3, 4, 5, 6\}$ *and* $\mathrm{dof}_{h,2} := \{4, 5, 6, 7, 8, 9, 10\}$, *we represent* $u_h = 1$ *in Figure 3.3. Notice that,* $\mathrm{R}_1(u_h)$ *is not the usual restriction of* u_h *since it is linear between* 6 *and* 7.

- *Another example in 2D is given Figure 3.4 for a square with* $\mathbb{P}_1$ *shape function. The nodes in* $\mathrm{dof}_{h,1}$ *are represented with rectangle nodes.*

- *A minimal overlap of one mesh cell corresponds to algebraic restriction and extension operators without algebraic overlap (see Figure 3.3).*

We have the relations $\mathbf{R}_p\mathbf{R}_p^T = \mathbf{I}_d \in \mathbb{R}^{N_p\times N_p}$ and $\mathrm{R}_p\,\mathrm{R}_p^T = \mathrm{I_d}$. We also define the following

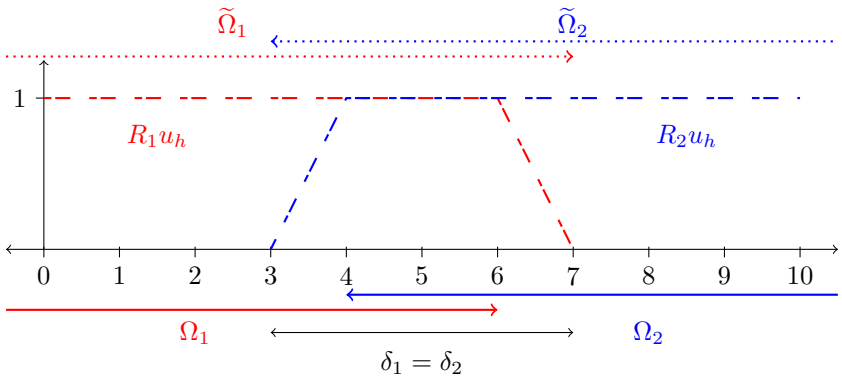


Figure 3.3 – Example of an overlap in 1D for $\mathbb{P}_1$ finite elements with $\Omega = \Omega_1 \cup \Omega_2$ and $u_h \in \mathcal{V}_h$ such that $u_h = 1$, that is to say, $u_{h,j} = 1$ for $1 \leq j \leq N$.

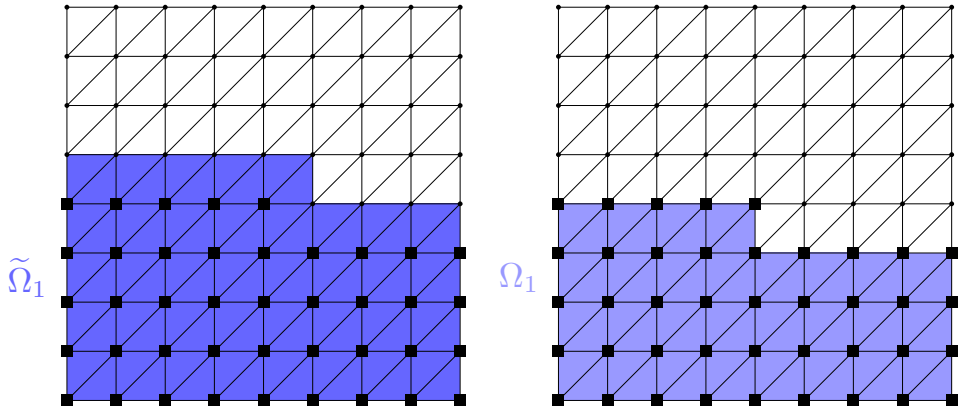


Figure 3.4 – Example of a subdomain in 2D for $\mathbb{P}_1$ finite elements where the rectangle nodes denote $\mathbb{P}_1$ functions in $\mathcal{V}_{h,1}$.

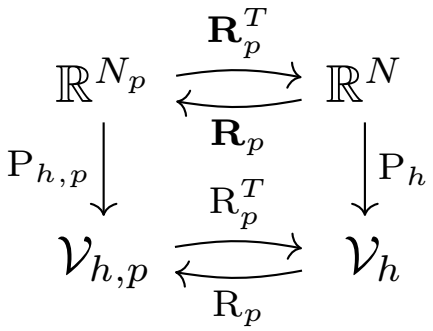


Figure 3.5 – Relations between restriction, extension operators, P_h and $\mathrm{P}_{h,p}$ for finite element functions.

linear applications

$$\begin{cases} \mathrm{P}_h : \mathbb{R}^N \to \mathcal{V}_h, \\ \mathbf{u}_h = (u_{h,j})_{j=1}^N \mapsto u_h = \displaystyle\sum_{j=1}^N u_{h,j}\varphi_j, \end{cases} \quad \text{and} \quad \begin{cases} \mathrm{P}_{h,p} : \mathbb{R}^{N_p} \to \mathcal{V}_{h,p}, \\ \mathbf{u}_h^p = (u_{h,j}^p)_{j=1}^{N_p} \mapsto u_h^p = \displaystyle\sum_{j=1}^{N_p} u_{h,j}^p \varphi_{\sigma_p(j)}, \end{cases}$$

that allow going back and forth between the finite element and the algebraic point of view. Several relations exist between the operators introduced so far, they are summarized in Figure 3.5.

We will also need partitions of unity that we define as an operator [143] and a function [140]. We will mainly use the former so that we only give the properties of the latter and refer to [140] for more details.

Definition 3.5 (Partition of unity — function)**.** *In [140, Proposition 2.1], the authors defined a partition of the unity* $\{\chi_p\}_{p=1}^n$ *using a coarse mesh and the nodal interpolant* I_h *defined as*

$$\mathrm{I}_h(f)(x) := \sum_{j=1}^N \varphi_j(x) f(x_j), \quad \forall f \in C^0(\Omega),$$

with $\{x_j\}_{j=1}^N$ *the vertices of the mesh, so that* $\varphi_j(x_i) = \delta_{i,j}$*. This partition of unity as the following properties*

$$0 \le \chi_p \le 1, \quad \text{and} \quad |\nabla \chi_p| \lesssim H_p^{-1}, \quad \text{with } H_p = \operatorname{diam}(\widetilde{\Omega}_p).$$

Definition 3.6 (Partition of unity — operator)**.** *For each degree of freedom* $1 \le j \le N$*, we define a family of weights* $\{\mu_{p,j} \,|\, j \in \mathrm{dof}_{h,p}, 1 \le p \le n\}$ *for* $1 \le p \le N_p$ *such that*

$$\mu_{p,j} \ge 1 \quad \text{and} \quad \sum_{\{p | j \in \mathrm{dof}_{h,p}\}} \frac{1}{\mu_{p,j}} = 1.$$

Then, we define the local partition of unity operator for the subdomain p as follows

$$\Xi_p : \mathcal{V}_{h,p} \to \mathcal{V}_{h,p}$$
$$v_h = \sum_{j=1}^{N_p} v_{h,j}\varphi_{\sigma_p(j)}|_{\widetilde{\Omega}_p} \mapsto \sum_{j=1}^{N_p} \frac{1}{\mu_{p,\sigma_p(j)}} v_{h,j}\varphi_{\sigma_p(j)}|_{\widetilde{\Omega}_p}.$$

We also denote $\mathbf{D}_p$ its matrix representation, which is a diagonal matrix of size $N_p \times N_p$ whose elements are the $\mu_{p,\sigma_p(j)}$ for $1 \leq j \leq N_p$. We have the following identity:

$$\mathbf{I}_d = \sum_{p=1}^{n} \mathbf{R}_p^T \mathbf{D}_p \mathbf{R}_p. \tag{3.9}$$

A few constants will characterize the domain decomposition, they will appear in the study of the efficiency of DDM preconditioners. All these definitions are inspired by [45].

Definition 3.7.

- *k_0 denotes the maximum multiplicity of the interaction between subdomains plus one, i.e.,*

$$k_0 := \max_{1 \leq j \leq N} \#\left\{p \,|\, \mathbf{R}_p \mathbf{M} \mathbf{R}_j^T \neq 0\right\}, \tag{3.10}$$

  where $(\mathbf{M})_{i,j} = \int_\Omega \varphi_i(x)\varphi_j(x)\,\mathrm{d}x$ is the mass matrix.

- *k_1 denotes the maximal multiplicity of the subdomain intersection, i.e., the largest integer m such that there exist m different subdomains whose intersection has a nonzero measure.*

$$k_1 := \operatorname{ess\,sup}\left\{\sum_{p=1}^{n} 1_{\Omega_p}\right\} \tag{3.11}$$

- *N_c is the minimum number of colors we can use to color a decomposition such that any two domains sharing an overlap have different colors*

As for the domain decomposition, these constants are not supposed to increase with the number of subdomains a priori because they only depend on the local interaction between subdomains. The domain decomposition could be considered as a graph where subdomains are the vertices and then, two vertices are related by an edge if the corresponding subdomains overlap. Using the terminology from graph theory to describe this graph, $k_0 - 1$ corresponds to the maximum vertex degree [160, Definition 1.3.1], N_c corresponds to the chromatic number [160, Definition 5.1.1], and we have $N_c \leq k_0$ [160, Proposition 5.1.13]. Brook's theorem [160, Theorem 5.1.22] states that this bound can be improved to $N_c \leq k_0 - 1$ if the graph is connected [160, Definition 1.2.6] (meaning that there is always a path between two vertices) but not complete (meaning that each vertex is not connected to every other vertex) and not an odd cycle.

3.3 Schwarz methods at the discrete level

Discretization of Equations (3.2) and (3.3) using Lagrange $\mathbb{P}_1$ elements with a minimal overlap of one mesh cell (or without algebraic overlap, that is to say $\mathrm{dof}_{h,p} \cap \mathrm{dof}_{h,l} = \emptyset$ for $p \neq l$), lead respectively to a Gauss-Seidel method and a Block Jacobi method (see [45, Section 1.2]), and

thus both define a preconditioner. In practice, they are not necessarily used in a fixed-point method, Krylov solvers are usually preferred to them because they are more efficient.

The *Additive Schwarz Method (ASM)*, introduced in [162], refers to the preconditioner $\mathbf{M}_{\text{ASM}}^{-1}$ which stems from the discretization of (3.3) in the case of a minimal overlap

$$\mathbf{M}_{\text{ASM}}^{-1} := \sum_{p=1}^{n} \mathbf{R}_p^T (\mathbf{R}_p \mathbf{A} \mathbf{R}_p^T)^{-1} \mathbf{R}_p, \tag{3.12}$$

but its definition is extended to the case of a more general overlap.

Remark 3.8. *The preconditioner which stems from the discretization of Equation* (3.2) *is usually called* Multiplicative Schwarz Method (MSM). *Historically, the ASM has been introduced as a modification of the MSM such that it became additive in [162, 122]. In the case of a general overlap, the ASM does not correspond to the discretization of Equation* (3.3) *anymore.*

The use of Krylov solvers instead of Block Jacobi iterations is even more justified by the fact that the ASM preconditioner does not necessarily converge with the latter. Indeed, it has been shown for example in [47, Theorem 3.5] that, for a Poisson problem with two subdomains, the method does not converge in the overlap. In [122] a damping factor "sufficiently small" is used to make the method converge (see [47] for an insight on the value of damping parameter). This shows that the ASM does not correspond to the Parallel Schwarz Method since the latter always converges in this case, cf. [112]. This last remark led to the introduction of the *Restricted Additive Schwarz method (RAS)* (see [47]) which corresponds to the discretization of Equation (3.3). Nevertheless, the ASM preconditioner has very desirable properties, it is symmetric when $\mathbf{A}$ is symmetric, which is not the case with RAS, and thanks to its additive expression, it is tailored for parallelization, which is not the case of the MSM that is sequential by nature.

3.3.1 Fictitious space lemma

Since we want to use the ASM preconditioner with Krylov solver, typically the conjugate gradient, we need an estimate on the condition number of the preconditioned system. To do so, we will use the following abstract theorem [129, 72]:

Theorem 3.9 (Fictitious space lemma)**.** *Let H and H_D be two Hilbert spaces, respectively with the scalar product $(\cdot,\cdot)$ and $(\cdot,\cdot)_D$. Let the symmetric positive definite bilinear forms $\widetilde{a} : H \times H \to \mathbb{R}$ and $\widetilde{b} : H_D \times H_D \to \mathbb{R}$ generated by symmetric positive definite operators $A : H \to H$ and $B : H_D \to H_D$ respectively. Suppose that there exists a linear operator $\mathcal{R} : H_D \to H$ such that the following hold:*

(i) $\mathcal{R}$ is surjective.

(ii) There exists a positive constant c_R such that

$$\widetilde{a}(\mathcal{R}u_D, \mathcal{R}u_D) \leq c_R \widetilde{b}(u_D, u_D) \quad \forall u_D \in H_D. \tag{3.13}$$

(iii) There exists a positive constant c_T such that for all $u \in H$ there exists $u_D \in H_D$ with $\mathcal{R}u_D = u$ and

$$c_T \widetilde{b}(u_D, u_D) \leq \widetilde{a}(\mathcal{R}u_D, \mathcal{R}u_D) = \widetilde{a}(u, u) \tag{3.14}$$

And let us introduce the adjoint operator $\mathcal{R}^* : H \to H_D$ *by* $(\mathcal{R}u_D, u) = (u_D, \mathcal{R}^* u)_D$ *for all* $u_D \in H_D$ *and* $u \in H$*. Then, we have the spectral estimate*

$$c_T \widetilde{a}(u,u) \leq \widetilde{a}(\mathcal{R}B^{-1}\mathcal{R}^* Au, u) \leq c_R \widetilde{a}(u,u), \quad \forall u \in H,$$

which shows that the spectrum of the operator $\mathcal{R}B^{-1}\mathcal{R}^* A$ *are bounded from below by* c_T *and from above by* c_R*.*

Remark 3.10. *The estimate in Equation* (3.14) *is usually referred to as* stable decomposition.

3.3.2 Analysis without coarse space

Let us reformulate the ASM in the framework of the fictitious space lemma:

Definition 3.11.

- $H := \mathbb{R}^N$ *is endowed with the standard euclidean scalar product* $(\cdot,\cdot)$*, and we consider the following bilinear form:*

$$\begin{aligned} \widetilde{a} : \mathbb{R}^N \times \mathbb{R}^N &\to \mathbb{R} \\ (\mathbf{u}_h, \mathbf{v}_h) &\mapsto (\mathbf{A}\mathbf{u}_h, \mathbf{v}_h) = a(u_h, v_h), \end{aligned}$$

  with $\mathbf{A}$ *defined in Equation* (3.8) *while* $u_h = \mathrm{P}_h(\mathbf{u}_h)$ *and* $v_h = \mathrm{P}_h(\mathbf{v}_h)$*. In the following, we will often use this vector representation in the finite element basis (bold font) of finite element functions (normal font).*

- $H_D := \prod_{p=1}^n \mathbb{R}^{N_p}$ *is endowed with the standard euclidean scalar product, and we consider the following bilinear form:*

$$\begin{aligned} \widetilde{b} : \prod_{p=1}^n \mathbb{R}^{N_p} \times \prod_{p=1}^n \mathbb{R}^{N_p} &\to \mathbb{R} \\ (\mathbf{u}_h^p, \mathbf{v}_h^p)_{p=1}^n &\mapsto \sum_{p=1}^n (\mathbf{A}\mathbf{R}_p^T \mathbf{u}_h^p, \mathbf{R}_p^T \mathbf{v}_h^p) = \sum_{p=1}^n a(\mathrm{R}_p^T\, u_h^p, \mathrm{R}_p^T\, v_h^p), \end{aligned}$$

  with $u_h^p = \mathrm{P}_{h,p}(\mathbf{u}_h^p)$ *and* $v_h^p = \mathrm{P}_{h,p}(\mathbf{v}_h^p)$*. Then,* $B : H_D \to H_D$ *is defined by*

$$\forall\, \mathbf{U}_h = (\mathbf{u}_h^p)_{p=1}^n \in \prod_{p=1}^n \mathbb{R}^{N_p}, \quad B(\mathbf{U}_h) := (\mathbf{R}_p \mathbf{A} \mathbf{R}_p^T \mathbf{u}_h^p)_{1 \leq p \leq n}.$$

  Notice that B *is a block diagonal operator, such that*

$$\forall\, \mathbf{U}_h = (\mathbf{u}_h^p)_{p=1}^n \in \prod_{p=1}^n \mathbb{R}^{N_p}, \quad B^{-1}(\mathbf{U}_h) := ((\mathbf{R}_p \mathbf{A} \mathbf{R}_p^T)^{-1} \mathbf{u}_h^p)_{1 \leq p \leq n}.$$

- *The linear operator* $\mathcal{R}_{\mathrm{ASM}}$ *is defined as follows*

$$\begin{aligned} \mathcal{R}_{\mathrm{ASM}} : H_D &\to H \\ (\mathbf{u}_h^p)_{1 \leq p \leq n} &\mapsto \sum_{p=1}^n \mathbf{R}_p^T \mathbf{u}_h^p. \end{aligned} \tag{3.15}$$

We have now all the ingredients to study the ASM in the framework of the fictitious space lemma. First, since $\mathbf{M}_{\mathrm{ASM}}^{-1} = \mathcal{R}_{\mathrm{ASM}} B^{-1} \mathcal{R}_{\mathrm{ASM}}^{*}$ according to Equation (3.12) and the previous framework, we just need to check the hypothesis of Theorem 3.9 to obtain the needed estimate.

Lemma 3.12 (Surjectivity of $\mathcal{R}_{\mathrm{ASM}}$)**.** *The operator $\mathcal{R}_{\mathrm{ASM}}$ defined in Equation (3.15) is surjective.*

Proof. For all $\mathbf{u}_h \in \mathbb{R}^N$, $\mathbf{u}_h = \mathcal{R}_{\mathrm{ASM}}(\mathbf{U}_h)$ with $\mathbf{U}_h = (\mathbf{D}_p \mathbf{R}_p \mathbf{u}_h)_{1 \leq p \leq n}$ according to Equation (3.9). □

Lemma 3.13 (Continuity of $\mathcal{R}_{\mathrm{ASM}}$)**.** *For $\mathbf{U}_h = (\mathbf{u}_h^p)_{1 \leq p \leq n} \in H_D$, we have the following inequality:*

$$\widetilde{a}(\mathcal{R}_{\mathrm{ASM}} \mathbf{U}_h, \mathcal{R}_{\mathrm{ASM}} \mathbf{U}_h) \leq k_0 \widetilde{b}(\mathbf{U}_h, \mathbf{U}_h),$$

where k_0 is defined in Equation (3.10) as the maximum number of neighbors a subdomain can have.

Proof. The proof is the same as [45, Lemma 7.9]. □

The previous lemma shows that Equation (3.13) is satisfied. Let us see now why the ASM with only one-level can be insufficient in practical computations.

Lemma 3.14 (Stable decomposition)**.** *Let us assume that $\mathcal{V}_h$ is the standard space of continuous and piecewise linear functions. For $\mathbf{u}_h \in \mathbb{R}^N$, let $\mathbf{U}_h = (\mathbf{u}_h^p)_{1 \leq p \leq n} \in H_D$ such that $\mathbf{u}_h^p = \mathbf{R}_p \, \mathrm{P}_h^{-1} \, \mathrm{I}_h \, \chi_p \, \mathrm{P}_h \, \mathbf{u}_h$ with $(\chi_p)_{p=1}^n$ and I_h, respectively a partition of the unity and the nodal interpolant, defined in Definition 3.5. Then, we have $\mathbf{u}_h = \mathcal{R}_{\mathrm{ASM}}(\mathbf{U}_h)$ and*

$$\widetilde{b}(\mathbf{U}_h, \mathbf{U}_h) \leq 2 k_1 C (1 + H^2 \max_{p=1 \ldots n} \frac{1}{H_p^2}) \widetilde{a}(\mathbf{u}_h, \mathbf{u}_h)$$

where k_1 is the maximal multiplicity of the subdomain intersections defined in Equation (3.11), H is the diameter of Ω, H_p is the diameter of the subdomain $\widetilde{\Omega}_p$, $C > 0$ is a constant independent of h and that only depends on the shape of Ω, but not on its size.

Proof. Denoting $u_h = \mathrm{P}_h(\mathbf{u}_h)$ and $u_h^p = \mathrm{P}_{h,p}(\mathbf{u}_h^p)$, notice that we have the following relation:

$$\begin{aligned}
(\mathbf{A} \mathbf{R}_p^T \mathbf{u}_h^p, \mathbf{R}_p^T \mathbf{u}_h^p) &= a(\mathrm{P}_h \, \mathbf{R}_p^T \mathbf{u}_h^p, \mathrm{P}_h \, \mathbf{R}_p^T \mathbf{u}_h^p) \\
&= a(\mathrm{P}_h \, \mathbf{R}_p^T \mathbf{R}_p \, \mathrm{P}_h^{-1} \, \mathrm{I}_h(\chi_p u_h), \mathrm{P}_h \, \mathbf{R}_p^T \mathbf{R}_p \, \mathrm{P}_h^{-1} \, \mathrm{I}_h(\chi_p u_h)) \\
&= a(\mathrm{I}_h(\chi_p u_h), \mathrm{I}_h(\chi_p u_h)).
\end{aligned}$$

Using the fact that I_h applied to piecewise quadratic functions is stable in norm H_0^1 (see [140, Lemma 2.3]), that is to say

$$a(\mathrm{I}_h(v_h), \mathrm{I}_h(v_h)) \leq C_{\mathrm{inter}} a(v_h, v_h),$$

for every $v_h \in \mathcal{V}_h$ with C_{inter} is independent of h and the diameter of Ω. Taking $v_h = \chi_p u_h$, we

obtain

$$(\mathbf{A}\mathbf{R}_p^T\mathbf{u}_h^p, \mathbf{R}_p^T\mathbf{u}_h^p) \leq C_{\text{inter}} \int_{\widetilde{\Omega}_p} |\nabla(\chi_p u_h)|^2$$
$$\leq 2C_{\text{inter}} \left(\int_{\widetilde{\Omega}_p} |\nabla u_h|^2 + \frac{C_p}{H_p^2} |u_h|^2 \right)$$

using $|\nabla\chi_p|^2 \leq C_p/H_p^2$, where C_p is a constant (see Definition 3.5) and H_p is the diameter of the subdomain $\widetilde{\Omega}_p$. Summing over p, we obtain

$$\sum_{p=1}^{n} (\mathbf{A}\mathbf{R}_p^T\mathbf{u}_h^p, \mathbf{R}_p^T\mathbf{u}_h^p) \leq 2k_1 C_{\text{inter}} \int_\Omega |\nabla u_h|^2 + 2k_1 C_{\text{inter}} \max_{p=1\ldots n} \left(\frac{C_p}{H_p}^2 \right) \int_\Omega |u_h|^2.$$

Then, using the following Poincaré inequality [153, Corollary A.15], we have

$$\int_\Omega |u_h|^2 \leq C_{\text{P}} H^2 \int_\Omega \nabla |u_h|^2,$$

where C_{P} depends only on the shape of Ω, but not on its size, so that we obtain

$$\widetilde{b}(\mathbf{U}_h, \mathbf{U}_h) = \sum_{p=1}^{n} (\mathbf{A}\mathbf{R}_p^T\mathbf{u}_h^p, \mathbf{R}_p^T\mathbf{v}_h^p) \leq 2k_1 C \left(1 + H^2 \max_{p=1\ldots n} \frac{1}{H_p^2} \right) a(u_h, u_h)$$

where H is the diameter of Ω, $C > 0$ is a constant that depends only on the shape of Ω but not on its size and k_1 is the maximal multiplicity of the subdomain intersection. Finally, notice that

$$\mathcal{R}_{\text{ASM}}(\mathbf{U}_h) = \sum_{p=1}^{n} \mathbf{R}_p^T\mathbf{u}_h^p = \sum_{p=1}^{n} \mathbf{R}_p^T\mathbf{R}_p \, \mathrm{P}_h^{-1} \, \mathrm{I}_h \, \chi_p \, \mathrm{P}_h \, \mathbf{u}_h$$
$$= \mathrm{P}_h^{-1} \, \mathrm{I}_h \sum_{p=1}^{n} \chi_p \, \mathrm{P}_h \, \mathbf{u}_h = \mathrm{P}_h^{-1} \, \mathrm{I}_h \, \mathrm{P}_h \, \mathbf{u}_h.$$

Applying I_h to a piecewise linear function like $\mathrm{P}_h \, \mathbf{u}_h$ reduces to the identity operator, so that we obtain $\mathcal{R}_{\text{ASM}}(\mathbf{U}_h) = \mathbf{u}_h$. □

Theorem 3.15 (Spectral estimate for the ASM)**.** *We have*

$$\kappa(\mathbf{M}_{\text{ASM}}^{-1}\mathbf{A}) \leq 2k_0k_1C(1 + H^2 \max_{p=1\ldots n} \frac{1}{H_p^2}), \tag{3.16}$$

where k_1 is the maximal multiplicity of the subdomain intersections defined inEquation (3.11), k_0 is defined inEquation (3.10) as the maximum number of neighbors a subdomain can have, H_p is the diameter of the subdomain $\widetilde{\Omega}_p$, H is the diameter of Ω, C is a constant that depends only on the shape of Ω but not on its size.

Proof. Since $\mathbf{M}_{\text{ASM}}^{-1} = \mathcal{R}_{\text{ASM}} B^{-1} \mathcal{R}_{\text{ASM}}^*$, the framework introduced previously needs to satisfy the three hypothesis of Theorem 3.9. The first is satisfied according to Lemma 3.12. The second is given by Lemma 3.13 with $c_R = k_0$. And finally, the last hypothesis is given by Lemma 3.14 with $c_T^{-1} = 2k_1C(1 + H^2 \max_{p=1\ldots n}(1/H_p^2))$. □

Let us define two notions of scalability (cf. [45, Chapter 4]).

Definition 3.16 (Strong scalability)**.** *Strong scalability is defined as how the solution time varies with the number of cores for a fixed total problem size. Ideally, the elapsed time is inversely proportional to the number of processing units (CPUs, cores, GPUs).*

Definition 3.17 (Weak scalability)**.** *Weak scalability is defined as how the solution time varies with the number of cores for a fixed problem size per core. Ideally, the elapsed time is constant for a fixed ratio between the size of the problem and the number of processing units (CPUs, cores, GPUs).*

The previous estimate given in Equation (3.16) does not guarantee any scalability:

- If we increase the number of subdomains for a fixed domain Ω, so that the size of the subdomains decrease, we have $H/H_p \to \infty$.
- If we increase the number of subdomains and the size of the domain with the same ratio, so that the size of the subdomains is fixed but the size of Ω increases, we also have $H/H_p \to \infty$.

In both cases, if our bound is optimal, the condition number will increase so that the number of iterations using CG will also increase. So, the method is not scalable, both weakly and strongly.

And indeed, the method is not scalable, this is due to the fact that the solution is non-local, its value at one point depends on the right-hand side at any other point. And since the ASM performs communications between neighbors, we understand that the method becomes slower when the number of subdomains is large. This is observed in actual numerical experiments, see [45, Chapter 4] for examples of this phenomenon.

3.3.3 Analysis with GenEO coarse space

To solve the lack of scalability of the ASM, a mechanism which performs global communications can be added. It is usually done via a coarse space correction, i.e. we want to decompose $\mathcal{V}_h$ as follows

$$\mathcal{V}_h = \mathrm{R}_0^T \mathcal{V}_{h,0} + \sum_{p=1}^{n} \mathrm{R}_p^T \mathcal{V}_{h,p},$$

where $\mathcal{V}_{h,0}$ is a coarse space and $\mathrm{R}_0^T : \mathcal{V}_{h,0} \to \mathcal{V}_h$, an interpolation operator. This coarse correction can be added to the ASM as follows

$$\mathbf{M}_{\mathrm{ASM},2}^{-1} := \mathbf{R}_0^T (\mathbf{R}_0 \mathbf{A} \mathbf{R}_0^T)^{-1} \mathbf{R}_0 + \sum_{p=1}^{n} \mathbf{R}_p^T \mathbf{A}_{p,p}^{-1} \mathbf{R}_p, \tag{3.17}$$

but there exist several other ways to add a coarse correction (see [152] for example). The idea behind this coarse space is to directly take care of some parts of the function space $\mathcal{V}_h$. For example, we see in the proof of Lemma 3.14 that the reason why the Poincaré constant of the whole domain appears is that we cannot apply the Poincaré inequality locally. And this is because the functions under consideration are not in $H_0^1(\widetilde{\Omega}_p)$. To solve this, we can remove a coarse component that contains at least the constants, so that we can apply local Poincaré inequalities to a zero mean value functions instead. This argument is called a *quotient space* argument (see [153, Chapter 3.5] for more details).

The coarse spaces proposed in the literature generally rely on the kernel of the local solvers, for example, constants for Poisson problems and rigid body motions for elasticity. This is in part

due to the fact that they are necessary in substructuring methods for the local problem to be well-posed. One can also enlarge the coarse space to obtain more robust coarse spaces at the price of more expensive computations to deal with the coarse component. We refer to [161, 118] for historical reviews.

Here, we want to use the GenEO method (see [143] for the initial method and [45] for its present version) that builds a coarse space tailored for the condition number required by the user. In the analysis of the GenEO method, the coarse space is not initially prescribed. We first define a generic coarse space $\mathcal{V}_{h,0}$ and we denote its size by N_0. The coarse space is spanned by the columns of a rectangular matrix $\mathbf{R}_0^T$ of size $N \times N_0$. Actually, most of the assumptions of Theorem 3.9 are satisfied without any a priori knowledge of $\mathcal{V}_{h,0}$.

We formulate the ASM with GenEO coarse space as follows

Definition 3.18.

- *$H := \mathbb{R}^N$ is endowed with the standard euclidean scalar product $(\cdot,\cdot)$, and bilinear form $\widetilde{a}$ is defined as in Definition 3.11.*
- *$H_D := \prod_{p=0}^{n} \mathbb{R}^{N_p}$ is endowed with the standard euclidean scalar product, and we consider the following bilinear form:*

$$\begin{aligned}\widetilde{b} : \prod_{p=0}^{n} \mathbb{R}^{N_p} \times \prod_{p=0}^{n} \mathbb{R}^{N_p} &\to \mathbb{R} \\ (\mathbf{u}_h^p, \mathbf{v}_h^p)_{p=0}^n &\mapsto \sum_{p=0}^{n} (\mathbf{A}\mathbf{R}_p^T \mathbf{u}_h^p, \mathbf{R}_p^T \mathbf{v}_h^p) = (\mathbf{A}\mathbf{R}_0^T \mathbf{u}_h^0, \mathbf{R}_0^T \mathbf{v}_h^0) \\ &\qquad + \sum_{p=1}^{n} a(\mathrm{R}_p^T u_h^p, \mathrm{R}_p^T v_h^p),\end{aligned}$$

  with $u_h^p = \mathrm{P}_{h,p}\, \mathbf{u}_h^p$ and $v_h^p = \mathrm{P}_{h,p}\, \mathbf{v}_h^p$ for $1 \le p \le n$. Then, $B : H_D \to H_D$ is defined by

$$\forall\, \mathbf{U}_h = (\mathbf{u}_h^p)_{p=0}^n \in \prod_{p=0}^{n} \mathbb{R}^{N_p}, \quad B(\mathbf{U}_h) := (\mathbf{R}_p \mathbf{A} \mathbf{R}_p^T \mathbf{u}_h^p)_{0 \le p \le n}.$$

  Notice that B is a block diagonal operator,

$$\forall\, \mathbf{U}_h = (\mathbf{u}_h^p)_{p=0}^n \in \prod_{p=0}^{n} \mathbb{R}^{N_p}, \quad B^{-1}(\mathbf{U}_h) := ((\mathbf{R}_p \mathbf{A} \mathbf{R}_p^T)^{-1} \mathbf{u}_h^p)_{0 \le p \le n}.$$

- *The linear operator $\mathcal{R}_{\mathrm{ASM},2}$ is defined as follows*

$$\begin{aligned}\mathcal{R}_{\mathrm{ASM},2} : H_D &\to H \\ (\mathbf{u}_h^p)_{0 \le p \le n} &\mapsto \sum_{p=0}^{n} \mathbf{R}_p^T \mathbf{u}_h^p.\end{aligned} \tag{3.18}$$

With these definitions, we have $\mathbf{M}_{ASM,2}^{-1} = \mathcal{R}_{\mathrm{ASM},2} B^{-1} \mathcal{R}_{\mathrm{ASM},2}^*$. We just need to satisfy the three hypothesis of Theorem 3.9 using Definition 3.18.

Lemma 3.19 (Surjectivity of $\mathcal{R}_{\mathrm{ASM},2}$, [45, Lemma 7.10])**.** *The operator $\mathcal{R}_{\mathrm{ASM},2}$ defined in Equation (3.18) is surjective.*

Proof. For all $\mathbf{u}_h \in \mathbb{R}^N$, $\mathbf{u}_h = \mathcal{R}_{\text{ASM}}(\mathbf{U}_h)$ with $\mathbf{U}_h = (0, (\mathbf{D}_p \mathbf{R}_p \mathbf{u}_h)_{1 \leq p \leq n})$ according to Equation (3.9). □

Lemma 3.20 (Continuity of $\mathcal{R}_{\text{ASM},2}$, [45, Lemma 7.11])**.** *For $\mathbf{U}_h = (\mathbf{u}_h^p)_{0 \leq p \leq n} \in H_D$, we have the following inequality:*

$$\widetilde{a}(\mathcal{R}_{\text{ASM},2}\mathbf{U}_h, \mathcal{R}_{\text{ASM},2}\mathbf{U}_h) \leq 2k_0 \widetilde{b}(\mathbf{U}_h, \mathbf{U}_h),$$

where k_0 is defined in Equation (3.10) as the maximum number of neighbors a subdomain can have.

Proof. First, we use the Cauchy-Schwarz inequality

$$\begin{aligned}\widetilde{a}(\mathcal{R}_{\text{ASM},2}\mathbf{U}_h, \mathcal{R}_{\text{ASM},2}\mathbf{U}_h) &= \left(\mathbf{A}\left(\sum_{p=0}^{n} \mathbf{R}_p^T \mathbf{u}_h^p\right), \sum_{p=0}^{n} \mathbf{R}_p^T \mathbf{u}_h^p\right) \\ &\leq 2\left((\mathbf{A}\mathbf{R}_0^T \mathbf{u}_h^0, \mathbf{R}_0^T \mathbf{v}_h^0) + a\left(\sum_{p=1}^{n} \mathrm{R}_p^T\, u_h^p, \sum_{p=1}^{n} \mathrm{R}_p^T\, v_h^p\right)\right),\end{aligned}$$

and then we use Lemma 3.13 with the last term to obtain the expected result. □

Then, notice that we have the following lemma for any additive coarse space.

Lemma 3.21 ([45, Lemma 7.12])**.** *Let $\mathbf{u}_h \in H$ and $\mathbf{U}_h = (\mathbf{u}_h^p)_{0 \leq p \leq n} \in H_D$ with $\mathcal{R}_{\text{ASM},2}(\mathbf{U}_h) = \mathbf{u}_h$. Then, we have*

$$\widetilde{b}(\mathbf{U}_h, \mathbf{U}_h) \leq 2\widetilde{a}(\mathbf{u}_h, \mathbf{u}_h) + (2k_0 + 1) \sum_{p=1}^{N} \widetilde{a}(\mathbf{R}_p^T \mathbf{u}_h^p, \mathbf{R}_p^T \mathbf{u}_h^p).$$

Proof. First, we use the definition of $\mathcal{R}_{\text{ASM},2}$ and the Cauchy-Schwarz inequality,

$$\begin{aligned}\widetilde{b}(\mathbf{U}_h, \mathbf{U}_h) &= \widetilde{a}(\mathbf{R}_0^T \mathbf{u}_h^0, \mathbf{R}_0^T \mathbf{u}_h^0) + \sum_{p=1}^{n} \widetilde{a}(\mathbf{R}_p^T \mathbf{u}_h^p, \mathbf{R}_p^T \mathbf{u}_h^p) \\ &= \widetilde{a}(\mathbf{u}_h - \sum_{p=1}^{n} \mathbf{R}_p^T \mathbf{u}_h^p, \mathbf{u}_h - \sum_{p=1}^{n} \mathbf{R}_p^T \mathbf{u}_h^p) + \sum_{p=1}^{n} \widetilde{a}(\mathbf{R}_p^T \mathbf{u}_h^p, \mathbf{R}_p^T \mathbf{u}_h^p) \\ &\leq 2\left[\widetilde{a}(\mathbf{u}_h, \mathbf{u}_h) + \widetilde{a}\left(\sum_{p=1}^{n} \mathbf{R}_p^T \mathbf{u}_h^p, \sum_{p=1}^{n} \mathbf{R}_p^T \mathbf{u}_h^p\right)\right] + \sum_{p=1}^{n} \widetilde{a}(\mathbf{R}_p^T \mathbf{u}_h^p, \mathbf{R}_p^T \mathbf{u}_h^p).\end{aligned}$$

Then, we can apply Lemma 3.13 to the second term to obtain the result. □

Besides, we have the following lemma using our finite element setting.

Lemma 3.22 ([45, Lemma 7.13])**.** *For all $\mathbf{u}_h \in H$ we have*

$$\sum_{p=1}^{n} a_{\Omega_p}(\mathrm{P}_h\, \mathbf{u}_h, \mathrm{P}_h\, \mathbf{u}_h) \leq k_1 \widetilde{a}(\mathbf{u}_h, \mathbf{u}_h),$$

where a_{Ω_p} corresponds to the local bilinear form associated with Ω_p, that is to say, for $u, v \in H^1(\Omega)$

$$a_{\Omega_p}(u,v) = \int_{\Omega_p} \nabla u(\mathbf{x}) \cdot \nabla v(\mathbf{x}) \, \mathrm{d}\mathbf{x}.$$

Proof.

$$\sum_{p=1}^{n} a_{\Omega_p}(\mathrm{P}_h \, \mathbf{u}_h, \mathrm{P}_h \, \mathbf{u}_h) = \sum_{p=1}^{n} \int_{\Omega_p} |\nabla(\mathrm{P}_h \, \mathbf{u}_h)|^2 \, \mathrm{d}\mathbf{x} \leq k_1 \int_{\Omega} |\nabla(\mathrm{P}_h \, \mathbf{u}_h)|^2 \, \mathrm{d}\mathbf{x} = k_1 \widetilde{a}(\mathbf{u}_h, \mathbf{u}_h).$$

□

According to the last two lemmas, a sufficient condition for $\mathbf{U}_h = (\mathbf{u}_h^p)_{0 \leq p \leq n} \in H_D$ to satisfy the third hypothesis of Theorem 3.9 is to satisfy the following relation: for every $1 \leq p \leq n$,

$$(\mathbf{A}\mathbf{R}_p^T \mathbf{u}_h^p, \mathbf{R}_p^T \mathbf{u}_h^p) \leq \tau a_{\Omega_p}(\mathrm{P}_h \, \mathbf{u}_h, \mathrm{P}_h \, \mathbf{u}_h) = \tau(\widetilde{\mathbf{A}}_p \mathbf{R}_p \mathbf{u}_h, \mathbf{R}_p \mathbf{u}_h), \tag{3.19}$$

where $\tau > 0$ is a given parameter and $\widetilde{\mathbf{A}}_p$ is the finite element matrix associated with a_{Ω_p}. Notice that the condition given in Equation (3.19) is local. The definition of $\mathbf{U}_h$ for a given τ will be a consequence of the definition of the coarse space. To be able to find a $\mathbf{U}_h \in H_D$ that satisfies Equation (3.19), we solve the associated generalized eigenvalue problems: for $1 \leq p \leq n$, find $(\mathbf{v}_{h,k}^p, \lambda_k^p)$

$$\widetilde{\mathbf{P}}_p \mathbf{D}_p \mathbf{A}_{p,p} \mathbf{D}_p \widetilde{\mathbf{P}}_p \mathbf{v}_{h,k}^p = \lambda_k^p \widetilde{\mathbf{A}}_p \mathbf{v}_{h,k}^p \tag{3.20}$$

where $\widetilde{\mathbf{P}}_p$ is the projection on $\mathrm{range}(\widetilde{\mathbf{A}}_p)$ parallel to $\ker(\widetilde{\mathbf{A}}_p)$.

Definition 3.23 ([45, Lemma 7.7])**.** *Let $\tau > 0$, we define the following space related to the generalized eigenvalue problem defined in Equation (3.20):*

$$Z_{p,\tau} := \ker(\mathbf{B}_p) \cup \mathrm{Span}(\mathbf{v}_{h,k}^p \,|\, \textit{ for every } k \textit{ s.t. } \lambda_k^p > \tau),$$

Then, we also define the local projection π_p on $Z_{p,\tau}$ parallel to $\mathrm{Span}(\mathbf{v}_{h,k}^p \,|\, \lambda_k^p \leq \tau)$.

Then, according to [45, Lemma 7.7], we have the following theorem:

Theorem 3.24 ([45, Lemma 7.15])**.** *For every $1 \leq p \leq n$ and $\mathbf{v}_h^p \in \mathbb{R}^{N_p}$, we have*

$$(\mathbf{R}_p^T \mathbf{D}_p (\mathbf{I}_d - \pi_p) \mathbf{v}_h^p)^T \mathbf{A} (\mathbf{R}_p^T \mathbf{D}_p (\mathbf{I}_d - \pi_p) \mathbf{v}_h^p) \leq \tau (\mathbf{v}_h^p)^T \widetilde{\mathbf{A}}_p \mathbf{v}_h^p$$

Notice that the last relation is similar to Equation (3.19). That is why we define $\mathbf{u}_h^p$ for $1 \leq p \leq n$ as follows

$$\mathbf{u}_h^p := \mathbf{D}_p (\mathbf{I}_d - \pi_p) \mathbf{R}_p \mathbf{u}_h. \tag{3.21}$$

Then, it remains to define the coarse space $\mathcal{V}_{h,0}$ and the associated component $\mathbf{u}_h^0$ such that $\mathcal{R}_{\mathrm{ASM},2}(\mathbf{U}_h) = \mathbf{u}_h$ where $\mathbf{U}_h = (\mathbf{u}_h^p)_{p=0}^n$.

Definition 3.25 (GenEO coarse space, [45, Definition 7.16])**.** *The GenEO coarse space is defined as a sum of local contributions weighted with the partition of unity:*

$$\mathcal{V}_{h,0} = \mathrm{Span}(\mathbf{R}_p^T \mathbf{D}_p \mathbf{v}_h^p \,|\, 1 \leq p \leq N, \, \mathbf{v}_h^p \in Z_{p,\tau})$$

Let $\mathbf{Z}_\tau \in \mathbb{R}^{N\times N_0}$ be a column matrix so that $\mathcal{V}_{h,0}$ is spanned by its columns and $N_0 = \dim(\mathcal{V}_{h,0})$. We denote its transpose by $\mathbf{R}_0 := \mathbf{Z}_\tau^T$.

Theorem 3.26 (GenEO coarse component, [45, Theorem 7.17])**.** *Let $\mathbf{u}_h \in H$, we define $\mathbf{u}_h^p$ for $1 \le p \le n$ as in Equation (3.21) and $\mathbf{u}_h^0$ by*

$$\mathbf{u}_h^0 := (\mathbf{R}_0\mathbf{R}_0^T)^{-1}\mathbf{R}_0\left(\sum_{p=1}^n \mathbf{R}_p^T\mathbf{D}_p\pi_p\mathbf{R}_p\mathbf{u}_h\right), \tag{3.22}$$

Then, $\mathbf{U}_h = (\mathbf{u}_h^p)_{p=0}^n \in H_D$ is a stable decomposition of $\mathbf{u}_h$, that is to say

$$\mathcal{R}_{\mathrm{ASM},2}(\mathbf{U}_h) = \mathbf{u}_h \quad \text{and} \quad c_T\widetilde{b}(\mathbf{U}_h,\mathbf{U}_h) \le \widetilde{a}(\mathbf{u}_h,\mathbf{u}_h),$$

where $c_T^{-1} = (2 + (2k_0+1)k_1\tau)$.

Proof. Let $\mathbf{u}_h \in \mathbb{R}^N$ and $\mathbf{U}_h = (\mathbf{u}_h^p)_{0\le p\le n} \in H_D$ defined using Equations (3.21) and (3.22). First notice that we have

$$\mathbf{w}_0 = \mathbf{R}_0^T(\mathbf{R}_0\mathbf{R}_0^T)^{-1}\mathbf{R}_0\mathbf{w}_0$$

for every $\mathbf{w}_0 \in \mathcal{V}_{h,0}$ since $\mathbf{R}_0^T(\mathbf{R}_0\mathbf{R}_0^T)^{-1}\mathbf{R}_0$ is a projector on $\mathcal{V}_{h,0}$. Then, we have

$$\begin{aligned}
\mathcal{R}_{ASM,2}(\mathbf{U}_h) &= \sum_{p=0}^n \mathbf{R}_p^T\mathbf{u}_h^p = \mathbf{R}_0^T\mathbf{u}_h^0 + \sum_{p=1}^n \mathbf{R}_p^T\mathbf{u}_h^p \\
&= \mathbf{R}_0^T(\mathbf{R}_0\mathbf{R}_0^T)^{-1}\mathbf{R}_0\underbrace{\left(\sum_{p=1}^n \mathbf{R}_p^T\mathbf{D}_p\pi_p\mathbf{R}_p\mathbf{u}_h\right)}_{\in\mathcal{V}_{h,0}} + \sum_{p=1}^n \mathbf{R}_p^T\mathbf{u}_h^p \\
&= \sum_{p=1}^n \mathbf{R}_p^T\mathbf{D}_p\pi_p\mathbf{R}_p\mathbf{u}_h + \sum_{p=1}^n \mathbf{R}_p^T\mathbf{D}_p(\mathbf{I}_d - \pi_p)\mathbf{R}_p\mathbf{u}_h \\
&= \sum_{p=1}^n \mathbf{R}_p^T\mathbf{D}_p\mathbf{R}_p\mathbf{u}_h = \mathbf{u}_h,
\end{aligned}$$

where we used the fact that $\mathbf{D}_p$ defines a partition of unity in the last line. Then, using Lemma 3.21 and Theorem 3.24 with $\mathbf{v}_h^p = \mathbf{R}_p\mathbf{u}_h$ for $1 \le p \le n$, we have

$$\begin{aligned}
\sum_{p=0}^n a(\mathrm{P}_h\,\mathbf{R}_p^T\mathbf{u}_h^p, \mathrm{P}_h\,\mathbf{R}_p^T\mathbf{u}_h^p) &= \sum_{p=0}^n (\mathbf{A}_h\mathbf{R}_p^T\mathbf{u}_h^p, \mathbf{R}_p^T\mathbf{u}_h^p) \\
&\le 2(\mathbf{A}_h\mathbf{u}_h,\mathbf{u}_h) + (2k_0+1)\sum_{p=1}^n (\mathbf{A}_h\mathbf{R}_p^T\mathbf{u}_h^p, \mathbf{R}_p^T\mathbf{u}_h^p) \\
&\le 2(\mathbf{A}_h\mathbf{u}_h,\mathbf{u}_h) + (2k_0+1)\tau\sum_{p=1}^n (\widetilde{\mathbf{A}}_h\mathbf{R}_p\mathbf{u}_h, \mathbf{R}_p\mathbf{u}_h)
\end{aligned}$$

Finally, using Lemma 3.22 we deduce

$$\sum_{p=0}^{n} a(\mathrm{P}_h \, \mathbf{R}_p^T \mathbf{u}_h^p, \mathrm{P}_h \, \mathbf{R}_p^T \mathbf{u}_h^p) \leq (2 + (2k_0 + 1)\, \tau k_1)\, a(P_h \mathbf{u}_h, P_h \mathbf{u}_h).$$

□

Theorem 3.27 ([45, Theorem 7.18])**.** *We have the following condition number estimate*

$$\kappa(\mathbf{M}_{ASM,2}^{-1} \mathbf{A}) \leq 2k_0(2 + (2k_0 + 1)k_1\tau).$$

Proof. Again, we use Theorem 3.9. The first hypothesis is satisfied according to Lemma 3.19. The second is given by Lemma 3.20 with $c_R = 2k_0$. And finally, the last hypothesis is given by Theorem 3.26 such that $c_T^{-1} = (2 + (2k_0 + 1)k_1\tau)$. □

Remark 3.28. *It should be noted that the decomposition $(\mathbf{u}_h^p)_{p=1}^n$ is necessary for the analysis, but it does not need to be computed in practice. To use this coarse space, it is sufficient to compute $\mathbf{Z}_\tau$ and then $(\mathbf{R}_0 \mathbf{A}_h \mathbf{R}_0^T)^{-1}$ to be able to apply $\mathbf{M}_{ASM}^{-1}$ (see Equation (3.12)). And to do so, each local contribution to the coarse space given in Definition 3.23 can be computed independently in parallel, so that computing $\mathbf{Z}_\tau$ can be done efficiently. Then, to compute $\mathbf{R}_0 \mathbf{A}_h \mathbf{R}_0^T$, one can use the fact that $\mathbf{Z}_\tau$ is sparse and distributed (since the eigenvectors are computed in parallel). Finally, we need to invert $(\mathbf{R}_0 \mathbf{A}_h \mathbf{R}_0^T) \in \mathbb{R}^{N_0 \times N_0}$ and that is why a good coarse space should be of minimum size while still containing relevant information for the convergence.*

In conclusion, this approach requires the solution to local generalized eigenvalue problems, which is a parallel task, and we obtain a condition number independent of the number of subdomains, contrary to the one-level approach of Section 3.3.2. The advantage of this method over other methods using a two-level approach is that it can work in a black-box manner and it is more robust, in the sense that the coarse space is built automatically from the local generalized eigenvalue problems and its efficiency is proved as long as the problem is symmetric and positive definite. However the coarse space can be large so that inverting $\mathbf{R}_0 \mathbf{A}_h \mathbf{R}_0^T$ can be expensive at some point. In this case, approximate solvers can be used, and the effect of the approximation on the bound are studied in [127].

CHAPTER 4

Local multi-trace formulation and optimal Schwarz method

Outline of the current chapter

In Chapter 3, we presented two different points of view on Schwarz methods, either as iterative methods or as preconditioners. In the case of BEM, we could also think of two approaches for DDM, as pointed out in the introduction of [80]. We could make a decomposition of the original domain where the solution is sought, a *volume decomposition*, and then formulate a BIE in each subdomain with some ways of coupling them. Or we could first set up a BIE on the whole computational domain and then apply a domain decomposition of its boundary, a *surface decomposition*. In this chapter, we will explore the first approach and show a relation

with the first point of view introduced on Schwarz methods. This approach can be useful to solve multi-subdomain scattering problems.

Several formulations for this approach have been introduced where the difference lies usually in the way the BIE are coupled with each other: the PMCHWT (Poggio-Miller-Chang-Harrington-Wu-Tsai) formulation [131, 23, 163] also called Rumsey's reaction principle, the Boundary Element Tearing and Interconnecting (BETI) method [110] and the Multi-Trace Formulations (MTF) [38]. The latter consists in doubling unknown traces at each point of each interface, and these duplicated unknowns are disconnected a priori from one another. This allows solving independently problems in each subdomains, which makes MTF similar to DDM as presented in Chapter 3, while the transmission condition is enforced weakly with a transmission operator.

There exist several variants of MTF and they differ in the choice of the transmission operator. We will focus on local MTF [96] where the transmission consists in exchanging the traces of the neighbouring subdomains, which is local and similar to Schwarz methods like the alternating Schwarz method given in Equation (3.2). Interesting results about this formulation have been proved in [33] for Helmholtz with a constant material coefficient. The authors introduced a particular Jacobi method that led to a convergence in a finite number of iterations for two and three subdomains. This is similar to the Optimal Schwarz Method (OSM) [128] described in Section 3.1.2, and they actually proved an equivalence between their Jacobi method applied to local MTF and OSM in the case of two subdomains for $d = 1$.

In this chapter, we will first introduce the framework of multi-subdomain problems in Section 4.1, then we will present local MTF and the properties of its boundary integral operators in Section 4.2 using results from [31]. With these properties, we will be able to extend the results from [33] to n subdomains in Section 4.3 and we will generalize the equivalence between local MTF and OSM. Finally, we will illustrate these results numerically in Section 4.4, and explore how the Jacobi method proposed for a homogeneous material can lead to a preconditioner that can be useful for a heterogenous material. This work also appears in [39] and has been submitted.

4.1 Multi-subdomain problems

4.1.1 Geometric settings

We consider a non-overlapping partition of the whole space $\mathbb{R}^d = \bigcup_{p=0}^{n} \overline{\Omega}_p$ where $n+1$ is the number of domains, Ω_0 is the exterior domain by convention and each Ω_p is a Lipschitz open set such that $\Omega_p \cap \Omega_k = \emptyset$ for $p \neq k$. We denote $\Gamma_p := \partial\Omega_p$, $\Gamma_{p,k} := \Gamma_p \cap \Gamma_k$, and we call Σ the skeleton of the partition, defined as follows:

$$\Sigma := \bigcup_{p=0}^{n} \Gamma_p \qquad \text{with} \qquad \Gamma_p := \partial\Omega_p.$$

We also make the strong assumption that there is no junction point, that is to say

$$\Gamma_p \cap \Gamma_k \cap \Gamma_m = \emptyset, \qquad \text{for } p \neq k, p \neq m, m \neq k. \tag{4.1}$$

We define the adjacency graph $G = (V, E)$ associated with the partition:

$$V = \{\Omega_p,\, p = 0, \dots, n\},$$
$$E = \{\{\Omega_p, \Omega_k\}, \text{ such that } \Gamma_p \cap \Gamma_k \neq \emptyset \text{ and } p \neq k, \text{ for } p = 0, \dots, n \text{ and } k = 0, \dots, n\},$$

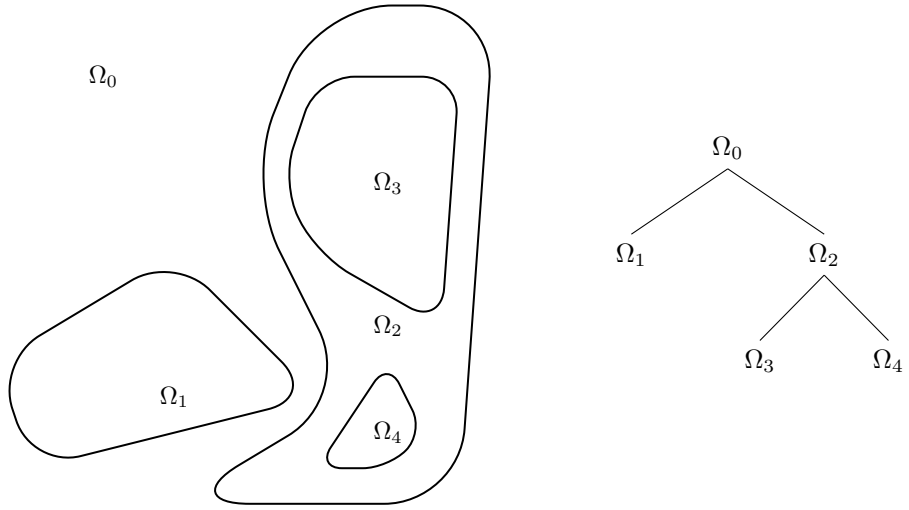


Figure 4.1 – Example of a partition of $\mathbb{R}^2$ without junction point and its associated adjacency graph.

and we denote P_G the maximal length (here, the number of edges) of a simple (without any repeated vertices) path $(\Omega_{p_1}, \ldots, \Omega_{p_{P_G+1}})$. Notice that the assumption (4.1) implies that if $\{\Omega_p, \Omega_k\} \in E$ for $0 \leq p, k \leq n$, then either $\Omega_p \subset \Omega_k$ or $\Omega_k \subset \Omega_p$. We deduce that G is a tree whose root is Ω_0, and we give an example in Figure 4.1 where $P_G = 3$.

4.1.2 Problem under consideration

We consider the Helmholtz equation in piecewise homogeneous domains. First, let us recall the form of the equation:

$$\begin{cases} -\Delta u - \kappa_p^2 u = 0 \text{ in } \Omega_p & \forall p = 0 \ldots n, \quad (4.2a) \\ u - u_{\text{inc}} \text{ is } \kappa_0\text{-outgoing}, & (4.2b) \\ \gamma_D^p(u) - \gamma_D^k(u) = 0 & \\ \mu_p^{-1}\gamma_N^p(u) + \mu_k^{-1}\gamma_N^k(u) = 0 & \text{on } \Gamma_p \cap \Gamma_k, \forall p, k = 0 \ldots n. \quad (4.2c) \end{cases}$$

which corresponds locally to the elliptic model problem given in Equation (2.1) with $\mathbf{A} = \mathbf{I}_{\text{d}}$, $\mathbf{b} = \mathbf{0}$ and $c = -\kappa^2$, where κ is the wavenumber, a function on $\mathbb{R}^d$ piecewise constant such that $\kappa(\mathbf{x}) = \kappa_p$ for $\mathbf{x} \in \Omega_p$. Similarly, $\mu : \mathbb{R}^d \to \mathbb{R}^+$ such that $\mu(\mathbf{x}) = \mu_p > 0$ for $\mathbf{x} \in \Omega_p$. We will assume that it satisfies the condition considered in [159, 37]:

$$\kappa_p \neq 0, \, \text{Re}(\kappa_p) \geq 0 \text{ and } \text{Im}(\kappa_p) \geq 0, \quad \forall p = 0, \ldots, n. \quad (4.3)$$

These two last conditions correspond respectively to a choice of a direction of propagation and a dissipative domain. In the transmission condition, γ_D^p and γ_N^p are the trace operators associated with Ω_p and μ_p is called the magnetic permeability of Ω_p in electromagnetism for example. For the exterior domain Ω_0, one has to choose a condition for the behavior of the solution at infinity, and it will depend on κ_0. We say that u is κ_0-outgoing if it satisfies

- if $\text{Im}(\kappa_0) > 0$, $u|_{\Omega_0} \in H^1(\Omega_0)$

- if $\kappa_0 > 0$, we have the Sommerfeld radiation condition whose purpose is to impose that the energy which is radiated from the sources must propagate toward infinity:

$$\left.\begin{aligned} |u(\mathbf{x})| &= \mathcal{O}(|\mathbf{x}|^{-\frac{d-1}{2}}) \\ \left|\frac{\partial u}{\partial r}(\mathbf{x}) - iku(\mathbf{x})\right| &= o(|\mathbf{x}|^{-\frac{d-1}{2}}) \end{aligned}\right\} \quad \text{for } |\mathbf{x}| \to \infty,$$

These two conditions are the same as the ones given in Remark 2.8.

The source term comes from the incident wave u_{inc} solution of the Helmholtz equation in $\mathbb{R}^d$ with κ_0. For example, a plane wave $u_{\text{inc}} = \exp(i\kappa_0 \mathbf{d} \cdot \mathbf{x})$ with $\mathbf{d} \in \mathbb{R}^3$, $|\mathbf{d}| = 1$, a direction of propagation. Notice that the incident wave is not a priori κ_0-outgoing

This equation comes from the study of periodic stationary solutions of the wave equation. That is why it is a model for stationary wave propagation. Typically, Ω_0 represents the background medium while $\bigcup_{p=1}^n \Omega_p$ is a composite scatterer. Equation (4.2) has been proved to be well-posed in [159].

4.1.3 Functional setting for multi-subdomain problems

We need to adapt the notations and the functional setting developed in Chapter 2 to the multi-subdomain problem we are considering. The notations we use are the same as in [30, Section 2.1]. We shall denote γ_D^p, γ_N^p, γ^p (resp. $\gamma_{D,c}^p$, $\gamma_{N,c}^p$, γ_c^p) interior (resp. exterior) traces on the boundary of Ω_p. We define the integral operators $\mathrm{SL}_{\kappa_p}^p$, $\mathrm{DL}_{\kappa_p}^p$, $\mathrm{G}_{\kappa_p}^p$, $\mathrm{V}_{\kappa_p}^p$, $\mathrm{W}_{\kappa_p}^p$, $\mathrm{K}_{\kappa_p}^p$, $\mathrm{K}_{\kappa_p}'^p$, $\mathrm{P}_{\kappa_p}^p$ and $\mathrm{A}_{\kappa_p}^p$ as in Sections 2.3.1 to 2.3.3 with $\Omega = \Omega_p$ and $\mathbf{n} = \mathbf{n}_p$ the normal vector directed towards the exterior of Ω_p.

We denote the local trace space $\mathcal{H}(\Gamma_p) = H^{1/2}(\Gamma_p) \times H^{-1/2}(\Gamma_p)$, the trace space associated with Γ_p, equipped with the norm $\|(v,q)\|_{\mathcal{H}(\Gamma_p)}^2 := \|v\|_{H^{1/2}(\Gamma_p)}^2 + \|q\|_{H^{-1/2}(\Gamma_p)}^2$. We also consider the following pairing

$$[\mathfrak{u}, \mathfrak{v}]_{\Gamma_p} := \int_{\Gamma_p} (uq - vp)\, \mathrm{d}\sigma$$

where $\mathfrak{u} = (u,p) \in \mathcal{H}(\Gamma_p)$ and $\mathfrak{v} = (v,q) \in \mathcal{H}(\Gamma_p)$.

Lemma 4.1. *The pairing $[\cdot,\cdot]_{\Gamma_p}$ puts $\mathcal{H}(\Gamma_p)$ in duality with itself, meaning that the application*

$$\begin{aligned} i : \mathcal{H}(\Gamma_p) &\to \mathcal{H}(\Gamma_p)^* \\ \mathfrak{v} &\mapsto \psi_{\mathfrak{v}}, \end{aligned}$$

with $\psi_{\mathfrak{v}}(\mathfrak{u}) := [\mathfrak{u}, \mathfrak{v}]$ is an isomorphism,.

Proof. Linearity is straightforward, we need to prove the injectivity and surjectivity of the considered application.

- *injectivity*: Let $\mathfrak{v} = (v,q) \in \mathcal{H}(\Gamma_p)$, let us assume that for every $\mathfrak{u} \in \mathcal{H}(\Gamma_p)$, we have $[\mathfrak{u}, \mathfrak{v}] = 0$. Taking $\mathfrak{u} = (u,0)$ and then, $\mathfrak{u} = (0,p)$, we obtain that

$$\int_{\Gamma_p} vp\, \mathrm{d}\sigma = 0,\ \forall p \in H^{-1/2}(\Gamma_p)$$
$$\int_{\Gamma_p} qu\, \mathrm{d}\sigma = 0,\ \forall u \in H^{1/2}(\Gamma_p)$$

so that $\mathfrak{v} = 0$ and $\ker(i) = \{0\}$.

- *surjectivity*: Let $\tilde{\mathfrak{v}} = (q, \tilde{v}) \in \mathcal{H}(\Gamma_p)^* = H^{1/2}(\Gamma_p) \times H^{-1/2}(\Gamma_p)$, we have

$$\langle \mathfrak{u}, \tilde{\mathfrak{v}} \rangle_{\mathcal{H}(\Gamma_p) \times \mathcal{H}(\Gamma_p)^*} = \int_{\Gamma_p} (uq + \tilde{v}p) \, \mathrm{d}\sigma = [\mathfrak{u}, \mathfrak{v}],$$

defining $\mathfrak{v} = (-\tilde{v}, q) \in \mathcal{H}(\Gamma_p)$, so that $i(\mathfrak{v}) = \tilde{\mathfrak{v}}$.

□

Definition 4.2 (Multi-trace space)**.** *Let us denote the multi-trace space* $\mathbb{H}(\Sigma)$ *as the cartesian product of traces on the boundary of each domain:*

$$\mathbb{H}(\Sigma) := \mathcal{H}(\Gamma_0) \times \cdots \times \mathcal{H}(\Gamma_n)$$

This space will be equipped with its natural cartesian product norm: for every $\mathfrak{u} = (\mathfrak{u}_p)_{p=0}^n \in \mathbb{H}(\Sigma)$

$$\|\mathfrak{u}\|^2_{\mathbb{H}(\Sigma)} := \sum_{p=0}^n \|\mathfrak{u}_p\|^2_{\mathcal{H}(\Gamma_p)}.$$

Then, we define as well a duality pairing:

$$[\![\mathfrak{u}, \mathfrak{v}]\!] := \sum_{p=0}^n [\mathfrak{u}_p, \mathfrak{v}_p]_{\Gamma_p}$$

for every $\mathfrak{u} = (\mathfrak{u}_p)_{p=0}^n$ and $\mathfrak{v} = (\mathfrak{v}_p)_{p=0}^n$ belonging to $\mathbb{H}(\Sigma)$, and this pairing puts also $\mathbb{H}(\Sigma)$ in duality with itself.

Definition 4.3 (Single-trace space)**.** *Let us denote the single-trace space* $\mathbb{X}(\Sigma)$ *defined as*

$$\begin{aligned}\mathbb{X}(\Sigma) := \big\{ \mathfrak{u} = &(v_p, p_p)_{p=0}^n \in \mathbb{H}(\Sigma) \\ &\exists v \in H^1(\mathbb{R}^d) \text{ such that } v|_{\Gamma_p} = v_p \\ &\exists \mathbf{q} \in H(\mathrm{div}, \mathbb{R}^d) \text{ such that } \mathbf{n}_p \cdot \mathbf{q}|_{\Gamma_p} = q_p \quad \forall p = 0, \dots, n \big\},\end{aligned}$$

where $H(\mathrm{div}, \mathbb{R}^d) := \{\mathbf{q} \in L^2(\mathbb{R}^d)^d \mid \mathrm{div}(\mathbf{q}) \in L^2(\mathbb{R}^d)\}$.

It corresponds to a subspace of $\mathbb{H}(\Sigma)$ whose functions satisfy the transmission conditions. It can be seen as a space taking into account the transmission condition as an essential condition when $\mu_1 = \cdots = \mu_n$. This denomination "single trace" comes from the fact that only one pair of Dirichlet and Neumann traces come into play on each Γ_p contrary to the multi-trace space. The following lemma gives a characterization of the single trace space:

Lemma 4.4 ([29, Proposition 2.1])**.** *For any* $\mathfrak{u} \in \mathbb{H}(\Sigma)$ *we have* $\mathfrak{u} \in \mathbb{X}(\Sigma) \iff [\![\mathfrak{u}, \mathfrak{v}]\!] = 0 \quad \forall \mathfrak{v} \in \mathbb{X}(\Sigma)$.

We generalize the definition of the solution space given in Theorem 2.24 in a multi-subdomain setting:

$$\begin{aligned}C_{(\kappa)}(\Sigma) := \big\{ \mathfrak{u} = (\gamma^p(u_p))_{p=0} | u_p \in H^1_{\mathrm{loc}}(\overline{\Omega}_p) \text{ and } &-\Delta u_p - \kappa_p^2 u_p = 0 \text{ in } \Omega_p \\ &\text{and } u_p \text{ is } \kappa_p\text{-outgoing if } \Omega_p \text{ is unbounded } \big\}\end{aligned}$$

where (κ) stands for the tuple $(\kappa_0, \ldots, \kappa_n)$. Then, this space has a characterization similar to Lemma 4.4 for the single-trace space:

Lemma 4.5 ([37, Lemma 6.2]). *For any* $\mathfrak{u} \in \mathbb{H}(\Sigma)$ *we have* $\mathfrak{u} \in C_{(\kappa)}(\Sigma) \iff [\![\mathfrak{u}, \mathfrak{v}]\!] = 0 \quad \forall \mathfrak{v} \in C_{(\kappa)}(\Sigma)$.

The multi-trace space is actually a direct sum of the single-trace space and the cauchy data set:

Lemma 4.6 ([37, Proposition 6.1]). $\mathbb{H}(\Sigma) = \mathbb{X}(\Sigma) \bigoplus C_{(\kappa)}(\Sigma)$.

4.2 Formulations with boundary integral operators

4.2.1 Integral operators in a multi-subdomain setting

Using the operators introduced in Chapter 2, we define integral operators acting on the multi-trace space. Let us define the Calderón operator $\mathbb{P}_{(\kappa)} : \mathbb{H}(\Sigma) \to \mathbb{H}(\Sigma)$ and $\mathbb{A}_{(\kappa)} : \mathbb{H}(\Sigma) \to \mathbb{H}(\Sigma)$ as:

$$[\![\mathbb{P}_{(\kappa)}\left(\mathfrak{v}\right), \mathfrak{v}']\!] := \sum_{p=0}^{n} \left[\mathrm{P}^p_{\kappa_p}\, \mathfrak{v}_p, \mathfrak{v}'_p\right]_{\Gamma_p},$$

$$[\![\mathbb{A}_{(\kappa)}(\mathfrak{v}), \mathfrak{v}']\!] := \sum_{p=0}^{n} \left[\mathrm{A}^p_{\kappa_p}\, \mathfrak{v}_p, \mathfrak{v}'_p\right]_{\Gamma_p},$$

for all $\mathfrak{v} = (\mathfrak{v}_p)_{p=0}^n$, $\mathfrak{v}' = (\mathfrak{v}'_p)_{p=0}^n \in \mathbb{H}(\Sigma)$ where $\mathrm{P}^p_{\kappa_p}$ is the local Calderón projector associated with Ω_p and $\mathrm{A}^p_{\kappa_p} = 2\,\mathrm{P}^p_{\kappa_p} - \mathrm{I_d}$ as defined in Section 2.3.3. We may also use the equivalent notation in matrix-type form as follows:

$$\mathbb{P}_{(\kappa)} = \operatorname*{diag}_{p=0,\ldots,n}\left(\mathrm{P}^p_{\kappa_p}\right),$$

$$\mathbb{A}_{(\kappa)} = \operatorname*{diag}_{p=0,\ldots,n}\left(\mathrm{A}^p_{\kappa_p}\right),$$

acting on vectors such as $\mathfrak{v} = (\mathfrak{v}_0, \ldots, \mathfrak{v}_n)^T$. Since we are considering a scattering problem with several domains, we need extension and restriction operators to decompose traces accordingly. Let us denote $\Gamma_{p,k} = \Gamma_p \cap \Gamma_k$ so that $\Gamma_p = \bigcup_{k \neq p} \Gamma_{p,k}$. For a given pair (p,k) with $p \neq k$, we define:

$$\begin{aligned} \mathrm{R}^p_k : \mathcal{H}(\Gamma_p) &\to \mathcal{H}(\Gamma_{p,k}) \\ \mathfrak{v}^p &\mapsto \mathfrak{v}^p|_{\Gamma_{p,k}}, \end{aligned}$$

and we will denote $\mathfrak{v}^p_k = \mathrm{R}^p_k(\mathfrak{v}^p)$. The adjoint of this restriction operator is given by:

$$\begin{aligned} (\mathrm{R}^p_k)^* : \mathcal{H}(\Gamma_{p,k}) &\to \mathcal{H}(\Gamma_p) \\ \mathfrak{v} &\mapsto \begin{cases} \mathfrak{v} \text{ on } \Gamma_{p,k} \\ 0 \text{ on } \Gamma_p \setminus \Gamma_{p,k}. \end{cases} \end{aligned}$$

Now that we can decompose traces using restriction and extension operators, each $\mathrm{P}^p_{\kappa_p}$ (resp. $\mathrm{A}^p_{\kappa_p}$) induces a matrix of integral operators denoted $[\mathrm{P}^p_{\kappa_p}]$ (resp. $[\mathrm{A}^p_{\kappa_p}]$) whose maximal size is

$(n \times n)$. For example, we have

$$\left[\mathrm{P}^p_{\kappa_p}\right] := \begin{bmatrix} \mathrm{P}^p_{\kappa_p,0,0} & \cdots & \mathrm{P}^p_{\kappa_p,0,n} \\ \vdots & & \vdots \\ \mathrm{P}^p_{\kappa_p,n,0} & \cdots & \mathrm{P}^p_{\kappa_p,n,n} \end{bmatrix} \quad \text{and} \quad \mathrm{P}^p_{\kappa_p,k,m} := \mathrm{R}^p_k \circ \mathrm{P}^p_{\kappa_p} \circ (\mathrm{R}^p_m)^*. \tag{4.4}$$

But it should be noted that the rows and columns associated with indices k such that $\Gamma_{p,k} = \partial\Omega_p \cap \partial\Omega_k = \emptyset$ and $k = p$ must be omitted. So that, if we define for each domain Ω_p, the set of indices of its neighbors:

$$\Lambda_p := \{k \,|k \neq p,\, \partial\Omega_p \cap \partial\Omega_k \neq \emptyset\},$$

the matrix in Equation (4.4) is square with $\mathrm{card}(\Lambda_p)$ rows. We also define the following matrix Q:

$$\mathrm{Q} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

It will be used to apply transmission conditions. Indeed, because of the orientation of the normal in the definition of the trace operator γ, we see that for a continuous function u across $\Gamma_{p,k} = \Gamma_p \cap \Gamma_k$, we have:

$$\gamma^p(u) = \mathrm{Q}\,\gamma^k(u).$$

This matrix is the elementary building block of the transmission operator Π that is the key ingredient of the multi-trace formulation introduced in Section 4.2.2.

4.2.2 Several formulation

There are several ways to formulate the scattering problem defined in Equation (4.2) by means of boundary integral equations, and the properties of each of these formulations can be quite different.

4.2.2.1 Rumsey's reaction principle (PMCHWT)

One of the most popular approach to solve Equation (4.2) is called Rumsey's reaction principle, or PMCHWT which stands for Poggio-Miller-Chang-Harrington-Wu-Tsai from the various authors who introduced independently this formulation (see [131, 23, 163]). The mathematical analysis has been carried in [159] (see also [38, Section 3.2]). This formulation can be expressed with the previous notation in the following way: let us denote the scattering source:

$$\mathfrak{u}_{\mathrm{inc}} := \big(\gamma^0(u_{\mathrm{inc}}), 0, \ldots, 0\big)$$

where u_{inc} satisfies $-\Delta u_{\mathrm{inc}} + \kappa_0^2 u_{\mathrm{inc}} = 0$, in $\mathbb{R}^d$. As in Equation (4.2) we call the solution u, and we write $\tilde{\mathfrak{u}} = (\tau_{\mu_p} \circ \gamma^p(u))_{p=0}^n$ where $\tau_\alpha(v, q) := (v, q/\alpha)$ for $\alpha > 0$. Then Equation (4.2c) can be rewritten as an essential condition $\tilde{\mathfrak{u}} \in \mathbb{X}(\Sigma)$. We denote $\tau_{(\mu)}(\mathfrak{v}) := (\tau_{\mu_p}(\mathfrak{v}_p))_{p=0}^n$ for $\mu = (\mu_0, \ldots, \mu_n)$ and $\mathfrak{v} \in \mathbb{H}(\Sigma)$, in particular, we have $\tau_{(\mu)}^{-1}(\tilde{\mathfrak{u}}) = \mathfrak{u} = (\gamma^p(u))_{p=0}^n$. According to Equations (4.2a) and (4.2b) we can apply Theorem 2.24 to $u - \mathrm{E}_{\Omega_0}(u_{\mathrm{inc}})$ such that:

$$\mathbb{P}_{(\kappa)}(\tau_{(\mu)}^{-1}(\tilde{\mathfrak{u}}) - \mathfrak{u}_{\mathrm{inc}}) = \tau_{(\mu)}^{-1}(\tilde{\mathfrak{u}}) - \mathfrak{u}_{\mathrm{inc}}.$$

This last equation can be rewritten as follows using the relation between $\mathrm{A}^p_{\kappa_p}$ and $\mathrm{P}^p_{\kappa_p}$:

$$\left(\mathbb{A}_{(\kappa)} - \mathrm{I_d}\right)\left(\tau^{-1}_{(\mu)}(\tilde{\mathfrak{u}})\right) = \left(\left(\mathrm{A}^0_{\kappa_0} - \mathrm{I_d}\right)\gamma^0(u_{\mathrm{inc}}), 0, \ldots, 0\right). \tag{4.5}$$

To simplify the right-hand side, we can use the fact that u_{inc} is solution of the Helmholtz equation with a uniform coefficient κ_0 in the whole space, so that according to Corollary 2.20:

$$\gamma^0_c \circ G^0_{\kappa_0} \circ \gamma^0_c(u_{\mathrm{inc}}) = -\gamma^0_c(u_{\mathrm{inc}}). \tag{4.6}$$

Applying the identity $\gamma^0_c = \{\gamma^0\} - [\gamma^0]/2$ to the equation Equation (4.6) and using the jump formula from Theorem 2.22, we obtain:

$$\left(\frac{\mathrm{A}^0_{\kappa_0} - \mathrm{I_d}}{2}\right)\gamma^0_c(u_{\mathrm{inc}}) = -\gamma^0_c(u_{\mathrm{inc}}).$$

And since $\gamma^0(u_{\mathrm{inc}}) = \gamma^0_c(u_{\mathrm{inc}})$ by elliptic regularity, we have the following identity:

$$\mathrm{A}^0_{\kappa_0}\,\gamma^0(u_{\mathrm{inc}}) = -\gamma^0(u_{\mathrm{inc}}). \tag{4.7}$$

Then, Equation (4.7) allows us to simplify the right-hand side in Equation (4.5) and, testing this equation with a test function $\mathfrak{v} \in \mathbb{X}(\Sigma)$ and using Lemma 4.4, we obtain the following variational formulation: find $\tilde{\mathfrak{u}} \in \mathbb{X}(\Sigma)$ such that

$$\left[\!\left[\mathbb{A}_{(\kappa,\mu)}(\tilde{\mathfrak{u}}), \mathfrak{v}\right]\!\right] = -2\left[\!\left[\tau_{(\mu)}(\mathfrak{u}_{inc}), \mathfrak{v}\right]\!\right], \quad \forall \mathfrak{v} \in \mathbb{H}(\Sigma), \tag{4.8}$$

where $\mathbb{A}_{(\kappa,\mu)} := \tau_{(\mu)} \circ \mathbb{A}_{(\kappa)} \circ \tau^{-1}_{(\mu)}$. As we did in Chapter 2 using Theorem 2.14, the key to prove well-posedness is to show that the operator $\mathbb{A}_{(\kappa,\mu)}$ is injective and satisfies a Gårding inequality. To do so, we have the following theorem for the Helmholtz equation:

Theorem 4.7 ([95, Theorem 2.16])**.** *Set $\theta(v,q) = (-v,q)$ and $\Theta(\mathfrak{v}) = (\theta(\mathfrak{v}_p))_{p=0}^n$. There exists a constant $C > 0$ and a compact operator $\mathbb{K}_{(\kappa)} : \mathbb{H}(\Sigma) \to \mathbb{H}(\Sigma)$ such that*

$$\mathrm{Re}\left(\left[\!\left[\left(\mathbb{A}_{(\kappa,\mu)} + \mathbb{K}_{(\kappa)}\right)(\mathfrak{v}), \Theta(\overline{\mathfrak{v}})\right]\!\right]\right) \geq C\|\mathfrak{v}\|^2_{\mathbb{H}(\Sigma)} \quad \forall \mathfrak{v} \in \mathbb{H}(\Sigma).$$

Hence, we deduce that $\mathbb{A}_{(\kappa,\mu)}$ is an isomorphism if and only if it is injective, which is true under our assumption given by Equation (4.3):

Theorem 4.8 ([38, Lemma 3.4])**.** *Under the assumption given by Equation (4.3), $\mathbb{A}_{(\kappa,\mu)}$ is injective, so that Equation (4.8) is well-posed.*

As explained in Section 2.4.5, we need to have a preconditioning strategy because the matrices stemming from boundary element methods are usually ill-conditioned. According to [38, Section 4], this is also the case for the Galerkin approximation of Equation (4.8). It is also explained in [38] that it is not possible to use the operator preconditioning strategy described in [146, 27, 28, 94] for the PMCHWT approach with a geometry involving junction points, and this is what led the development of the alternative formulation called *Multri-trace* formulations. But here, we will be interested in these methods from a domain decomposition point of view.

4.2.2.2 Multi-trace formulation (MTF)

Dealing with multi-subdomain scattering problems, several alternatives to the PMCHWT approach exist. For example, there is the *Boundary Element Tearing and Interconnecting* (BETI)

method introduced in [110] which is the boundary element counterpart of the *Finite Element Tearing and Interconnecting* (FETI) method. Here, we will focus on another alternative called local multi-trace formulation introduced in [96] and [100]. It should be noted that there exists also a global multi-trace formulation (see [35]) and a quasi-local multi-trace formulation ([32]).

All these formulations are dubbed *Multi-Trace Formulation* (MTF) because at each point of each interface, two pairs of Dirichlet/Neumann traces come into play, contrary to the PMCHWT approach. They differ on the way transmission conditions are imposed. Concerning local MTF, the transmission condition is weakly imposed with a transmission operator Π defined as follows: $\forall \mathfrak{u} = (\mathfrak{u}^p)_{p=0}^n \in \mathbb{H}(\Sigma), \mathfrak{v} = (\mathfrak{v}^p)_{p=0}^n \in \mathbb{H}(\Sigma)$ such that $\mathfrak{u}^k = (u^k, p^k)$, $\mathfrak{v}^k = (v^k, q^k) \in \mathcal{H}(\Gamma_p)$ for $1 \leq k \leq n$, we have

$$\begin{aligned}\Pi(\mathfrak{u}) = \mathfrak{v} &\iff \mathfrak{v}_k^p = \mathrm{Q}\,\mathfrak{u}_p^k \text{ on } \Gamma_{p,k}, \quad \forall k \in \Delta_p \text{ and } \forall\, 0 \leq p \leq n, \\ &\iff \begin{cases} v^p = & u^k \\ q^p = & -p^k \end{cases} \text{ on } \Gamma_{p,k} \quad \forall k \in \Delta_p \text{ and } \forall\, 0 \leq p \leq n.\end{aligned}$$

This operator maps continuously $\mathbb{H}(\Sigma)$ onto $\mathbb{H}(\Sigma)$ under the assumption that there is no junction point in the geometry. It has the interesting property to only exchange traces between neighboring subdomains, similarly to Schwarz methods. Then, we have the following properties:

$$\Pi^2 = \mathrm{I_d}, \quad \text{and} \quad [\![\Pi(\mathfrak{u}), \mathfrak{v}]\!] = [\![\Pi(\mathfrak{v}), \mathfrak{u}]\!] \quad \forall \mathfrak{u}, \mathfrak{v} \in \mathbb{L}^2(\Sigma)$$

where $\mathbb{L}^2(\Sigma)$ is defined as the cartesian product of $L^2(\Gamma_0) \times \cdots \times L^2(\Gamma_n)$ for every $0 \leq p \leq n$. For every $\mathfrak{u} \in \mathbb{H}(\Sigma)$, we have $\mathfrak{u} \in \mathbb{X}(\Sigma) \iff \Pi(\mathfrak{u}) = \mathfrak{u}$. Another way to express the transmission operator is to use the matrix operator Q: for every $\mathfrak{u}, \mathfrak{v} \in \mathbb{H}(\Sigma)$

$$[\![\Pi(\mathfrak{u}), \mathfrak{v}]\!] = \sum_{p=0}^{n} \sum_{k \in \Lambda_p} \left[\mathrm{Q}(\mathfrak{u}_p^k), \mathfrak{v}_k^p\right]_{\Gamma_{p,k}}.$$

Using Equation (4.5) and simplifying the right-hand side as we did in the last section, the transmission condition is imposed weakly so that we have the following variational formulation of Equation (4.2): find $\tilde{\mathfrak{u}} \in \mathbb{H}(\Sigma)$ such that

$$[\![\left(\mathbb{P}_{(\kappa,\mu)} - \mathrm{I_d}\right)(\tilde{\mathfrak{u}}), \mathfrak{v}]\!] + \alpha\, [\![(\mathrm{I_d} - \Pi)(\tilde{\mathfrak{u}}), \mathfrak{v}]\!] = -[\![\tau_{(\mu)}(\mathfrak{u}_{inc}), \mathfrak{v}]\!], \quad \forall \mathfrak{v} \in \mathbb{H}(\Sigma). \tag{4.9}$$

This formulation was originally proposed in [96] with $\alpha = \frac{1}{2}$, and then, a variant with the relaxation parameter α was introduced in [100]. Well-posedness in the first case given in [96] and extended to the second case in [31] for $\alpha \in \mathbb{C} \setminus \{0\}$. The value $\alpha = 0$ is clearly forbidden since $\mathrm{I_d} - \mathbb{P}_{(\kappa,\mu)}$ is a non-trivial projector, and hence it cannot be invertible. Let us denote the local-MTF operator $\mathrm{MTF}_{\alpha,(\kappa,\mu)} := \mathbb{P}_{(\kappa,\mu)} + (\alpha - 1)\,\mathrm{I_d} - \alpha\Pi$.

4.2.3 Algebraic relations

In this section, we will derive some algebraic relations concerning the operators introduced in Section 4.2.1, similarly to what is done in [31, Section 6]. However, we focus on the operator $\mathbb{P}_{(\kappa,\mu)}$ instead of $\mathbb{A}_{(\kappa,\mu)}$ since the former has the good property of being a projector.

First, we show some relations for the operators defined on trace spaces $\mathcal{H}(\Gamma_p)$ for $0 \leq p \leq n$, and then these relations will help us to derive relations for operators defined on the multi-trace space $\mathbb{H}(\Sigma)$.

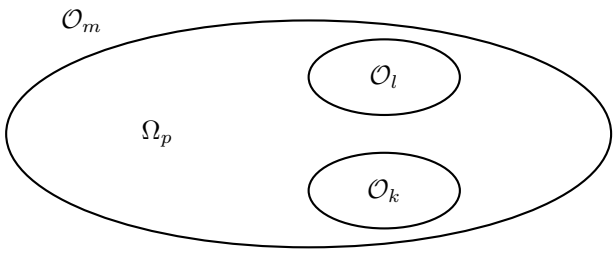


Figure 4.2 – Example of a multi-subdomain setting for Lemma 4.9.

Lemma 4.9 ([31, Lemma 6.1])**.** *For $p = 0, \dots, n$, and every $k, l, m \in \Lambda_p$ with $k \neq m$ and $k \neq l$, we have*

(i) $\mathrm{P}^p_{lk} \circ \mathrm{P}^p_{km} = 0$,

(ii) $(\mathrm{P}^p_{kk})^2 = \mathrm{P}^p_{kk}$,

(iii) $\mathrm{P}^p_{lk} \circ \mathrm{P}^p_{kk} = \mathrm{P}^p_{lk}$

(iv) $\mathrm{Q}\,\mathrm{P}^p_{kk} = -\mathrm{P}^k_{pp}\,\mathrm{Q} + \mathrm{Q}$ *for* $\kappa_p = \kappa_l$.

Proof. Let $p = 0, \dots, n$ and $k, l, m \in \Delta_p$ with $k \neq l$ and $k \neq m$. Let us denote $\mathcal{O}_l$ the maximal open set such that $\partial\mathcal{O}_l = \Gamma_{pl}$ and $\mathcal{O}_l \cap \Omega_p = \emptyset$ (see Section 4.2.3). Take an arbitrary $\mathfrak{v} \in \mathcal{H}(\Gamma_{p,m})$ and $\widetilde{\mathfrak{v}} = (\mathrm{R}^p_m)^* \mathfrak{v}$, it should be noticed that $\mathrm{G}^p(\widetilde{\mathfrak{v}})$ does not have a jump on $\Gamma_{p,k}$ according to the jump relations from Theorem 2.22, because $\Gamma_{p,k} \cap \Gamma_{p,m} = \emptyset$. We deduce that

$$\mathrm{P}^p_{km}(\mathfrak{v}) = \mathrm{R}^p_k \circ \gamma^p \circ \mathrm{G}^p(\widetilde{\mathfrak{v}}) = \mathrm{R}^p_k \circ \gamma^p_c \circ \mathrm{G}^p(\widetilde{\mathfrak{v}}).$$

Let us denote w the function defined as follows:

$$w = \begin{cases} \mathrm{G}^p(\widetilde{\mathfrak{v}}) \text{ in } \mathcal{O}_k \\ \quad 0 \text{ in } \mathcal{O}_i \text{ for } i \neq k \text{ and } i \in \Lambda_p. \end{cases}$$

Notice that w is solution to the Helmholtz equation in the complementary of Ω_p. Besides, we have the relation

$$\gamma^p_c(w) = (\mathrm{R}^p_k)^* \circ \mathrm{R}^p_k \circ \gamma^p_c \circ \mathrm{G}^p(\widetilde{\mathfrak{v}}) = (\mathrm{R}^p_k)^* \circ \mathrm{P}^p_{km}(\mathfrak{v})$$

According to the second part of Corollary 2.20, we have

$$\mathrm{G}^p \circ \gamma^p_c(w) = \begin{cases} -\mathrm{G}^p(\widetilde{\mathfrak{v}}) \text{ in } \mathcal{O}_k \\ \quad 0 \text{ in } \mathcal{O}_i \text{ for } i \neq k \text{ and } i \in \Lambda_p \\ \quad 0 \text{ in } \Omega_p. \end{cases}$$

We conclude that the interior trace γ^p of $\mathrm{G}^p \circ \gamma^p_c(w)$ vanishes, that is to say $\gamma^p \circ \mathrm{G}^p \circ (\mathrm{R}^p_k)^* \circ \mathrm{P}^p_{km} = 0$, and therefore we have the first relation, $\mathrm{P}^p_{lk} \circ \mathrm{P}^p_{km} = 0$[1].

Now we want to prove second and third relation. We know that $(\mathrm{P}^p)^2 = \mathrm{P}^p$ because it is the

[1] We actually do not need the assumption $l \neq k$ made in [31, Lemma 6.1]

Calderón projector associated with Ω_p. We deduce

$$\sum_{m\in\Delta_p} \mathrm{P}^p_{km}\circ\mathrm{P}^p_{mk} = \mathrm{P}^p_{kk} \text{ and } \sum_{m\in\Delta_p} \mathrm{P}^p_{lm}\circ\mathrm{P}^p_{mk} = \mathrm{P}^p_{lk}\,.$$

But according to the first relation, every term such that $k \neq m$ is equal to zero. Thus, we obtain $(\mathrm{P}^p_{kk})^2 = \mathrm{P}^p_{kk}$ and $\mathrm{P}^p_{lk}\circ\mathrm{P}^p_{kk} = \mathrm{P}^p_{lk}$. Finally, let us prove the fourth relation. Using the definition of the potentials, we have

$$\mathrm{G}^p\circ(\mathrm{R}^p_k)^* = -\mathrm{G}^k\circ(\mathrm{R}^k_p)^*\,\mathrm{Q}\,. \tag{4.10}$$

TThen, we remark that we have $\mathrm{R}^p_k\,\gamma^p = \mathrm{Q}\,\mathrm{R}^k_p\,\gamma^k_c$, so that

$$\begin{aligned}\mathrm{P}^p_{kk} &= \mathrm{R}^p_k\,\gamma^p\mathrm{G}^p(\mathrm{R}^p_k)^* = \mathrm{Q}\,\mathrm{R}^k_p\,\gamma^k_c\mathrm{G}^p(\mathrm{R}^p_k)^*\\ &= \mathrm{Q}\,\mathrm{R}^k_p(\gamma^k - [\gamma^k])\mathrm{G}^p(\mathrm{R}^p_k)^*\\ &= \mathrm{Q}\,\mathrm{R}^k_p\,\gamma^k\mathrm{G}^k\circ(\mathrm{R}^k_p)^*\,\mathrm{Q} - \mathrm{Q}\,\mathrm{R}^k_p[\gamma^k]\mathrm{G}^k\circ(\mathrm{R}^k_p)^*\,\mathrm{Q},\end{aligned}$$

where we used Equation (2.5) in the second line and Equation (4.10) in the third one. The second term in the right-hand side is equal to the identity using Theorem 2.22 and the first term is equal to $-\,\mathrm{Q}\,\mathrm{P}^k_{pp}\,\mathrm{Q}$ by definition, then we obtain the fourth relation by multiplying by Q. □

Let us define the operators $\mathbb{D}_{(\kappa)}$ and $\mathbb{T}_{(\kappa)}$:

$$\begin{aligned}\big[\!\big[\mathbb{D}_{(\kappa)}(\mathfrak{v}),\mathfrak{v}'\big]\!\big] &:= \sum_{p=0}^{n}\sum_{k\in\Lambda_p}\big[\mathrm{P}^p_{kk}(\mathfrak{v}^p_k),\mathfrak{v}'^p_k\big]_{\Gamma_{p,k}} \qquad \forall\mathfrak{v},\mathfrak{v}'\in\mathbb{H}(\Sigma),\\ \big[\!\big[\mathbb{T}_{(\kappa)}(\mathfrak{v}),\mathfrak{v}'\big]\!\big] &:= \sum_{p=0}^{n}\sum_{\substack{k,m\in\Lambda_p\\ k\neq m}}\big[\mathrm{P}^p_{km}(\mathfrak{v}^p_m),\mathfrak{v}'^p_k\big]_{\Gamma_{p,k}} \qquad \forall\mathfrak{v},\mathfrak{v}'\in\mathbb{H}(\Sigma),\end{aligned}$$

such that $\mathbb{P}_{(\kappa)} = \mathbb{D}_{(\kappa)} + \mathbb{T}_{(\kappa)}$. The first operator can be seen in a matrix-type notation as the diagonal of $\mathbb{P}$ and the second one as the extra-diagonal part. Notice that for $\mathfrak{v}\in\mathbb{H}(\Sigma)$, the components of $\mathbb{D}_{(\kappa)}$ and $\mathbb{T}_{(\kappa)}$ associated with $\Gamma_{p,k}$ are respectively $\mathrm{P}^p_{kk}(\mathfrak{v}^p_k)$ and $\sum_{m\in\Delta_p,m\neq k}\mathrm{P}^p_{km}(\mathfrak{v}^p_m)$. We have a similar result as [36, Lemma 6.2] which gives some properties of the operator $\mathbb{D}_{(\kappa)}$.

Lemma 4.10 ([31, Lemma 6.2])**.** *We have* $\mathbb{D}^2_{(\kappa)} = \mathbb{D}_{(\kappa)}$ *and* $\Pi\circ\mathbb{D}_{(\kappa)} + \mathbb{D}_{(\kappa)}\circ\Pi = \Pi$

Proof. These results come from direct computations and Lemma 4.9: $\forall\mathfrak{v},\mathfrak{v}'\in\mathbb{H}(\Sigma)$,

$$\begin{aligned}\Big[\!\Big[\mathbb{D}^2_{(\kappa)}(\mathfrak{v}),\mathfrak{v}'\Big]\!\Big] &= \sum_{p=0}^{n}\sum_{k\in\Lambda_p}\Big[\underbrace{(\mathrm{P}^p_{kk})^2}_{=\mathrm{P}^p_{kk}}(\mathfrak{v}^p_k),\mathfrak{v}'^p_k\Big]_{\Gamma_{p,k}} = \big[\!\big[\mathbb{D}_{(\kappa)}(\mathfrak{v}),\mathfrak{v}'\big]\!\big]\,,\\ \big[\!\big[\Pi\circ\mathbb{D}_{(\kappa)}(\mathfrak{v}),\mathfrak{v}'\big]\!\big] &= \sum_{p=0}^{n}\sum_{k\in\Lambda_p}\Big[\mathrm{Q}\,\mathrm{P}^k_{pp}(\mathfrak{v}^k_p),\mathfrak{v}'^p_k\Big]_{\Gamma_{p,k}}\\ &= \sum_{p=0}^{n}\sum_{k\in\Lambda_p} -\big[\mathrm{P}^p_{kk}\circ\mathrm{Q}(\mathfrak{v}^k_p),\mathfrak{v}'^p_k\big]_{\Gamma_{p,k}} + \big[\mathrm{Q}(\mathfrak{v}^k_p),\mathfrak{v}'^p_k\big]_{\Gamma_{p,k}}\\ &= -\big[\!\big[\mathbb{D}_{(\kappa)}\circ\Pi(\mathfrak{v}),\mathfrak{v}'\big]\!\big] + \big[\!\big[\Pi(\mathfrak{v}),\mathfrak{v}'\big]\!\big]\,.\end{aligned}$$

□

Concerning the operator $\mathbb{T}_{(\kappa)}$, similarly to [31, Lemma 6.3] and [31, Corollary 6.1], we have the following lemma:

Lemma 4.11 ([31, Lemmas 6.3 and Corollary 6.1])**.** *We have the following relations*

$$\mathbb{T}_{(\kappa)}^2 = 0 \quad \textit{and} \quad \left(\Pi \circ \mathbb{T}_{(\kappa)} + \mathbb{T}_{(\kappa)} \circ \Pi\right)^k = \left(\Pi \circ \mathbb{T}_{(\kappa)}\right)^k + \left(\mathbb{T}_{(\kappa)} \circ \Pi\right)^k.$$

Proof. As in the last lemma, the proof is just a direct computation using Lemma 4.9: $\forall \mathfrak{v}, \mathfrak{v}' \in \mathbb{H}(\Sigma)$,

$$\left[\!\left[\mathbb{T}_{(\kappa)}^2(\mathfrak{v}), \mathfrak{v}'\right]\!\right] := \sum_{p=0}^{n} \sum_{\substack{k,l,m \in \Lambda_p \\ k \neq l, l \neq m}} \Big[\underbrace{\mathrm{P}_{kl}^p \circ \mathrm{P}_{lm}^p}_{=0}(\mathfrak{v}_m^p), \mathfrak{v}_k'^p\Big]_{\Gamma_{p,k}} = 0.$$

The second relation is a direct consequence of the first one and the identity $\Pi^2 = \mathrm{I_d}$. □

We are now interested in relations between the two operators $\mathbb{D}_{(\kappa)}$ and $\mathbb{T}_{(\kappa)}$.

Lemma 4.12. *We have* $\mathbb{D}_{(\kappa)} \circ \mathbb{T}_{(\kappa)} = 0$ *and* $\mathbb{T}_{(\kappa)} \circ \mathbb{D}_{(\kappa)} = \mathbb{T}_{(\kappa)}$.

Proof. Again, we use results from Lemma 4.9: $\forall \mathfrak{v}, \mathfrak{v}' \in \mathbb{H}(\Sigma)$,

$$\left[\!\left[\mathbb{D}_{(\kappa)} \circ \mathbb{T}_{(\kappa)}(\mathfrak{v}), \mathfrak{v}'\right]\!\right] = \sum_{p=0}^{n} \sum_{\substack{k,m \in \Lambda_p \\ k \neq m}} \Big[\underbrace{\mathrm{P}_{kk}^p \circ \mathrm{P}_{km}^p}_{=0}(\mathfrak{v}_m^p), \mathfrak{v}_k'^p\Big]_{\Gamma_{p,k}} = 0$$

$$\left[\!\left[\mathbb{T}_{(\kappa)} \circ \mathbb{D}_{(\kappa)}(\mathfrak{v}), \mathfrak{v}'\right]\!\right] = \sum_{p=0}^{n} \sum_{\substack{k,m \in \Lambda_p \\ k \neq m}} \Big[\underbrace{\mathrm{P}_{km}^p \circ \mathrm{P}_{mm}^p}_{=\mathrm{P}_{km}^p}(\mathfrak{v}_m^p), \mathfrak{v}_k'^p\Big]_{\Gamma_{p,k}} = \left[\!\left[\mathbb{T}_{(\kappa)}(\mathfrak{v}), \mathfrak{v}'\right]\!\right]$$

□

4.2.4 Spectrum of the local-MTF

Similarly to [31, Theorem 6.1], we know precisely the spectrum of the local-MTF operator in the case of a homogeneous material, meaning $\kappa_0 = \cdots = \kappa_n$ and $\mu_0 = \cdots = \mu_n$. We drop the subscripts (κ, μ) and (κ) in this case, i.e. $\mathbb{T}_{(\kappa)}$ will be denoted $\mathbb{T}$. First, we prove that $\Pi\mathbb{T}$ and $\mathbb{T}\Pi$ are nilpotent operators of index P_G similarly to [31, Proposition 6.1], but with more precise results on the index.

Lemma 4.13. *If* $\kappa_0 = \cdots = \kappa_n$*, then* $(\Pi\mathbb{T})^{P_G} = (\mathbb{T}\Pi)^{P_G} = (\Pi\mathbb{T} + \mathbb{T}\Pi)^{P_G} = 0$ *where* P_G *is defined in Section 4.1.1 as the maximal length of a simple path in the adjacency graph associated with the partition.*

Proof. A direct computation gives

$$\left[\!\left[(\Pi \circ \mathbb{T})(\mathfrak{u}), \mathfrak{v}\right]\!\right] = \sum_{p_0=0}^{n} \sum_{p_1 \in \Delta_{p_0}} \sum_{\substack{p_2 \in \Delta_{p_1} \\ p_2 \neq p_0}} \left[\left(\mathrm{Q} \circ \mathrm{P}_{p_0 p_2}^{p_1}\right)(\mathfrak{u}_{p_2}^{p_1}), \mathfrak{v}_{p_1}^{p_0}\right]_{\Gamma_{p_0,p_1}},$$

$$[\![(\Pi\circ\mathbb{T})^n(\mathfrak{u}),\mathfrak{v}]\!]=\sum_{p_0=0}^{n}\sum_{p_1\in\Delta_{p_0}}\sum_{\substack{p_2\in\Delta_{p_1}\\ p_2\neq p_0}}\cdots\sum_{\substack{p_{n+1}\in\Delta_{p_n}\\ p_{n+1}\neq p_{n-1}}}\left[\mathrm{Q}\circ\mathrm{P}^{p_1}_{p_0p_2}\cdots\mathrm{Q}\circ\mathrm{P}^{p_n}_{p_{n-1}p_{n+1}}(\mathfrak{u}^{p_n}_{p_{n+1}}),\mathfrak{v}^{p_0}_{p_1}\right]_{\Gamma_{p_0,p_1}},$$

such that each term in the last equation corresponds to a simple path in the adjacency graph of length $n+1$. Since there is no such path of length P_G+1, we deduce that $\left[\!\!\left[(\Pi\circ\mathbb{T})^{P_G}(\mathfrak{u}),\mathfrak{v}\right]\!\!\right]=0$. Then, we have $(\mathbb{T}\Pi)^{P_G}=\Pi(\Pi\mathbb{T})^{P_G}\Pi=0$, and according to Lemma 4.11, $(\Pi\circ\mathbb{T}+\mathbb{T}\circ\Pi)^{P_G}=(\mathbb{T}\Pi)^{P_G}+(\Pi\mathbb{T})^{P_G}=0$. □

We deduce the following theorem

Theorem 4.14 ([31, Theorem 6.1]). *If $\kappa_0=\cdots=\kappa_n$ and $\mu_0=\cdots=\mu_n$, then*

$$\sigma(\mathbb{A}-\alpha\Pi)=\sigma_p(\mathbb{A}-\alpha\Pi)=\{+\sqrt{1+\alpha^2},-\sqrt{1+\alpha^2}\},$$

where $\sigma(\sigma_p(\mathbb{A}-\alpha\Pi))$ denotes the spectrum of $(\mathbb{A}-\alpha\Pi)$ and $\sigma_p(\mathbb{A}-\alpha\Pi)$, its point spectrum.

Proof. If $\alpha=0$, the result is clear because $\frac{\mathrm{I_d}+\mathbb{A}}{2}$ is a projector. Let us assume now that $\alpha\neq 0$.

First, taking the square of $\mathbb{A}-\alpha\Pi$, we obtain

$$\begin{aligned}(\mathbb{A}-\alpha\Pi)^2&=\mathbb{A}^2+\alpha^2\,\mathrm{I_d}-\alpha(2\mathbb{P}-\mathrm{I_d})\Pi-\alpha\Pi(2\mathbb{P}-\mathrm{I_d})\\&=(1+\alpha^2)\,\mathrm{I_d}-2\alpha(\mathbb{D}\Pi+\Pi\mathbb{D}-\Pi)-2\alpha(\Pi\mathbb{T}+\mathbb{T}\Pi)\\&=(1+\alpha^2)\,\mathrm{I_d}-2\alpha(\Pi\mathbb{T}+\mathbb{T}\Pi),\end{aligned}$$

where we used $(\mathbb{A}^p)^2=\mathrm{I_d}$ and Lemma 4.10. Remark that according to Lemma 4.13, $\Pi\mathbb{T}+\mathbb{T}\Pi$ is nilpotent, and according to the spectral radius formula [135, Theorem 10.13 (b)], we have $\sigma(\Pi\mathbb{T}+\mathbb{T}\Pi)=\{0\}$ and we deduce that $\sigma((\mathbb{A}-\alpha\Pi)^2)=\{1+\alpha^2\}$.

Using the spectral mapping theorem [135, Theorem 10.28], one can show $\sigma(\mathbb{A}-\alpha\Pi)\subset\{+\sqrt{1+\alpha^2},-\sqrt{1+\alpha^2}\}$. According to the spectral mapping theorem for the point spectrum [135, Theorem 10.33], we also have $\sigma_p(\mathbb{A}-\alpha\Pi)\subset\{+\sqrt{1+\alpha^2},-\sqrt{1+\alpha^2}\}$. To finish the proof, let us show that if $\lambda\in\sigma_p(\mathbb{A}-\alpha\Pi)$ then $-\lambda\in\sigma_p(\mathbb{A}-\alpha\Pi)$.

Let $\mathfrak{u}\in\mathbb{H}(\Sigma)\setminus\{0\}$ be the eigenvector associated with the eigenvalue λ, then $(\mathbb{A}-\alpha\Pi)\mathfrak{u}=\lambda\mathfrak{u}$ and notice that $(\mathbb{A}\Pi-\Pi\mathbb{A})(\mathbb{A}-\alpha\Pi)=-(\mathbb{A}-\alpha\Pi)(\mathbb{A}\Pi-\Pi\mathbb{A})$, so that

$$\begin{aligned}(\mathbb{A}-\alpha\Pi)(\mathbb{A}\Pi-\Pi\mathbb{A})\mathfrak{u}&=-(\mathbb{A}\Pi-\Pi\mathbb{A})(\mathbb{A}-\alpha\Pi)\mathfrak{u}\\&=-\lambda(\mathbb{A}\Pi-\Pi\mathbb{A})\mathfrak{u}.\end{aligned}$$

Hence, $-\lambda$ is an eigenvalue of $\mathbb{A}-\alpha\Pi$ if $(\mathbb{A}\Pi-\Pi\mathbb{A})\mathfrak{u}\neq 0$. Notice that

$$\Pi\mathbb{A}\mathfrak{u}=\lambda\Pi\mathfrak{u}+\alpha\mathfrak{u}\quad\text{and}\quad\mathbb{A}\Pi\mathfrak{u}=\frac{1}{\alpha}\mathfrak{u}-\frac{\lambda}{\alpha}\mathbb{A}\mathfrak{u},$$

so that

$$(\Pi\mathbb{A}-\mathbb{A}\Pi)\mathfrak{u}=\left(\alpha-\frac{1}{\alpha}\right)\mathfrak{u}-\frac{\lambda}{\alpha}(\mathbb{A}-\alpha\Pi)\mathfrak{u}.$$

By hypothesis, $\lambda=\pm\sqrt{1+\alpha^2}$, so that the spectrum of $\left(\alpha-\frac{1}{\alpha}\right)\mathrm{I_d}+\frac{\lambda}{\alpha}(\mathbb{A}-\alpha\Pi)$ contains only

$$\alpha-\frac{1}{\alpha}\pm\frac{\lambda}{\alpha}\sqrt{1+\alpha^2}=\frac{\alpha^2-1\pm(1+\alpha^2)}{\alpha}=2\alpha\text{ or }\frac{-2}{\alpha}.$$

Since $\alpha \neq 0$, we deduce that $(\Pi\mathbb{A}-\mathbb{A}\Pi)\mathfrak{u}$ cannot be equal to zero, so that $-\lambda$ is also an eigenvalue of $\mathbb{A}-\alpha\Pi$. □

Corollary 4.15. *If $\kappa_0 = \cdots = \kappa_n$ and $\mu_0 = \cdots = \mu_n$, then*

$$\sigma(\mathrm{MTF}_\alpha) = \sigma_p(\mathrm{MTF}_\alpha) = \{\alpha - \frac{1}{2} + \frac{\sqrt{1+4\alpha^2}}{2}, \alpha - \frac{1}{2} - \frac{\sqrt{1+4\alpha^2}}{2}\},$$

Proof. By definition, we have

$$\begin{aligned}\mathrm{MTF}_\alpha &= \mathbb{P} + (\alpha-1)\,\mathrm{I_d} - \alpha\Pi \\ &= \frac{1}{2}(\mathbb{A} - 2\alpha\Pi + (2\alpha-1)\,\mathrm{I_d}),\end{aligned}$$

and we conclude using Theorem 4.14. □

Notice that it is consistent with the fact that $-\mathrm{MTF}_0$ is a projector in the last corollary.

4.3 Jacobi method

In [33], where local MTF is presented for the DDM community, a Jacobi method for Equation (4.9) is introduced and analysed for two and three subdomains with Dissipative Helmholtz equation. In particular, it is proved that the iteration operator is nilpotent, and it is explicitly shown that this iteration operator is nilpotent of index two for two subdomains and of index four for three subdomains in one dimension, while an explicit equivalence with the optimal Schwarz method is presented for two subdomains in one dimension. We show thereafter a generalization of these results for n subdomains in any dimension, based on algebraic properties introduced in Section 4.2.3.

4.3.1 Definition

First, let us introduce the Jacobi method introduced in [33, Equation (12)] for the Equation (4.9):

$$\mathfrak{u}^{m+1} = \mathbb{J}_{\alpha,(\kappa,\mu)}\mathfrak{u}^m + \mathbb{F} \tag{4.11}$$

where

$$\begin{aligned}\mathbb{J}_{\alpha,(\kappa,\mu)} &= \left(\mathbb{P}_{(\kappa,\mu)} + (\alpha-1)\,\mathrm{I_d}\right)^{-1}\alpha\Pi, \\ \mathbb{F} &= -\left(\mathbb{P}_{(\kappa,\mu)} + (\alpha-1)\,\mathrm{I_d}\right)^{-1}\tau_{(\mu)}(\mathfrak{u}_{inc}).\end{aligned}$$

It should be noticed that for $\alpha \neq 0, 1$, we can do the following computation

$$\begin{aligned}\left(\mathbb{P}_{(\kappa,\mu)} + (\alpha-1)\,\mathrm{I_d}\right)\circ\left(\mathbb{P}_{(\kappa,\mu)} - \alpha\,\mathrm{I_d}\right) &= \mathbb{P}^2_{(\kappa,\mu)} - \alpha\mathbb{P}_{(\kappa,\mu)} + (\alpha-1)\mathbb{P}_{(\kappa,\mu)} - \alpha(\alpha-1)\,\mathrm{I_d} \\ &= -\alpha(\alpha-1)\,\mathrm{I_d},\end{aligned}$$

so that

$$\left(\mathbb{P}_{(\kappa,\mu)} + (\alpha-1)\,\mathrm{I_d}\right)^{-1} = \frac{1}{\alpha(1-\alpha)}\left(\mathbb{P}_{(\kappa,\mu)} - \alpha\,\mathrm{I_d}\right).$$

Then, according to Equation (4.7), we have $\mathrm{A}^0_{\kappa_0}\,\gamma^0(u_{\mathrm{inc}}) = -\gamma^0(u_{\mathrm{inc}})$ so that

$$\begin{aligned}
\mathbb{F} &= \frac{1}{\alpha(\alpha-1)}\left(\mathbb{P}_{(\kappa,\mu)} - \alpha\,\mathrm{I_d}\right)\tau_{(\mu)}(\mathfrak{u}_{inc}) \\
&= \frac{1}{\alpha(\alpha-1)}\left(\frac{\mathrm{I_d}+\mathbb{A}_{(\kappa,\mu)}}{2} - \alpha\,\mathrm{I_d}\right)\circ\tau_{(\mu)}(\mathfrak{u}_{inc}) \\
&= \frac{1}{2\alpha(\alpha-1)}\left(\tau_{\mu_0}\circ\left((1-2\alpha)\,\mathrm{I_d} + \mathrm{A}^0_{\kappa_0}\right)\tau_{\mu_0}^{-1}\circ\tau_{\mu_0}\circ\gamma_0(u_{\mathrm{inc}}), 0, \ldots, 0\right) \\
&= \frac{1}{2\alpha(\alpha-1)}\left(\tau_{\mu_0}\circ\left((1-2\alpha)\,\mathrm{I_d} - \mathrm{I_d}\right)\gamma_0(u_{\mathrm{inc}}), 0, \ldots, 0\right) \\
&= \frac{1}{1-\alpha}\left(\tau_{\mu_0}\circ\gamma_0(u_{\mathrm{inc}}), 0, \ldots, 0\right) = \frac{1}{1-\alpha}\tau_{(\mu)}(\mathfrak{u}_{\mathrm{inc}}).
\end{aligned}$$

To conclude, we have

$$\begin{aligned}
\mathbb{J}_{\alpha,(\kappa,\mu)} &= \frac{1}{1-\alpha}\left(\mathbb{P}_{(\kappa,\mu)} - \alpha\,\mathrm{I_d}\right)\circ\Pi, \\
\mathbb{F} &= \frac{1}{1-\alpha}\tau_{(\mu)}(\mathfrak{u}_{inc}).
\end{aligned}$$

4.3.2 Nilpotent operator and optimal convergence

In [34], it has been proved that the spectral radius, which controls the speed of convergence of the Jacobi solver, is minimal if $\alpha = 0$ for two and three subdomains in the case of $\kappa_0 = \cdots = \kappa_n$ and $\mu_0 = \cdots = \mu_n$. First, we will generalize this property to a general configuration. But, one should notice that α cannot take the value 0 since it is equivalent to imposing no transmission condition (cf. Equation (4.9)). Nevertheless, it will give interesting results that we will justify thereafter.

Theorem 4.16. *If $\kappa_0 = \cdots = \kappa_n$ and $\mu_0 = \cdots = \mu_n$, then for $\alpha \in \mathbb{R}$, we have*

$$\begin{cases}
\sigma(\mathbb{J}_\alpha) = \sigma_p(\mathbb{J}_\alpha) = \{+i\sqrt{\dfrac{\alpha}{1-\alpha}}, -i\sqrt{\dfrac{\alpha}{1-\alpha}}\} & \text{if } |\alpha| < 1 \\
\sigma(\mathbb{J}_\alpha) = \sigma_p(\mathbb{J}_\alpha) = \{+\sqrt{\dfrac{\alpha}{\alpha-1}}, -\sqrt{\dfrac{\alpha}{\alpha-1}}\} & \text{if } |\alpha| > 1
\end{cases}$$

Proof. By definition, we have

$$\begin{aligned}
\lambda \in \sigma((\mathbb{P} - \alpha\,\mathrm{I_d})\circ\Pi) &\iff (\mathbb{P} - \alpha\,\mathrm{I_d})\circ\Pi - \lambda\,\mathrm{I_d} \text{ is not invertible} \\
&\iff \mathbb{P} - \alpha\,\mathrm{I_d} - \lambda\Pi \text{ is not invertible} \\
&\iff \mathbb{A} - 2\lambda\Pi - (2\alpha-1)\,\mathrm{I_d} \text{ is not invertible} \\
&\iff (2\alpha-1)^2 = 1 + 4\lambda^2 \\
&\iff \begin{cases} \lambda = \pm i\sqrt{\alpha(1-\alpha)} & \text{if } |\alpha| \le 1 \\ \lambda = \pm\sqrt{\alpha(\alpha-1)} & \text{if } |\alpha| > 1 \end{cases},
\end{aligned}$$

where we used Theorem 4.14. □

The spectral radius $\rho(\mathbb{J}_\alpha) := \sup\{|\lambda|,\, \lambda \in \sigma(\mathbb{J}_\alpha)\}$ is a critical parameter for the convergence on the Jacobi iteration Equation (4.11). The parameter α must be chosen such that the spectral

radius is as small as possible. According to Theorem 4.16, one should take $\alpha = 0$. The interesting feature of taking $\alpha = 0$ in the Jacobi method is that it gives a convergence in a finite number of iterations since it becomes nilpotent. More precisely, we prove that the operator $\mathbb{J}_0$ is nilpotent of order $P_G + 1$, the length of the adjacency graph associated with the partition of the domain (see Section 4.1.1).

Lemma 4.17. *If $\kappa_0 = \cdots = \kappa_n$ and $\mu_0 = \cdots = \mu_n$, we have $\mathbb{J}_0^{k+1} = (\mathbb{T} \circ \Pi)^{k+1} + \Pi \circ (\mathbb{T} \circ \Pi)^k$ for every integer $k \geq 1$.*

Proof. We prove this relation by induction. For $k = 1$, we have

$$\begin{aligned}\mathbb{J}_0^2 &= ((\mathbb{D} + \mathbb{T}) \circ \Pi)^2 \\ &= (\mathbb{D} \circ \Pi)^2 + (\mathbb{T} \circ \Pi)^2 + \mathbb{D} \circ \Pi \circ \mathbb{T} \circ \Pi + \mathbb{T} \circ \Pi \circ \mathbb{D} \circ \Pi.\end{aligned}$$

Using Lemma 4.10, we have $\mathbb{D} \circ \Pi = \Pi - \Pi \circ \mathbb{D}$ so that $(\mathbb{D} \circ \Pi)^2 = \mathbb{D} \circ \Pi \circ (\Pi - \Pi \circ \mathbb{D}) = 0$ and the last expression becomes

$$\mathbb{J}_0^2 = (\mathbb{T} \circ \Pi)^2 + \Pi \circ \mathbb{T} \circ \Pi - \Pi \circ \mathbb{D} \circ \mathbb{T} \circ \Pi + \mathbb{T} - \mathbb{T} \circ \mathbb{D}.$$

Since $\mathbb{T} \circ \mathbb{D} = \mathbb{T}$ and $\mathbb{D} \circ \mathbb{T} = 0$ according to Lemma 4.12, we obtain

$$\mathbb{J}_0^2 = (\mathbb{T} \circ \Pi)^2 + \Pi \circ \mathbb{T} \circ \Pi.$$

which is the expected relation. For $k > 1$, let us assume we have

$$\mathbb{J}_0^k = (\mathbb{T} \circ \Pi)^k + \Pi \circ (\mathbb{T} \circ \Pi)^{k-1}.$$

Then, we obtain

$$\begin{aligned}\mathbb{J}_0^{k+1} &= (\mathbb{D} \circ \Pi + \mathbb{T} \circ \Pi) \circ \left[(\mathbb{T} \circ \Pi)^k + \Pi \circ (\mathbb{T} \circ \Pi)^{k-1}\right] \\ &= \mathbb{D} \circ \Pi \circ (\mathbb{T} \circ \Pi)^k + \mathbb{D} \circ (\mathbb{T} \circ \Pi)^{k-1} \\ &\quad + (\mathbb{T} \circ \Pi)^{k+1} + \mathbb{T} \circ (\mathbb{T} \circ \Pi)^{k-1}.\end{aligned}$$

The second and the last terms vanish according to Lemmas 4.11 and 4.12. Simplifying the first term with the relation obtained in Lemma 4.10 gives

$$\mathbb{J}_0^{k+1} = (\mathbb{T} \circ \Pi)^{k+1} + \Pi \circ (\mathbb{T} \circ \Pi)^k.$$

By induction, we proved the expected relation. □

Theorem 4.18. *If $\kappa_0 = \cdots = \kappa_n$ and $\mu_0 = \cdots = \mu_n$, then the Jacobi method defined in Equation (4.11) converges in $P_G + 1$ iterations.*

Proof. We just combine Lemmas 4.13 and 4.17. □

Despite its convergence property, one could argue that it does not really make sense to take $\alpha = 0$. But actually, it can be justified by the following remark: the equation obtained with $\alpha = 0$ is

$$\mathfrak{u}^{m+1} = \mathbb{P} \circ \Pi \mathfrak{u}^m + \mathfrak{u}_{inc}, \tag{4.12}$$

which can be seen as a block Jacobi method applied to

$$\mathrm{MTF}_1 \circ \Pi(u) = (\mathbb{P} - \Pi)\,\Pi(\mathfrak{u}) = (\mathbb{J}_0 - \mathrm{I_d})\mathfrak{u} = -\mathfrak{u}_{inc}. \tag{4.13}$$

This last equation is consistent with Equation (4.9) for $\alpha = 1$ since the solution must satisfy $\mathfrak{u} = \Pi\mathfrak{u}$. In other words, taking $\alpha = 0$ in Equation (4.11) is equivalent to solving Equation (4.9) with a Jacobi method, $\alpha = 1$, and Π as right preconditioner.

In [96, Section 5.3], the authors introduced another preconditioning strategy for Equation (4.9) with $\alpha = \frac{1}{2}$ where the diagonal part of the operator $\mathrm{MTF}_{1/2,(\kappa,\mu)}$, that is to say $\mathbb{A}_{(\kappa,\mu)}$, is used as a preconditioner. But notice that we have the following relation

$$\begin{aligned}
\mathbb{A}_{(\kappa,\mu)}\,\mathrm{MTF}_{1/2,(\kappa,\mu)} &= \mathbb{A}_{(\kappa,\mu)}\frac{1}{2}(\mathbb{A}_{(\kappa,\mu)} - \Pi) \\
&= \frac{1}{2}(\mathrm{I_d} - \mathbb{A}_{(\kappa,\mu)}\Pi) \\
&= -(\mathbb{A}_{(\kappa,\mu)} - \Pi)\Pi \\
&= -\,\mathrm{MTF}_{1/2,(\kappa,\mu)}\,\Pi,
\end{aligned}$$

using $\mathbb{A}^2_{(\kappa,\mu)} = \mathrm{I_d}$ and $\Pi^2 = \mathrm{I_d}$. So that, the preconditioning strategy suggested in [96, Section 5.3] corresponds also to a right preconditioning by Π but with $\alpha = \frac{1}{2}$.

4.3.3 Equivalence with Optimal Schwarz Methods

According to Theorem 4.18, we have convergence of the Jacobi method applied to Equation (4.9) in $P_G + 1$ iterations. Thus, for the case of n concentric ring, we have convergence in n iterations. This last result is similar to the one we can obtain with Optimal Schwarz Method (OSM) (cf. [128] and Section 3.1.2). This suggests a link between MTF and OSM and this is what we are going to clarify.

Let us define the following bounded one-to-one operators

$$\mathrm{X}^p: \begin{array}{l} H^{1/2}(\Gamma_p) \to \mathcal{H}(\Gamma_p), \\ \qquad u \mapsto (u, \mathrm{DtN}^p(u)) \end{array} \quad \text{and} \quad \mathrm{Y}^p: \begin{array}{l} \mathcal{H}(\Gamma_p) \to H^{-1/2}(\Gamma_p). \\ \qquad (u,p) \mapsto p + \widetilde{\mathrm{DtN}}^p(u) \end{array},$$

where DtN^p is the Dirichlet-to-Neumann operator defined as in Section 3.1.2 associated with Ω_p, while $\widetilde{\mathrm{DtN}}^p$ is the Dirichlet-to-Neumann operator associated with $\mathbb{R}^d \setminus \overline{\Omega}_p$. We already defined these notations in Section 3.1.2, but we remind them here:

$$\begin{cases} \mathrm{DtN}^p : H^{1/2}(\partial\Omega_p) \to H^{-1/2}(\partial\Omega_p) \\ \qquad\qquad v \mapsto \nabla w \cdot \mathbf{n}_p \end{cases} \quad \text{and} \quad \begin{cases} \widetilde{\mathrm{DtN}}^p : H^{1/2}(\partial\Omega_p) \to H^{-1/2}(\partial\Omega_p) \\ \qquad\qquad v \mapsto -\nabla \widetilde{w} \cdot \mathbf{n}_p \end{cases}$$

where $\mathbf{n}_p$ is the outer normal relative to Ω_p, while w and $\widetilde{w}$ respectively satisfy the following problems:

$$\begin{cases} \mathrm{L}(w) = 0 \text{ in } \Omega_p, \\ \quad w = v \text{ on } \partial\Omega_p. \end{cases} \quad \text{and} \quad \begin{cases} \mathrm{L}(\widetilde{w}) = 0 \text{ in } \Omega_p^c = \mathbb{R}^d \setminus \overline{\Omega}_p, \\ \quad w = v \text{ on } \partial\Omega_p. \end{cases}$$

If Ω_p or Ω_p^c is unbounded, we add the appropriate conditions at infinity for the problem to be well-posed (see Section 4.1.2). More generally, we use a tilde to denote the operators associated with Ω_p^c.

Lemma 4.19. *We have the following relations*

$$\mathrm{DtN}^p = (\mathrm{V}^p)^{-1}\left(\frac{\mathrm{I_d}}{2} - \mathrm{K}^p\right) = \left(\frac{\mathrm{I_d}}{2} - \mathrm{K}'^p\right)^{-1}\mathrm{W}^p,$$
$$\widetilde{\mathrm{DtN}}^p = (\mathrm{V}^p)^{-1}\left(\frac{\mathrm{I_d}}{2} + \mathrm{K}^p\right) = \left(\frac{\mathrm{I_d}}{2} + \mathrm{K}'^p\right)^{-1}\mathrm{W}^p\,.$$

Proof. The first relation is just a reminder of Theorem 2.27. According to Theorem 2.22, we have

$$[\gamma^{\Omega_p}] \circ \mathrm{G_L}^{\Omega_p} = \mathrm{I_d},$$

where the superscript Ω_p means that the operator is relative to Ω_p and its outer normal $\mathbf{n}_p$. We deduce that

$$\begin{aligned}\gamma^{\Omega_p} \circ \mathrm{G_L}^{\Omega_p} &= \mathrm{I_d} + \gamma_c^{\Omega_p} \circ \mathrm{G_L}^{\Omega_p} \\ &= \mathrm{I_d} + \mathrm{Q}\,\gamma^{\Omega_p^c} \circ \mathrm{G_L}^{\Omega_p}\,.\end{aligned}$$

We can rewrite the last expression in matrix form using the operators defined in

$$\begin{aligned}\begin{bmatrix}\frac{1}{2}\mathrm{I_d} + \mathrm{K}^p & \mathrm{V}^p \\ \mathrm{W}^p & \frac{1}{2}\mathrm{I_d} + \mathrm{K}'^p\end{bmatrix} &= \begin{bmatrix}\mathrm{I_d} & 0 \\ 0 & \mathrm{I_d}\end{bmatrix} + \mathrm{Q}\begin{bmatrix}\gamma_D^{\Omega_p^c} \\ \gamma_N^{\Omega_p^c}\end{bmatrix}\begin{bmatrix}\mathrm{DL}^{\Omega_p} & \mathrm{SL}^{\Omega_p}\end{bmatrix} \\ &= \begin{bmatrix}\mathrm{I_d} & 0 \\ 0 & \mathrm{I_d}\end{bmatrix} + \begin{bmatrix}\gamma_D^{\Omega_p^c} \circ \mathrm{DL}^{\Omega_p} & \gamma_D^{\Omega_p^c} \circ \mathrm{SL}^{\Omega_p} \\ -\gamma_N^{\Omega_p^c} \circ \mathrm{DL}^{\Omega_p} & -\gamma_N^{\Omega_p^c} \circ \mathrm{SL}^{\Omega_p}\end{bmatrix}.\end{aligned}$$

Notice that we have $\mathrm{SL}^{\Omega_p} = \mathrm{SL}^{\Omega_p^c}$ and $\mathrm{DL}^{\Omega_p} = -\mathrm{DL}^{\Omega_p^c}$ (cf. definitions in Section 2.3.1), so that

$$\begin{aligned}\begin{bmatrix}\frac{1}{2}\mathrm{I_d} + \mathrm{K}^p & \mathrm{V}^p \\ \mathrm{W}^p & \frac{1}{2}\mathrm{I_d} + \mathrm{K}'^p\end{bmatrix} &= \begin{bmatrix}\mathrm{I_d} & 0 \\ 0 & \mathrm{I_d}\end{bmatrix} + \begin{bmatrix}-\gamma_D^{\Omega_p^c} \circ \mathrm{DL}^{\Omega_p^c} & \gamma_D^{\Omega_p^c} \circ \mathrm{SL}^{\Omega_p^c} \\ \gamma_N^{\Omega_p^c} \circ \mathrm{DL}^{\Omega_p^c} & -\gamma_N^{\Omega_p^c} \circ \mathrm{SL}^{\Omega_p^c}\end{bmatrix} \\ &= \begin{bmatrix}\mathrm{I_d} & 0 \\ 0 & \mathrm{I_d}\end{bmatrix} + \begin{bmatrix}-\frac{1}{2}\mathrm{I_d} - \widetilde{\mathrm{K}}^p & \widetilde{\mathrm{V}}^p \\ \widetilde{\mathrm{W}}^p & -\frac{1}{2}\mathrm{I_d} - \widetilde{\mathrm{K}}'^p\end{bmatrix}.\end{aligned}$$

We deduce that

$$\mathrm{K}^p = -\widetilde{\mathrm{K}}^p, \quad \mathrm{V}^p = \widetilde{\mathrm{V}}^p,$$
$$\mathrm{W}^p = \widetilde{\mathrm{W}}^p, \quad \mathrm{K}'^p = -\widetilde{\mathrm{K}}'^p,$$

and then, we just use Theorem 2.27 to obtain the second relation. □

Using the previous lemma, we can relate the operators X^p and Y^p with P^p

Lemma 4.20. *We have the following relations*

$$\mathrm{P}^p = \mathrm{X}^p \circ \mathrm{V}^p \circ \mathrm{Y}^p \quad \textit{and} \quad \mathrm{V}^p \circ \mathrm{Y}^p \circ \mathrm{X}^p = \mathrm{I_d}$$

Proof. Using matrix notations, we have

$$\begin{aligned}\begin{bmatrix} \mathrm{I_d} \\ \mathrm{DtN}^p \end{bmatrix} \circ \mathrm{V}^p \circ \begin{bmatrix} \widetilde{\mathrm{DtN}}^p & \mathrm{I_d} \end{bmatrix} &= \begin{bmatrix} \mathrm{V}^p\,\widetilde{\mathrm{DtN}}^p & \mathrm{V}^p \\ \mathrm{DtN}^p\,\mathrm{V}^p\,\widetilde{\mathrm{DtN}}^p & \mathrm{DtN}^p\,\mathrm{V}^p \end{bmatrix}, \\ &= \begin{bmatrix} \dfrac{1}{2} + \mathrm{K}^p & \mathrm{V}^p \\ \mathrm{DtN}^p\,\mathrm{V}^p\,\widetilde{\mathrm{DtN}}^p & \mathrm{DtN}^p\,\mathrm{V}^p \end{bmatrix},\end{aligned}$$

where we used Lemma 4.19. Then, we have $\mathrm{DtN}^p\,\mathrm{V}^p = \frac{\mathrm{I_d}}{2} + \mathrm{K}'^p$ with Lemma 4.19 and Equation (2.9a), so that $\mathrm{P}^p = \mathrm{X}^p \circ \mathrm{V}^p \circ \mathrm{Y}^p$. Concerning the second equation, we have

$$\mathrm{V}^p \circ \mathrm{Y}^p \circ \mathrm{X}^p = \mathrm{V}^p \circ \left(\mathrm{DtN}^p + \widetilde{\mathrm{DtN}}^p\right) = \mathrm{I_d},$$

using Lemma 4.19. □

We can now exhibit the link between the OSM and the local MTF in the geometric setting described in Section 4.1.1. Let us look at the error $\mathfrak{e}^m = \mathfrak{u} - \mathfrak{u}^m \in \mathbb{H}(\Sigma)$ of the Jacobi method given in Equation (4.11) with $\mathfrak{u}$ the trace of the solution of Equation (4.2) and $\alpha = 0$, we have

$$\mathfrak{e}^{m+1} = \mathbb{P} \circ \Pi(\mathfrak{e}^m),$$

which means that, for every $p = 0, \ldots, n$, we have

$$(\mathfrak{e}^p)^{m+1} = \mathrm{P}^p \circ \sum_{k \in \Delta_p} (\mathrm{R}_p^k)^* \, \mathrm{Q}(\mathfrak{e}_p^k)^m.$$

Then, we can apply $\mathrm{V}^p \circ \mathrm{Y}^p$ and Lemma 4.20 to obtain

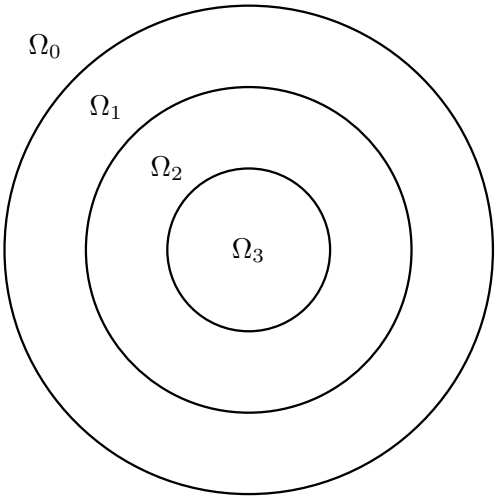
$$\mathrm{V}^p \circ \mathrm{Y}^p (\mathfrak{e}^p)^{m+1} = \mathrm{V}^p \circ \mathrm{Y}^p \circ \sum_{k \in \Delta_p} (\mathrm{R}_p^k)^* \, \mathrm{Q}(\mathfrak{e}_p^k)^m. \tag{4.14}$$

We can define $\widetilde{\mathrm{DtN}}_{kl}^p$ with $k, l \in \Delta_p$ as follows

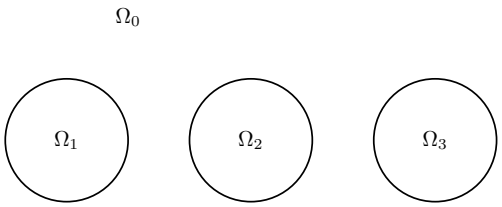
$$\begin{aligned}\widetilde{\mathrm{DtN}}_{kl}^p : H^{1/2}(\Gamma_{j,l}) &\to H^{-1/2}(\Gamma_{j,k}) \\ v_l^p &\mapsto \mathrm{R}_{k,N}^p \circ \widetilde{\mathrm{DtN}}^p \circ (\mathrm{R}_{l,D}^p)^*(v_l^p)\end{aligned}$$

where $\mathrm{R}_{k,N}^p$ and $(\mathrm{R}_{l,D}^p)^*$ are the restriction and extension operators applied to, respectively, a Neumann trace and a Dirichlet trace. Notice that if $k = l$, this operator is the Dirichlet-to-Neumann operator associated with the connected component of $\mathbb{R}^d \setminus \overline{\Omega}_p$ such that its intersection with $\overline{\Omega}_p$ is $\Gamma_{j,k}$ (similar to the operator $\Lambda_{j,r}$ or $\Lambda_{j,l}$ in [128]). Otherwise, if $k \neq l$, the operator is null. Thus multiplying by $(\mathrm{V}^p)^{-1}$ and applying the restriction operator $\mathrm{R}_{k,N}^p$ to Equation (4.14), we obtain

$$\widetilde{\mathrm{DtN}}_{kk}^p \left((e_k^p)^{m+1}\right) + (r_k^p)^{m+1} = \widetilde{\mathrm{DtN}}_{kk}^p \left((e_p^k)^m\right) - (r_p^k)^m$$



(a) Configuration I



(b) Configuration II

Figure 4.3 – Considered geometric configurations in our numerical tests with the local multi-trace formulation with P_G varying or constant.

where $\mathfrak{e}_k^p = (e_k^p, r_k^p) \in \mathcal{H}(\Gamma_p)$ for $1 \leq p \leq n$ and $k \in \Delta_p$. This is exactly the transmission condition used in the OSM as described in Equation (3.5). Thus, we understand why the convergence properties are similar, both methods actually perform the same iteration on the traces.

4.4 Numerical evidences

In this section, we present numerical results conducted on PDEs in 2D that illustrate the conclusions of the previous sections. Denote $\mathrm{D}(\mathbf{x}, r)$, the disk of center $\mathbf{x}$ and radius $r > 0$. We introduce two geometric configurations that we will use in our numerical experiments, they are illustrated in Figure 4.3, and defined as follows:

- **Configuration I**: It consists in centered nested disks. Considering a sequence of radii $r_p = n - p$, we set $\Omega_0 := \mathbb{R}^2 \setminus \overline{\mathrm{D}}(0, r_0)$, and then $\Omega_p := \mathrm{D}(0, r_{p-1}) \setminus \overline{\mathrm{D}}(0, r_p)$, for $p = 1, \ldots, n-1$, and $\Omega_n := \mathrm{D}(0, 1)$. In this configuration, the depth of the adjacency graph P_G equals n, i.e. it grows linearly with the number of subdomains, see Figure 4.3a.

- **Configuration II**: We define a sequence of centers $\mathbf{x}_p = (3(p-1), 0)$, and we consider translated unit disks $\Omega_p = \mathrm{D}(\mathbf{x}_p, 1)$, for $p = 1, \ldots, n$ with $\Omega_0 = \mathbb{R}^2 \setminus (\overline{\Omega}_1 \cup \cdots \cup \overline{\Omega}_n)$. In this configuration, the depth of the adjacency graph P_G systematically equals 2 for $n > 1$, see Figure 4.3b.

The point of these two geometries is to have one test case where P_G increases linearly, Configuration I, and another test case where it is constant, Configuration II. We will use these two test cases to illustrate Theorem 4.18. Note that we use GMSH [60] to discretize the considered geometry. More precisely, we set the length of the mesh cells to 0.05 for Configuration I and to 0.1 for Configuration II.

4.4.1 Discretization

We solve Equation (4.9) with a Galerkin discretization as described in Section 2.4.5 using BemTool[2]. We approximate $\mathbb{H}(\Sigma)$ by $\mathcal{V}_h := \prod_{p=0}^{n} \mathcal{V}_h(\Gamma_p) \times \mathcal{V}_h(\Gamma_p)$, where $\mathcal{V}_h(\Gamma_p)$ is a $\mathbb{P}_1$-Lagrange function space over a regular mesh (here a set of straight panels forming a polygonal line), with h as the maximal length of the panels. After discretization, Equation (4.9) becomes

$$\mathbf{MTF}_{\alpha,(\kappa,\mu)}(\mathfrak{u}_h) = (\mathbf{P}_{(\kappa,\mu)} - \mathbf{M} + \alpha(\mathbf{M} - \mathbf{\Pi}))\mathfrak{u}_h = -\mathbf{M}\tau_{(\mu)}(\mathfrak{u}_{h,\mathrm{inc}}), \tag{4.15}$$

where $\mathfrak{u}_h, \mathfrak{u}_{\mathrm{inc},h} \in \mathcal{V}_h$, while $\mathbf{P}_{(\kappa,\mu)}$, $\mathbf{M}$ are $\mathbf{\Pi}$ are the Galerkin matrices associated with, respectively $\mathbb{P}_{(\kappa,\mu)}$, $\mathrm{I_d}$ and Π. More precisely, let us denote $((\varphi_h^j)_{j=0}^{N})$ the boundary element basis, where N is the number of nodes in the mesh. It should be noticed that each function (φ_h^j) is used four times. Indeed, for each node, we have one degree of freedom for the Dirichlet trace and one for the Neumann trace on each side of the boundary.

We denote the basis functions associated with Γ_p $((\psi_h^j)_{j=N_{p-1}}^{N_p-1})$, such that $(N_{-1} = 0)$, $(N_n = 2N)$ and $(N_p - N_{p-1})$ is the number of nodes on Γ_p. Remark that every boundary function (φ_h^j) appear twice in $((\psi_h^j)_{j=0}^{2N-1})$, for example if $(\Gamma_{pk} \neq \emptyset)$, $((\psi_h^j)_{j=N_{p-1}}^{N_p-1})$ and $((\psi_h^j)_{j=N_{k-1}}^{N_k-1})$ both contain the functions (φ_h^j) located on (Γ_{pk}).

$$\mathfrak{u}_h^p := (\mathfrak{u}_h)_{j=2N_{p-1}}^{2N_p-1} = ((\mathfrak{u}_h^p)_D, (\mathfrak{u}_h^p)_N)^T,$$

$$\mathbf{M}^p := (\mathbf{M})_{j,k=2N_{p-1}}^{2N_p-1} = \begin{pmatrix} \left(\int_{\Gamma_p} \psi_j\psi_k\right)_{j,k=N_{p-1}}^{N_p-1} & 0 \\ 0 & \left(\int_{\Gamma_p} \psi_j\psi_k\right)_{j,k=N_{p-1}}^{N_p-1} \end{pmatrix},$$

$$\mathbf{A}_{(\kappa,\mu)}^p := \left(\mathbf{A}_{(\kappa,\mu)}\right)_{j,k=2N_{p-1}}^{2N_p-1} = 2\begin{pmatrix} \mathbf{K}^p & \mu^p\mathbf{V}^p \\ \mathbf{W}^p/\mu^p & \mathbf{K}'^p \end{pmatrix}, \quad \mathbf{P}_{(\kappa,\mu)}^p = \frac{\mathbf{A}_{(\kappa,\mu)}^p + \mathbf{M}^p}{2}$$

$$(\mathbf{\Pi})_{j=2N_{p-1},k=2N_{l-1}}^{2N_p-1,2N_l-1} := \begin{pmatrix} \left(\int_{\Gamma_{p,l}} \psi_j\psi_k\right)_{j=N_{p-1},k=m_{l-1}}^{N_p-1,m_l-1} & 0 \\ 0 & -\left(\int_{\Gamma_{p,l}} \psi_j\psi_k\right)_{j=N_{p-1},k=N_{l-1}}^{N_p-1,m_l-1} \end{pmatrix},$$

$$\mathbf{F} := \mathbf{M}\tau_{(\mu)}(\mathfrak{u}_{\mathrm{inc},h})$$

Notice that $\mathbf{A}_{(\kappa,\mu)}^p$ and $\mathbf{M}^p$ are block diagonal matrices where each block corresponds to a subdomain. And these blocks are applied to pair of traces, for example in $\mathbf{M}^p$ the first diagonal block is applied to Dirichlet traces and the second one to Neumann traces. Meanwhile, $\mathbf{\Pi}$ has only extra diagonal blocks corresponding to interfaces between subdomains. The other coefficients we have not defined yet vanish.

We are interested in solving Equation (4.15) for several values of α and various choices of $(\kappa_0, \ldots, \kappa_n)$ and $(\mu_0, \ldots, \mu_n)$. As explained in the previous sections, a good choice would be $\alpha = 1$ with Π as a right preconditioner, so that in the homogeneous case $\mathrm{MTF}_1 \circ \Pi = \mathbb{J}_0 - \mathrm{I_d}$ is a nilpotent perturbation of the identity.

Remark 4.21 (Implementation and parallelization)**.** *All the discretized operators, except* $\mathbf{\Pi}$ *are block diagonal. But notice that we have the relation* $\mathbf{\Pi}^{-1} = \mathbf{M}^{-1}\Pi$*, where* Π *exchanges traces as before, but applied to finite element functions. It means that an iterative solver using only matrix vector multiplications by the discretized operator and* $\mathbf{\Pi}^{-1}$ *would be very parallel. Each block can be assembled independently of one another, each global matrix vector multiplication can be*

[2] https://github.com/xclaeys/BemTool

performed as local matrix vector products in each subdomain, and exchanging trace is also quite local since it is only between neighboring subdomains. Such a distributed parallelism is usually performed with an implementation of the Message Passing Interface *(MPI) standard, where each subdomain is associated with a processor. Thus the exchange can be performed efficiently defining a MPI* distributed graph topology*, more precisely, we define in MPI a graph like the adjacency graph G defined in Section 4.1.1 to set how the subdomains/processors are related to one another. This allows a reranking of the processors to optimize the hardware to the communication patterns. The term* distributed *comes from the fact that this MPI topology is scalable, contrary to the usual MPI graph topology (see [101] for more details). For example, we only give to each subdomain its neighbors, we do not define the whole adjacency graph in each processor.*

4.4.2 Homogeneous material

We first examine the spectrum of the local multi-trace operator preconditioned by Π for several values of α with homogeneous coefficients $\kappa_0 = \cdots = \kappa_n = 1$ and $\alpha_0 = \cdots = \alpha_n = 1$. At the continuous level, we are interested in finding $\mathfrak{u} \neq 0$ and $\lambda \in \mathbb{C}$ such that $\mathrm{MTF}_\alpha \Pi \mathfrak{u} = \lambda \mathfrak{u}$, which is equivalent to determining $\mathfrak{v} \neq 0$ and $\lambda \in \mathbb{C}$ such that $\mathrm{MTF}_\alpha \mathfrak{v} = \lambda \Pi \mathfrak{v}$. The discrete counterpart of this eigenvalue problem writes

$$\mathbf{MTF}_\alpha(\mathfrak{v}_h) = \lambda \mathbf{\Pi} \mathfrak{v}_h. \tag{4.16}$$

The results are given in Figure 4.4 with three interfaces and a mesh size equal to 0.1, where Helmholtz equation stands for $\kappa \in \mathbb{R}$, while Dissipative Helmholtz equation corresponds to $\kappa \in i\mathbb{R}$. As expected, the case $\alpha = 1$ leads to one cluster of eigenvalues around the real value -1 since in this case, $\mathrm{MTF}_1 \circ \Pi$ is a nilpotent perturbation of $-\mathrm{I_d}$ according to Equation (4.13), and this regardless of the values of (κ) as long as it is uniform.

We also want to illustrate the convergence of the Jacobi method described in Section 4.3. As we have seen in this section, an optimal choice is to solve Equation (4.9) with $\alpha = 1$ and preconditioned by Π using Equation (4.12). In the discrete setting, this means that we want to solve Equation (4.15) with $\alpha = 1$, i.e.

$$(\mathbf{P} - \mathbf{\Pi})\mathfrak{u}_h = -\mathbf{M}(\mathfrak{u}_{h,\mathrm{inc}}),$$

using the change of variable $\mathbf{\Pi}\mathfrak{u}_h = \mathfrak{v}_h$, and the following Jacobi method

$$\mathfrak{v}_h^{m+1} = \mathbf{P}\mathbf{\Pi}^{-1}\mathfrak{v}_h^m + \mathbf{M}\mathfrak{u}_{h,\mathrm{inc}}, \tag{4.17}$$

we obtain the results from Figure 4.5 for a mesh size equal to 0.1, such that we always obtain a relative error of the order 10^{-3} compared to the solution u_{inc}. The number of iterations is constant in Configuration II, while it increases linearly in Configuration I, with the number of interfaces for Configuration I. This is expected since P_G is increasing with the number of interfaces in Configuration I and remains constant in Configuration 2 (expect between one and two interfaces). But notice that the number of iterations is not exactly the one expected from Theorem 4.18, there are about one more iteration, this is probably due to the numerical approximation, as we can see in Figure 4.4, the discretized operator is not exactly nilpotent. Notice that $\mathbf{\Pi}^{-1} = \mathbf{M}^{-1}\Pi$ as explained in Remark 4.21.

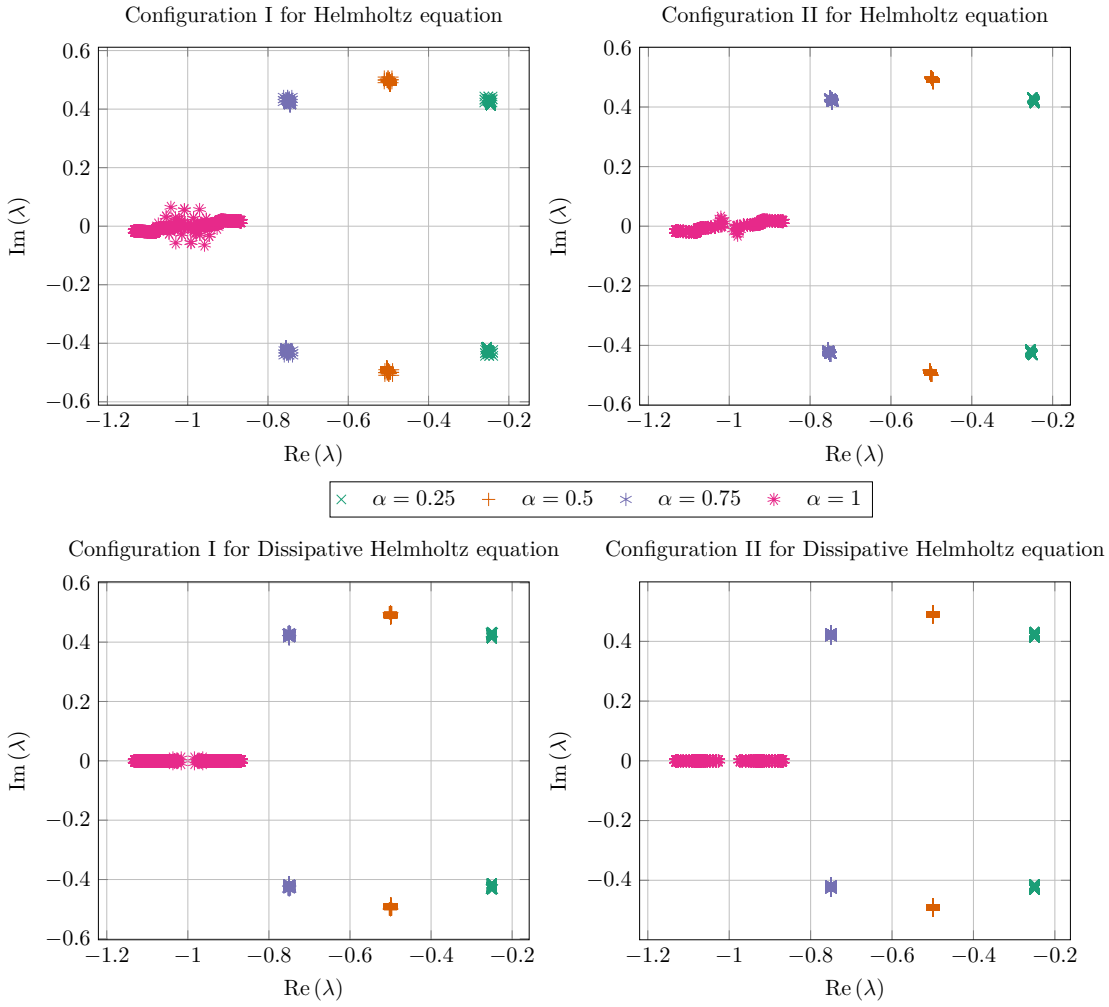


Figure 4.4 – Distribution of eigenvalues for the discretized local multi-trace operator (see Equation (4.16)) for $n = 3$, $h = 0.1$ and a homogeneous material.

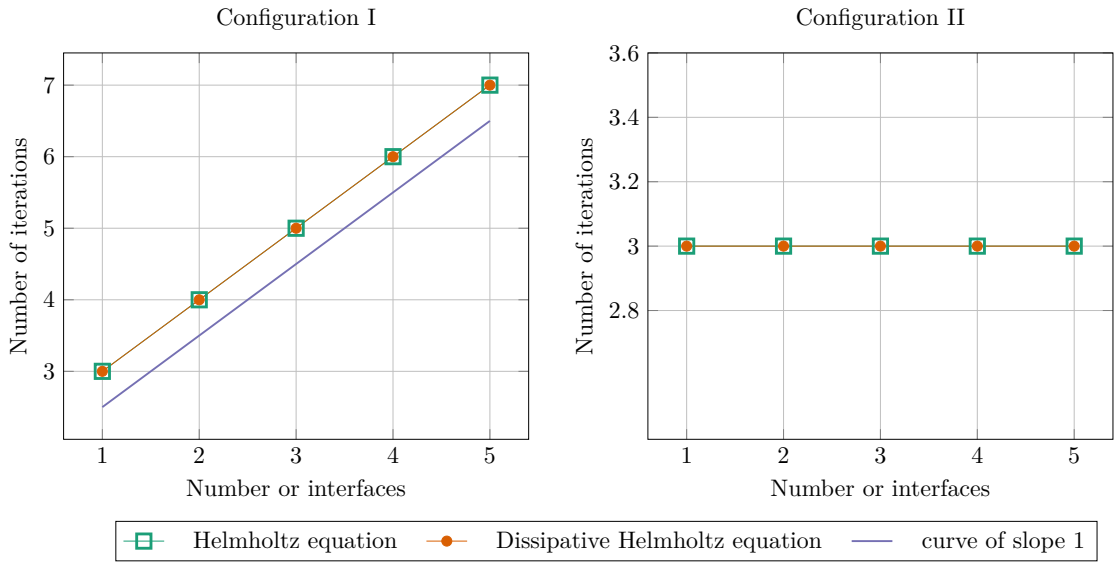


Figure 4.5 – Number of iterations to solve the local multi-trace formulation using the Jacobi method defined in Equation (4.17).

4.4.3 Heterogeneous material

In practice, we are also interested in solving problems with heterogeneous material. But in this case, the nilpotent property of the Jacobi operator does not hold anymore. Indeed, the Jacobi method defined in Equation (4.17) may not converge when applied to $\mathbf{P}_{(\kappa,\mu)}$ instead of just $\mathbf{P}$, which can be observed numerically.

It is well-known that fixed-point methods such as Jacobi do not converge very fast, because it is an algorithm without "memory", meaning that each iteration is just computed from the previous one, while Krylov methods such as the Conjugate Gradient method (CG) [83] or the Generalized Minimal Residual method (GMRes) [137] take into account all the previous iterations via Krylov spaces. It is common in DDM to define a preconditioner from the continuous point of view leading to a fixed-point method, and to use it in a Krylov solver as we saw in Section 3.3.

Similarly, we can still study numerically the efficiency of using $\mathbf{\Pi}^{-1}$ as a preconditioner for solving Equation (4.15) using GMRes, and we will see that $\alpha = 1$ will still be the optimal value, in the sense that it needs fewer iterations to be solved. We still expect interesting results in the case where (κ) is not uniform because it only induces compact perturbations of the formulation.

First, we can look at the spectrum of the following eigenvalue problem

$$\mathbf{MTF}_{\alpha,(\kappa,\mu)}(\mathfrak{v}_h) = \lambda \mathbf{\Pi} \mathfrak{v}_h, \tag{4.18}$$

which is really similar to Equation (4.16). Let us take $n = 3$, $\mu_0 = \mu_1 = \mu_2 = 1$ and $\kappa_0 = 1.00002$, $\kappa_1 = 1.26308$, $\kappa_2 = 2.51121$, values between 1 and 3 obtained randomly. The resulting spectrum is given in Figure 4.6 where we can still observe the same clusters of eigenvalues as in Figure 4.4 but more scattered.

Then, we solve Equation (4.15) using GMRes. We set the maximum number of iterations to 10 000, a restart at 200 iterations and the tolerance is equal to 10^{-6} to obtain at least a relative

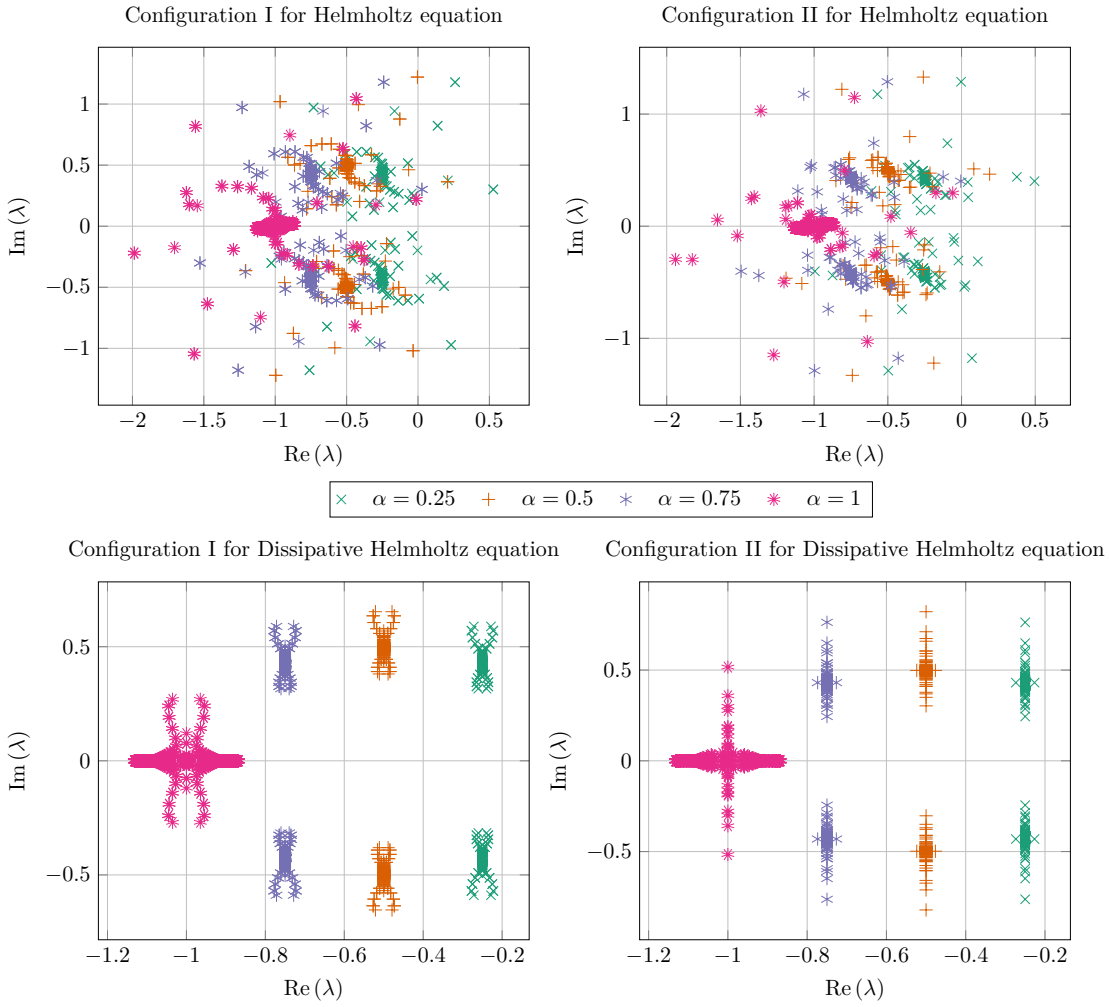


Figure 4.6 – Distribution of eigenvalues for the discretized local multi-trace operator (see Equation (4.18)) for $n = 3$, $\kappa_0 = 1.00002$, $\kappa_1 = 1.26308$ and $\kappa_2 = 2.51121$.

error of 10^{-3} in the homogeneous case. The preconditioner is $\mathbf{\Pi}^{-1} = \mathbf{M}^{-1}\Pi$ as explained in Remark 4.21. We show in Figure 4.7 the number of iterations of GMRes as a function of the number of interfaces for $\mu_0 = \cdots = \mu_5 = 1$ and $\kappa_0 = 1.00002$, $\kappa_1 = 1.26308$, $\kappa_2 = 2.51121$, $\kappa_4 = 2.06553$ and $\kappa_5 = 1.43792$. First, notice that we showed results without preconditioner just to justify the need of a preconditioner (we will not show them anymore to lighten the presentation), notice that the convergence is always faster except for $\alpha = 0.25$. We can observe that for the Helmholtz equation the number of iterations increases regardless of the geometric configuration, while it is constant for Dissipative Helmholtz. We also notice that Configuration II makes the problem easier to solve, which is consistent with the case of a homogeneous material. In particular for the Helmholtz equation, we see that the number of iterations increases less in Configuration II than in Configuration I. Finally, we remark that $\alpha = 1$ with $\mathbf{\Pi}^{-1}$ is still the best way to solve Equation (4.15).

If the variations between the κ_j's increase, the number of iterations increases. Taking the following randomly obtained values between 1 and 41, $\kappa_0 = 1.00031$, $\kappa_1 = 6.26151$, $\kappa_2 = 31.2242$, $\kappa_3 = 19.346$, $\kappa_4 = 22.3107$ and $\kappa_5 = 9.75837$, we obtain the results presented in Figure 4.8. In this case, the solver never converges without preconditioner for Helmholtz equation. Configuration I is really difficult to handle, the only cases that converge are $\alpha = 0.75$ and $\alpha = 1$ with the preconditioner, and they converge after about 6000 iterations. Concerning Configuration II for Helmholtz equation, the only cases that converge are still $\alpha = 0.75$ and $\alpha = 1$ with the preconditioner, but the number of iterations is more reasonable, about 200 iterations. Again, we see that the formulation associated with the preconditioner $\mathbf{\Pi}^{-1}$ is more adapted to this case. The case of Dissipative Helmholtz is less problematic, and we observe that the case $\alpha = 1$ with the preconditioner gives the best results.

We are also interested in looking at variations in (μ). We take randomly obtained values between 1 and 100, $\mu_0 = 1.00078$, $\mu_1 = 14.1538$, $\mu_2 = 76.5605$, $\mu_3 = 46.865$, $\mu_4 = 54.2767$ and $\mu_5 = 22.8959$ with $\kappa_0 = \cdots = \kappa_5 = 1$. Similarly to the previous numerical experiments, we obtain Figure 4.9. The results are similar to the case where we added a small perturbation to (κ), the case $\alpha = 1$ with the preconditioner gives the best results. The number of iterations increases, but variation is small compared to the one observed with important variations in (κ).

To conclude, this method can be particularly useful to solve piecewise constant problems in (κ) with Dissipative Helmholtz equation. Concerning Helmholtz equation, it gives interesting results for a small P_G, but it is not adequate otherwise. Concerning variations in (μ), the method is robust but it lacks a theoretical explanation.

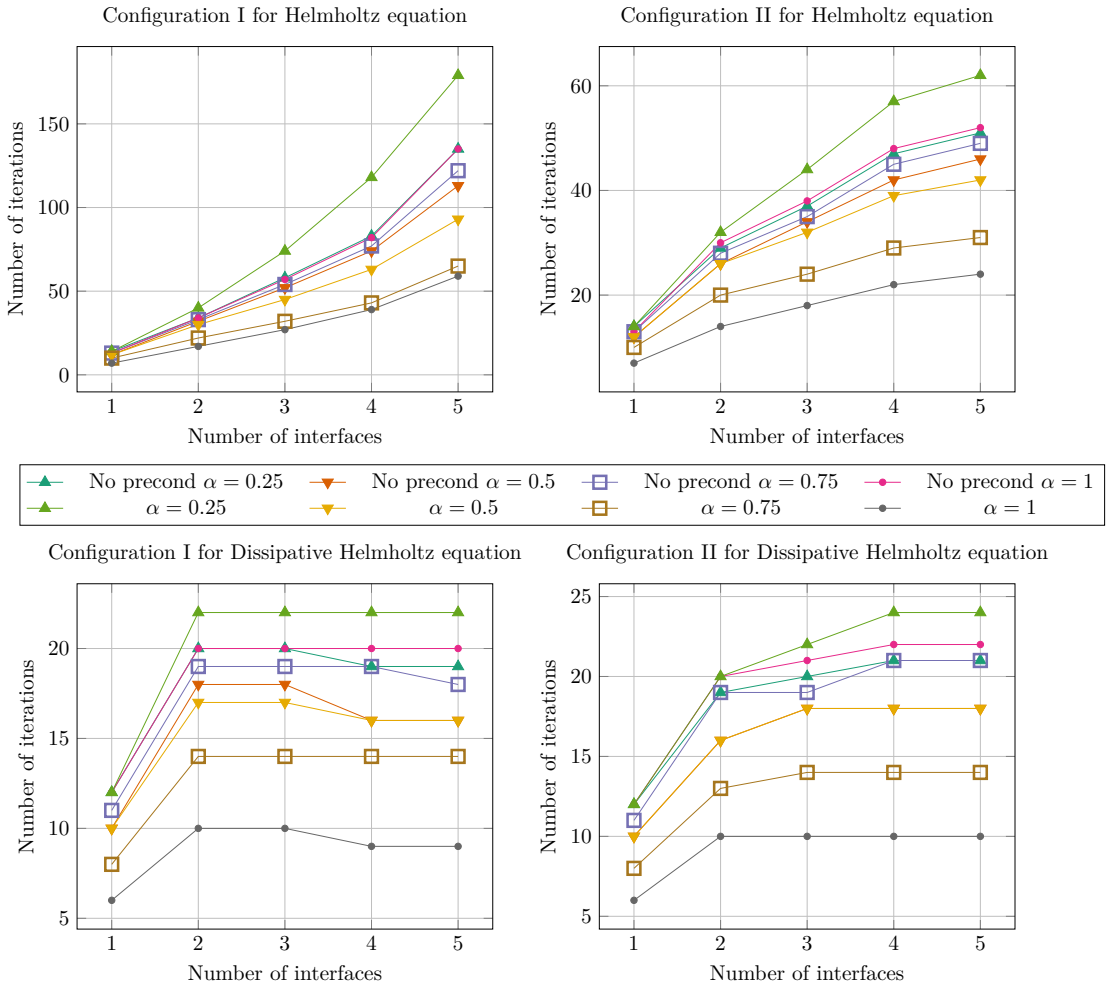


Figure 4.7 – Number of iterations for GMRes to solve the local multi-trace formulation using Equation (4.15) with $\mathbf{\Pi}^{-1}$ as a preconditioner, $\kappa_0 = 1.00002$, $\kappa_1 = 1.26308$, $\kappa_2 = 2.51121$, $\kappa_3 = 1.9173$, $\kappa_4 = 2.06553$ and $\kappa_5 = 1.43792$.

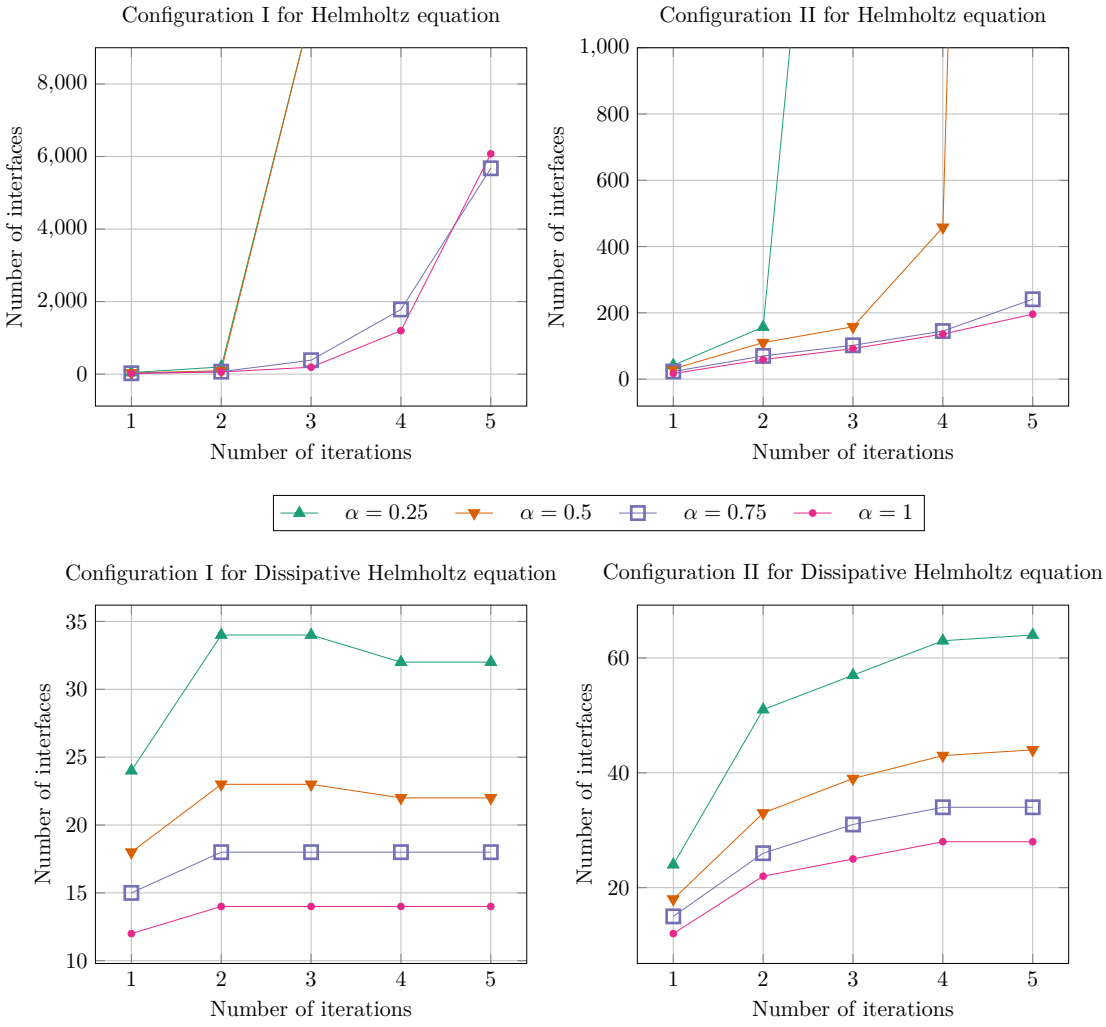


Figure 4.8 – Number of iterations for GMRes to solve the local multi-trace formulation using Equation (4.15) with $\mathbf{\Pi}^{-1}$ as a preconditioner, $\kappa_0 = 1.00031$, $\kappa_1 = 6.26151$, $\kappa_2 = 31.2242$, $\kappa_3 = 19.346$, $\kappa_4 = 22.3107$ and $\kappa_5 = 9.75837$.

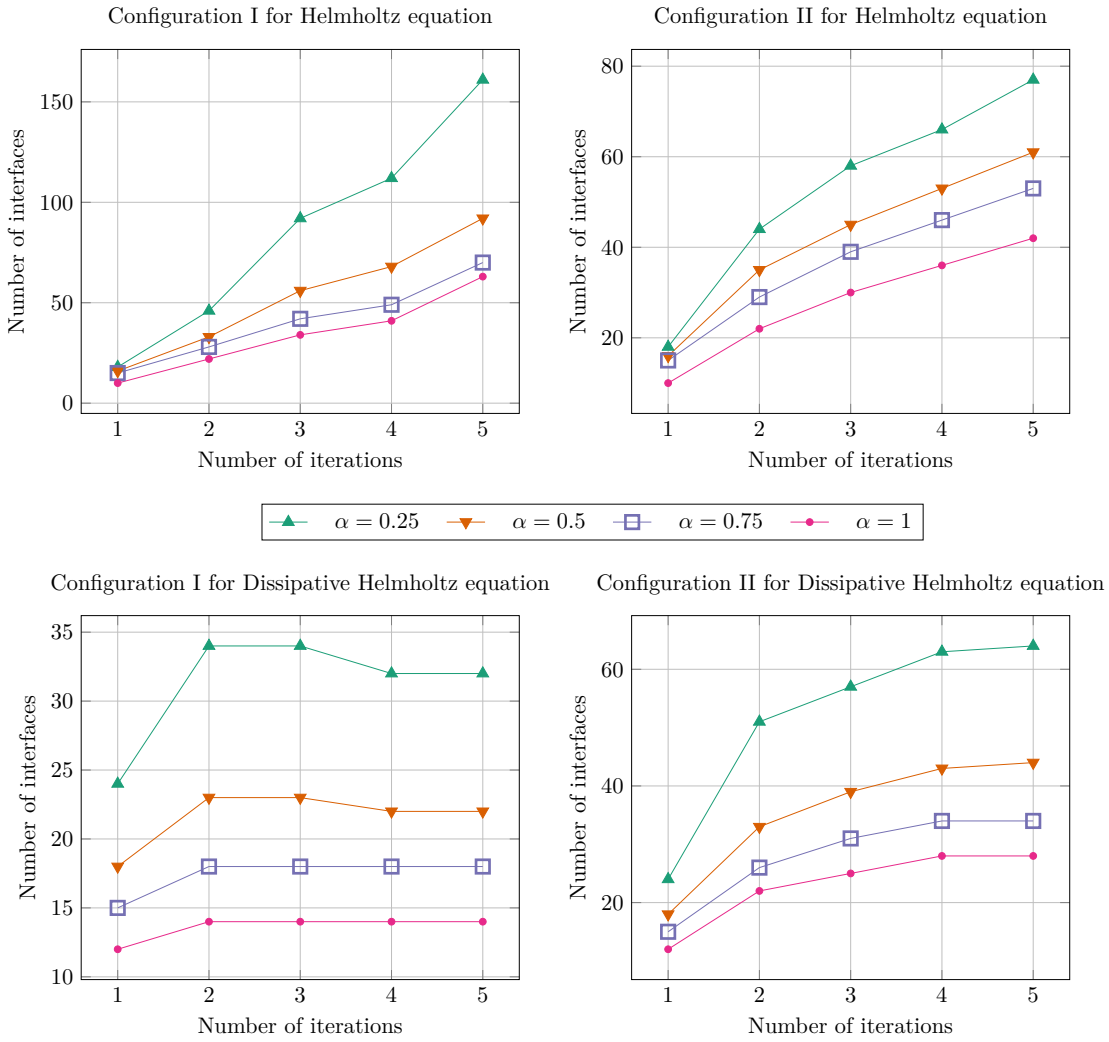


Figure 4.9 – Number of iterations for GMRes to solve the local multi-trace formulation using Equation (4.15) with $\mathbf{\Pi}^{-1}$ as a preconditioner, $\mu_0 = 1.00078$, $\mu_1 = 14.1538$, $\mu_2 = 76.5605$, $\mu_3 = 46.865$, $\mu_4 = 54.2767$ and $\mu_5 = 22.8959$.

CHAPTER 5

Preconditioners for the boundary element method

Outline of the current chapter

In Chapter 3, we presented two different points of view on Schwarz methods, either as iterative methods or as preconditioners. In [80], it was pointed out that in the case of Boundary Element Method (BEM), we could also think of two approaches for Domain Decomposition Method

(DDM). Indeed, we could make a decomposition of the original domain where the solution is sought, a *volume decomposition*, and then formulate a Boundary Integral Equation (BIE) in each subdomain with some ways of coupling them. Or we could first set up a BIE on the whole computational domain and then apply a domain decomposition of its boundary, a *surface decomposition*. In this chapter, we will explore the second approach to define Schwarz preconditioners and thus, we will adapt the second point of view on Schwarz method to BEM matrices.

It is quite counter intuitive to apply techniques based on locality as Additive Schwarz Method (ASM), to non-local problems as Boundary Integral Equations (BIE). But several works followed this idea and an intuition of why it works anyway would be that because of the singularity of the kernel of the BIE, the linear systems in the definite positive case are somehow "diagonal dominant". The first article seems to be [80] where a one-level strategy in two dimensions for the h-version applied to the weakly singular operator and the hypersingular operator with overlap has been studied. Then, the case of symmetric positive definite operators have been extended to two-level strategy, with and without overlap, for the p-version, h-version and hp-version in 2D [156, 155, 88, 93, 154] and 3D [86, 92, 87, 90, 84, 91, 157]. For an overview on the articles published before 1998, we refer to [147], we also refer to the habilitation thesis [89] for a summary of results on interpolation theory and fractional Sobolev spaces.

We seek to adapt the GenEO coarse space associated with the ASM as introduced in Section 3.3.3, to precondition BEM matrices described in Section 2.4.5. We saw in Section 3.3.1 that Theorem 3.9 allows us to study such preconditioners, and a key point to satisfy its hypothesis is how we can decompose a global energy. More precisely, to satisfy the second and third hypothesis of Theorem 3.9, we have to be able to respectively upper bound and lower bound the global energy by a sum of local energies. Studying BIE, a new obstacle appears compared to usual Partial Differential Equations (PDE) with the non-locality of the bilinear forms under study and of the norms of the underlying Sobolev spaces. Decomposing the global energy as we did with Lemma 3.13 for example with usual PDE, will be difficult because of this non-locality.

We found several possibilities using different approaches to decompose a global energy and to circumvent the non-locality of the problem. They all lead to a specific GenEO-type coarse space. It brought us to study numerically the efficiency of the coarse spaces we introduced.

In Section 5.1, we first adapt the discrete setting introduced in Section 3.2 to surface meshes. Then in Section 5.3, we present coarse spaces for BEM matrices inspired by the GenEO approach assuming we have a way to lower bound the global energy by a sum of local terms. In Section 5.2, we present several concrete possibilities to do so. Finally, in Section 5.4 we present several numerical studies of the introduced coarse spaces. This work also appears in [119] and has been submitted.

5.1 Finite element setting

As described in Section 3.2, we introduce a finite element setting with the same notation, the only difference being that we mesh Γ that we suppose polyhedral.

5.1.1 Meshes

Definition 5.1 (Surface mesh, [56, Definition 8])**.** *Let $\Gamma \subseteq \partial\Omega$ where Ω is a domain in $\mathbb{R}^d$. A mesh of Γ is a finite collection of disjoint non-degenerate simplices $\mathcal{T} = \{K_1, \ldots, K_{N_{\text{el}}}\}$ such that*

- $\overline{\Gamma} = \bigcup_{i=1}^{N_{\text{el}}} \overline{K}_i$,

- *the intersection* $\overline{K}_i \cap \overline{K}_j$ *is either empty, a common point or a common edge of both* K_i *and* K_j *for every* $1 \leq i, j \leq N_{\text{el}}$ *and* $i \neq j$.

Definition 5.2 (Meshsize, [53, Definition 1.13])**.** *For all* $K \in \mathcal{T}$, h_K *denotes the* diameter *of* K

$$\forall K \in \mathcal{T}, \quad h_K = \operatorname{diam} K = \max_{x,y \in K} |x - y|,$$

Then, we define the meshsize *of* $\mathcal{T}$ *as*

$$h := \max_{K \in \mathcal{T}} h_K.$$

We use the notation $\mathcal{T}_h$ for a mesh $\mathcal{T}$ of meshsize h.

Definition 5.3 (Shape regularity, [53, Definition 1.13])**.** *A family of meshes* $\{\mathcal{T}_h\}_{h>0}$ *is said to be* shape-regular *if there is* σ_0 *such that*

$$\forall h, \forall K \in \mathcal{T}_h, \quad \sigma_K = \frac{h_K}{\rho_K} \leq \sigma_0,$$

where ρ_K *is the radius of the largest ball that can be inscribed in* K.

Usually, the mesh $\mathcal{T}_h$ is generated using a *reference cell* K_{ref} and an affine geometric transformation mapping K_{ref} to the actual mesh cells. We denote this mapping F_K, so that $K = F_K(K_{\text{ref}})$.

5.1.2 Discrete spaces

In the following, we will denote $\mathcal{V}_h$ a generic space of finite element functions associated with $\mathcal{T}_h$, N its dimension and $(\varphi_j)_{j=1}^N$ the finite element basis of $\mathcal{V}_h$. But, we also introduce some more specific discrete spaces. First, we define the polynomial space on the reference element

$$\mathcal{P}^m(K_{\text{ref}}) := \operatorname{Span}\left\{ (x_i)_{i=1}^d \in \mathbb{R}^d \mapsto \prod_{i=1}^d x_i^{m_i} \,|\, 0 \leq \sum_{i=1}^d m_i \leq m \right\}.$$

Thus, polynomial spaces on a mesh $\mathcal{T}_h$ are defined by

$$\begin{aligned} \mathcal{P}^m(\mathcal{T}_h) &:= \{u \in L^\infty(\Omega) \,|\, \forall K \in \mathcal{T},\, u \circ F_K \in \mathcal{P}^m(K_{\text{ref}})\}, \\ \mathcal{S}^m(\mathcal{T}_h) &:= \mathcal{P}^m(\mathcal{T}_h) \cap C^0(\Gamma), \end{aligned}$$

where $C^0(\Gamma)$ is the space of continuous functions on Γ. We also define the discrete space with vanishing boundary conditions

$$\widetilde{\mathcal{S}}^m(\mathcal{T}_h) := \widetilde{H}^{1/2}(\Gamma) \cap \mathcal{S}^m(\mathcal{T}_h).$$

5.1.3 Galerkin approximation

As described in Section 2.4.5, the discretization of any BIE of the first kind derived in Sections 2.4.1 and 2.4.2 then reads: find $u_h \in \mathcal{V}_h \subset \widetilde{H}^s(\Gamma)$ such that

$$a(u_h, w_h) = b(w_h), \quad \forall w_h \in \mathcal{V}_h, \tag{5.1}$$

where $s = \pm\frac{1}{2}$, $a : \widetilde{H}^s(\Gamma) \times \widetilde{H}^s(\Gamma) \to \mathbb{R}$ and $b : \widetilde{H}^s(\Gamma) \to \mathbb{R}$. If $s = \frac{1}{2}$, it means that we are interested in Neumann problem (since we are looking for a Dirichlet trace) and we denote the bilinear form a_{W}. On the contrary, if $s = -\frac{1}{2}$, it means that we are interested in Dirichlet problem, and we denote the bilinear form a_{V}. In any case, we assume that a is symmetric positive definite, i.e.

$$C^- \|u\|^2_{\widetilde{H}^s(\Gamma)} \leqslant a(u,u) \leqslant C^+ \|u\|^2_{\widetilde{H}^s(\Gamma)}, \tag{5.2}$$

and more precisely, we denote the equivalence constants C^-_{W} and C^+_{W} for $s = \frac{1}{2}$ and C^-_{V} and C^+_{V} for $s = -\frac{1}{2}$.

We denote by $(\varphi_j)_{j=1}^N$ the finite element basis of $\mathcal{V}_h$ and $\dim(\mathcal{V}_h) = N$, then from (5.1), we obtain the following linear system

$$\mathbf{A}\mathbf{v}_h = \mathbf{b}_h,$$

where $(\mathbf{A})_{i,j} = a(\varphi_j, \varphi_i)$, $(\mathbf{b}_h)_j = \int_\Gamma g\varphi_j \,\mathrm{d}\sigma$ for $1 \leq i,j \leq N$ and $\mathbf{v}_h$ is the vector of coefficients corresponding to the unknown finite element function v_h. Since we assume that a is symmetric and positive definite, so is $\mathbf{A}$. The main difference with Chapter 3 is that $\mathbf{A}$ is dense because of the non-local nature of the underlying operators (see Section 2.4.5). Again, we will denote $\mathbf{A}_{\mathrm{W}}$ the matrix associated with a_{W} and $\mathbf{A}_{\mathrm{V}}$ the matrix associated with a_{V}.

5.1.4 Domain decomposition

We need to partition the domain Γ. Since we have in mind the implementation of $\mathcal{H}$-matrices, we consider the partition techniques used to define the hierarchy of this compression method. These techniques are generally based on a geometric clustering of the degrees of freedom, which is why we consider a partition of the degrees of freedom into n clusters, that induces a partition of the global numbering $\{1,\dots,N\} = \cup_{p=1}^n \mathrm{dof}'_{h,p}$ with $\mathrm{dof}'_{h,p} \cap \mathrm{dof}'_{h,l} = \emptyset$ for every $p,l \in \{1,\dots,n\}$ and $p \neq l$. This partition of the degrees of freedom induces a geometric partition.

This differs from what is usually done in a finite element context, where automatic partitioners such as METIS [105] or SCOTCH [25] are used to decompose the domain, and they are generally used to partition the elements of the mesh. By contrast, here we take account of a compression method working on degrees of freedom. Then, we can add several layers of mesh elements and their associated degrees of freedom to increase the overlap between neighboring subdomains. We denote the indices of the resulting pth subdomain $\mathrm{dof}_{h,p}$ so that $\mathrm{dof}'_{h,p} \subset \mathrm{dof}_{h,p}$ and $\{1,\dots,N\} \subset \cup_{p=1}^n \mathrm{dof}_{h,p}$. Its associated local finite element space is

$$\mathcal{V}_{h,p} := \mathrm{Span}(\varphi_j|_{\widetilde{\Gamma}_p} \,|\, j \in \mathrm{dof}_{h,p}),$$

where $\overline{\widetilde{\Gamma}}_p := \cup_{j\in\mathrm{dof}_{h,p}} \mathrm{supp}(\varphi_{\mathrm{j}})$, and thus $\overline{\Gamma} \subset \cup_{p=1}^n \overline{\widetilde{\Gamma}}_p$. Later, we will also need to use $\Gamma_p := \widetilde{\Gamma}_p \setminus \cup_{j\notin\mathrm{dof}_{h,p}} \mathrm{supp}(\varphi_{\mathrm{j}})$, see the remark below for an example. We also define an arbitrary local numbering of the degrees of freedom with the bijection $\sigma_p : \{1,2,\dots,N_p\} \to \mathrm{dof}_{h,p}$, where $N_p := \dim(\mathcal{V}_{h,p})$. Similarly to Section 3.2.3, we introduce the operator of extension by zero R_p^T associated with $\mathcal{V}_{h,p}$, and its matrix representation $\mathbf{R}_p^T$, we also denote the restriction operator R_p and its matrix representation $\mathbf{R}_p$ so that $\mathbf{R}_p\mathbf{R}_p^T = \mathbf{I}_d \in \mathbb{R}^{N_p\times N_p}$ and $\mathrm{R}_p\,\mathrm{R}_p^T = \mathrm{I_d}$. We can do the same remark as in Section 3.2.3, R_p is *not* a restriction in the natural sense. It restricts the set of shape functions used to represent an element of the discrete space $\mathcal{V}_{h,p}$ (see remark thereafter). Besides R_p^T is not the dual of R_p for the L^2 scalar product, they are respectively the

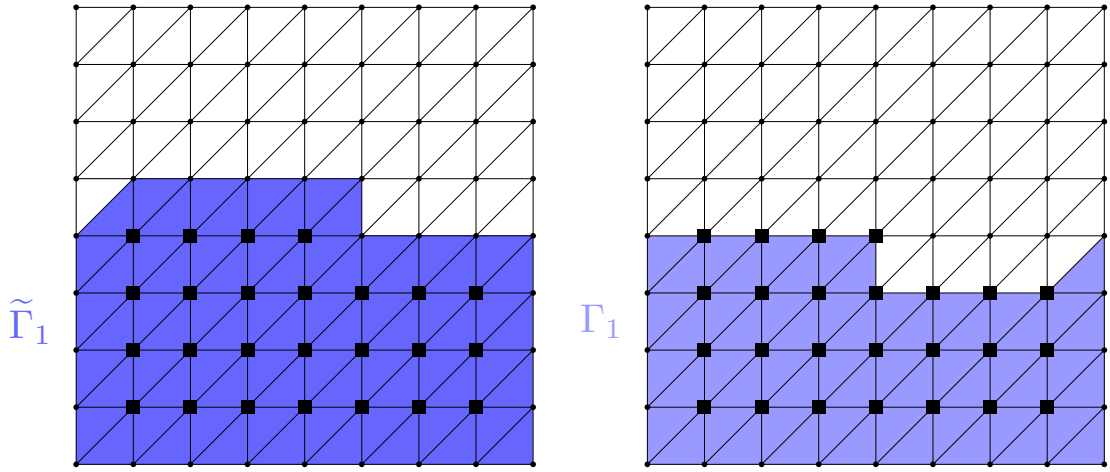


Figure 5.1 – Example of a subdomain in 2D for $\mathbb{P}_1$ finite elements where the rectangle nodes denote $\mathbb{P}_1$ functions in $\mathcal{V}_{h,1}$.

equivalent of $\mathbf{R}_p^T$ and $\mathbf{R}_p$ for finite element functions, and $\mathbf{R}_p^T$ is the transpose of $\mathbf{R}_p$.

Remark 5.4. *Figure 5.1 adapts the example given in Figure 3.4 with $\mathbb{P}_1$ shape functions to a screen problem in 3D. Let $\mathcal{V}_h \subset \widetilde{H}^{1/2}(\Gamma)$ so that finite element functions associated with nodes on the boundary are not in $\mathcal{V}_h$.*

We also introduce again the linear applications P_h, $P_{h,p}$ and their inverse to allow going back and forth the finite element and algebraic representation of functions. The relations between all these operators are again summarized in Figure 3.5. We define a partition of unity $\{\mathbf{D}_p\}_{p=0}^n \in (\mathbb{R}^{N_p \times N_p})^n$ as in Definition 3.6 such that $\mathbf{I}_d = \sum_{p=1}^n \mathbf{R}_p^T \mathbf{D}_p \mathbf{R}_p$. Similar constants to the ones defined in Definition 3.7 will be used in the analysis thereafter:

Definition 5.5.

- *k_0 denotes the maximum multiplicity of the interaction between subdomains plus one, i.e.,*

$$k_0 := \max_{1 \leq j \leq N} \# \left\{ p \,|\, \mathbf{R}_p \mathbf{M} \mathbf{R}_j^T \neq 0 \right\},$$

  where $(\mathbf{M})_{i,j} = \int_\Gamma \varphi_i \varphi_j \, \mathrm{d}\gamma$ is the mass matrix.

- *k_1 denotes the maximal multiplicity of the subdomain intersection, i.e., the largest integer m such that there exist m different subdomains whose intersection has a nonzero measure.*

$$k_1 := \operatorname{ess\,sup} \left\{ \sum_{p=1}^n 1_{\Gamma_p} \right\}$$

- *N_c is the minimum number of colors we can use to color a decomposition such that any two domains sharing an overlap have different colors*

And of course, the same remarks as in Section 3.2.3 can be done concerning the graph point of view on these constants. Finally, the definition of the preconditioner without coarse space is

again

$$\mathbf{M}_{\mathrm{ASM}}^{-1} := \sum_{p=1}^{n} \mathbf{R}_p^T \mathbf{A}_{p,p}^{-1} \mathbf{R}_p, \quad \text{with } \mathbf{A}_{l,p} = \mathbf{R}_l^T \mathbf{A} \mathbf{R}_p, \text{ and } 1 \leq l, p \leq n.$$

5.2 Inequalities for splitting norms

As we have seen in Section 3.3.1, the hypotheses of Theorem 3.9 are related to how we can decompose the global energy norm of the problem. Working directly on the analytical expression of the considered bilinear forms a is usually really difficult, so we will focus on decomposing fractional Sobolev norms. Thus, decomposing the global energy given by the bilinear form a will be given by the equivalence relation between a and fractional Sobolev norms.

Inequalities between the global norm and the sum of the local norms are essential in the analysis of the efficiency of domain decomposition methods and they are not trivial because the norms are non-local. The constants appearing in these inequalities are important because they will appear in the final estimate for the condition number.

5.2.1 Localization by local fractional order norms

In this section, we will show how to directly localize a global norm in $\widetilde{H}^{1/2}(\Gamma)$. We have the following lemma

Lemma 5.6 ([139, Lemma 4.1.49 (b)])**.** *For $(u_p)_{1\leqslant p\leqslant n} \in \prod_{p=1}^{n} \widetilde{H}^{1/2}(\Gamma'_p)$ with $(\Gamma'_p)_{p=1}^{n}$ a non-overlapping partition of Γ, i.e. $\overline{\Gamma} = \cup_{p=1}^{n} \overline{\Gamma'}_p$ and $\Gamma'_p \cap \Gamma'_l = \emptyset$ for $l \neq p$ and $1 \leqslant l, p \leqslant n$, we have the following inequality:*

$$\left\| \sum_{p=1}^{n} \mathrm{E}_{\Gamma'_p}(u_p) \right\|^2_{\widetilde{H}^{1/2}(\Gamma)} \leqslant \frac{5}{2} \sum_{p=1}^{n} \|u_p\|^2_{\widetilde{H}^{1/2}(\Gamma'_p)},$$

Then, we can obtain the following lemma using a *coloring argument*:

Lemma 5.7. *For $(u_p)_{1\leqslant p\leqslant n} \in \prod_{p=1}^{n} \widetilde{H}^{1/2}(\widetilde{\Gamma}_p)$, we have the following inequality:*

$$\left\| \sum_{p=1}^{n} \mathrm{E}_{\widetilde{\Gamma}_p}(u_p) \right\|^2_{\widetilde{H}^{1/2}(\Gamma)} \leqslant \frac{5}{2} N_c \sum_{p=1}^{n} \|u_p\|^2_{\widetilde{H}^{1/2}(\widetilde{\Gamma}_p)},$$

where N_c is defined in Definition 5.5.

Proof. If one colors each subdomain $\widetilde{\Gamma}_p$ such that two subdomains with the same color cannot be neighbors, that is to say, cannot overlap, then we can define $(\Gamma''_l)_{l=1}^{N_c}$ where each Γ''_l is the union of every subdomain with the same color, that we number l, and N_c is the number of colors. Let us define

$$w_l = \sum_{p | \widetilde{\Gamma}_p \subset \Gamma''_l} \mathrm{E}_{\widetilde{\Gamma}_p}(u_p) \in \widetilde{H}^{1/2}(\Gamma''_l).$$

Then, we have

$$\left\|\sum_{p=1}^{n} \mathrm{E}_{\widetilde{\Gamma}_p}(u_p)\right\|^2_{\widetilde{H}^{1/2}(\Gamma)} = \left\|\sum_{l=1}^{N_c} \sum_{p|\widetilde{\Gamma}_p \subset \Gamma_l''} \mathrm{E}_{\widetilde{\Gamma}_p}(u_p)\right\|^2_{\widetilde{H}^{1/2}(\Gamma)} = \left\|\sum_{l=1}^{N_c} w_l\right\|^2_{\widetilde{H}^{1/2}(\Gamma)} \leqslant N_c \sum_{l=1}^{N_c} \|w_l\|^2_{\widetilde{H}^{1/2}(\Gamma)}.$$

By definition of Γ_l'', we have that all subdomains $\widetilde{\Gamma}_p$ such that $\widetilde{\Gamma}_p \subset \Gamma_l''$ are disjoint so that we can use Lemma 5.6 in the last expression to obtain

$$\begin{aligned} N_c \sum_{l=1}^{N_c} \|w_l\|^2_{\widetilde{H}^{1/2}(\Gamma)} &\leqslant N_c \sum_{l=1}^{N_c} \left\| \sum_{p|\widetilde{\Gamma}_p \subset \Gamma_l''} \mathrm{E}_{\widetilde{\Gamma}_p}(u_p)\right\|^2_{\widetilde{H}^{1/2}(\Gamma)} \\ &\leqslant N_c \frac{5}{2} \sum_{l=1}^{N_c} \sum_{p|\widetilde{\Gamma}_p \subset \Gamma_l''} \|u_p\|^2_{\widetilde{H}^{1/2}(\widetilde{\Gamma}_p)} = N_c \frac{5}{2} \sum_{p=1}^{n} \|u_p\|^2_{\widetilde{H}^{1/2}(\widetilde{\Gamma}_p)}. \end{aligned}$$

□

The previous lemma gave a direct result about how we can upper bound a global energy in $\|\cdot\|^2_{\widetilde{H}^{1/2}(\Gamma)}$ by a sum of local norms. The reverse inequality is false as proved by a counter example in [2, Appendix], but we can lower bound the global energy by a sum of local $H^{1/2}$ norms

Lemma 5.8. *For $u \in H^{1/2}(\Gamma)$, there holds*

$$\sum_{p=1}^{n} \|u|_{\Gamma_p}\|^2_{H^{1/2}(\Gamma_p)} \leqslant k_1 \|u\|^2_{\widetilde{H}^{1/2}(\Gamma)},$$

where k_1 is defined in Definition 5.5.

Proof. The proof is in two parts. First, we bound the $\widetilde{H}^{1/2}$-norm from below by the $H^{1/2}$-norm, and then we split the $H^{1/2}$-norm. We denote

$$\int_\Gamma \int_{\Gamma'} [u] := \int_\Gamma \int_{\Gamma'} \frac{|u(\mathbf{x}) - u(\mathbf{y})|^2}{|\mathbf{x} - \mathbf{y}|^{d+1}} \, \mathrm{d}\gamma(\mathbf{x}, \mathbf{y}),$$

where we borrowed the notation from [139, Lemma 4.1.49 (b)].

- *First part:* We have $\Gamma \subset \widetilde{\Gamma} = \partial\Omega$ with $\Omega \subset \mathbb{R}^d$, a lipschitz polyhedron. Then, for $u \in \widetilde{H}^{1/2}(\Gamma)$, we have by definition

$$\begin{aligned} \|u\|^2_{\widetilde{H}^{1/2}(\Gamma)} &= \|\mathrm{E}_\Gamma(u)\|^2_{H^{1/2}(\widetilde{\Gamma})} = \int_\Gamma |u|^2 + \int_{\widetilde{\Gamma}} \int_{\widetilde{\Gamma}} [\mathrm{E}_\Gamma(u)] \\ &= \underbrace{\|u\|^2_{L^2(\Gamma)} + \int_\Gamma \int_\Gamma [u]}_{=\|u\|^2_{H^{1/2}(\Gamma)}} + \underbrace{2 \int_{\widetilde{\Gamma}\setminus\Gamma} \int_\Gamma [u] + \int_{\widetilde{\Gamma}\setminus\Gamma} \int_{\widetilde{\Gamma}\setminus\Gamma} [u]}_{\geq 0} \\ &\geq \|u\|^2_{H^{1/2}(\Gamma)}. \end{aligned}$$

- *Second part:* According to the definition of the norm, we have

$$\|u\|^2_{H^{1/2}(\Gamma)} = \|u\|^2_{L^2(\Gamma)} + \int_\Gamma \int_\Gamma [u].$$

Following Lemma 3.22, we have $\sum_{p=1}^n \|u|_{\Gamma_p}\|^2_{L^2(\Gamma_p)} \leqslant k_1 \|u\|^2_{L^2(\Gamma)}$. Thus, it remains to prove $\sum_{p=1}^n \int_{\Gamma_p} \int_{\Gamma_p} [u] \leqslant k_1 \int_\Gamma \int_\Gamma [u]$.

Let $1 \leq l \leq n$, we consider all the intersections of l subdomains among the n subdomains. Note that there can be several areas shared by possibly different sets of l subdomains, see Figure 5.2 for an example. That is why, we define an arbitrary numbering $\{1, \ldots, n_l\}$ of all the set of l subdomains that intersect each other, except for $l = 1$ where the numbering corresponds to the numbering of the subdomains.

Let us denote $\theta_{l,i} \subset \Gamma$ the ith set of l subdomains. In particular, we have $\theta_{1,p} := \Gamma_p \setminus \overline{(\cup_{m=1,m\neq p}^n \Gamma_m)}$, i.e. $\theta_{1,p}$ corresponds to the pth subdomain without its overlap. We denote

$$\Theta := \{(l,i) \in \mathbb{N}^2 \,|\, \theta_{l,i} \neq \emptyset\},$$

and remark that $\{\theta_{l,i}\}_{l,i}$ defines a partition of Γ, so that

$$\int_\Gamma \int_\Gamma [u] = \sum_{(l,i)\in\Theta} \sum_{(m,j)\in\Theta} \int_{\theta_{l,i}} \int_{\theta_{m,j}} [u]. \tag{5.3}$$

For every subdomain Γ_p, we also define

$$\Theta_p := \{(l,i) \in \mathbb{N}^2 \,|\, \theta_{l,i} \cap \Gamma_p \neq \emptyset\},$$

so that $\Gamma_p = \cup_{(l,i)\in\Theta_p} \theta_{l,i}$ and

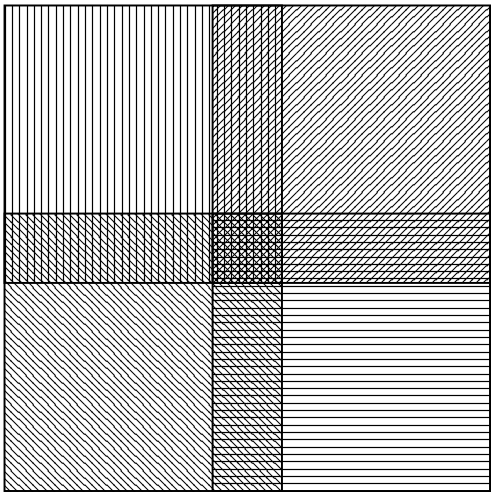
$$\begin{aligned}
\sum_{p=1}^n \int_{\Gamma_p} \int_{\Gamma_p} [u] &= \sum_{p=1}^n \sum_{(l_p,i_p)\in\Theta_p} \sum_{(m_p,j_p)\in\Theta_p} \int_{\theta_{l_p,i_p}} \int_{\theta_{m_p,j_p}} [u] \\
&= \sum_{p=1}^n \sum_{(l,i)\in\Theta} \sum_{(m,j)\in\Theta} \mathbb{1}_{(l,i)\in\Theta_p} \mathbb{1}_{(m,j)\in\Theta_p} \int_{\theta_{l,i}} \int_{\theta_{m,j}} [u] \\
&= \sum_{(l,i)\in\Theta} \sum_{(m,j)\in\Theta} \sum_{p=1}^n \mathbb{1}_{(l,i)\in\Theta_p} \mathbb{1}_{(m,j)\in\Theta_p} \int_{\theta_{l,i}} \int_{\theta_{m,j}} [u].
\end{aligned} \tag{5.4}$$

where $\mathbb{1}_{(l,i)\in\Theta_p} = 1$ if $(l,i) \in \Theta_p$ and 0 otherwise. And, by definition, a given pair $(l,i) \in \Theta$ cannot be in more that k_1 sets Θ_p, otherwise it would mean that $\theta_{l,i}$ is shared by more than k_1 subdomains. Thus,

$$\sum_{p=1}^n \mathbb{1}_{(l,i)\in\Theta_p} \mathbb{1}_{(m,j)\in\Theta_p} \leqslant k_1.$$

Finally, we obtain the expected result using Equations (5.3) and (5.4) associated with the previous inequality.

□



$\theta_{1,3}$	$\theta_{2,3}$	$\theta_{1,4}$
$\theta_{2,4}$	$\theta_{4,1}$	$\theta_{2,2}$
$\theta_{1,1}$	$\theta_{2,1}$	$\theta_{1,2}$

Figure 5.2 – Example of a partition defined as in the proof of Lemma 5.8.

5.2.2 Other localizations

The preceding inequalities are the ones that mimic the best what is done for PDEs: Lemma 5.7 is similar to Lemma 3.13 (remember that N_c is upper bounded by k_0) and Lemma 5.8 is similar to Lemma 3.22. But other approaches can be followed and they will lead to other coarse spaces.

The simplest is based on the continuous injection between $\widetilde{H}^{1/2}(\Gamma)$ and $L^2(\Gamma)$

Lemma 5.9. *Let $u \in \widetilde{H}^{1/2}(\Gamma)$, we have*

$$C_{\text{inj}} \sum_{p=1}^{n} \|u\|^2_{L^2(\Gamma_p)} \leqslant k_1 \|u\|^2_{\widetilde{H}^{1/2}(\Gamma)},$$

where k_1 is defined in Definition 5.5 and C_{inj} is the continuity constant of the injection of $\widetilde{H}^{1/2}(\Gamma)$ in $L^2(\Gamma)$.

Proof. Using the fact we have $\widetilde{H}^{1/2}(\Gamma)$ is compactly embedded in $L^2(\Gamma)$, we obtain

$$\|u\|^2_{\widetilde{H}^{1/2}(\Gamma)} \geq C_{\text{inj}} \|u\|^2_{L^2(\Gamma)}.$$

Then, the L^2 norm is local, so that we can decompose it like in Lemma 3.22. □

Another approach is to take techniques and inequalities used in adaptive BEM (see [56]). Let us denote $u \in \widetilde{H}^s$ the solution of the underlying PDE and $u_h \in \mathcal{V}_h$ its approximation, solution of Equation (5.1) associated with a mesh $\mathcal{T}_h$, the goal of adaptive BEM (and more generally adaptive FEM) is to estimate the error $u - u_h$ with a global computable quantity that serves as a stopping criterion and that can be decomposed in local contributions to also estimate the local error. Thus, it shows where to refine the mesh locally, until the stopping criterion is reached.

Usually, a posteriori estimators have the following form

$$\eta_{\mathcal{T}_h}^2 = \sum_{K \in \mathcal{T}_h} \eta_K^2.$$

Such an estimator is called *reliable* if $\|u_h - u\| \lesssim \eta_{\mathcal{T}_h}$, and it is also called *efficient* if $\eta_{\mathcal{T}_h} \lesssim \|u_h - u\|$, $\|\cdot\|$ is typically the energy norm of the considered problem. Since u is unknown and the estimator needs to be computable, a natural approach is to use *residual type estimators* where the unknown error is estimated by the residual error. But this residual error is usually measured in fractional Sobolev norms, thus localization techniques are needed and that is why it was interesting in our case to look at the methods at play in adaptive BEM. In particular, we have been interested in [8, Corollary 6.], that is used in the following lemma.

Lemma 5.10. *Let $u_h \in \widetilde{\mathcal{S}}^l(\mathcal{T}_h)$, we have*

$$C_{\text{ad}} \sum_{p=1}^{n} \|h_{\mathcal{T}}^{1/2} \nabla_\Gamma u_h|_{\Gamma_p}\|_{L^2(\Gamma_p)}^2 \leqslant k_1 \|u_h\|_{\widetilde{H}^{1/2}(\Gamma)}^2,$$

where C_{ad} depends only on Γ, the shape-regularity of $\mathcal{T}_h$ and l, while $h_{\mathcal{T}}|_K = h_K$ for every $K \in \mathcal{T}_h$.

Proof. According to [7, Corollary 6.], we have

$$\|u_h\|_{\widetilde{H}^{1/2}(\Gamma)}^2 \geq C_{\text{ad}} \|h_{\mathcal{T}}^{1/2} \nabla_\Gamma u_h\|_{L^2(\Gamma)}^2.$$

Then, the L^2 norm is local, so that we can decompose it like in Lemma 3.22. □

This result is actually an inverse inequality for fractional Sobolev norms and it is obtained using interpolation theory.

5.3 Two-level preconditioning for the hypersingular operator

In this section, we use the strategy of GenEO [45] as in Section 3.3.3 to precondition the matrix $\mathbf{A}_{\text{W}}$ defined in Section 5.1.3 with the hypersingular singular operator W. We define a generic coarse space $\mathcal{V}_{h,0}$, and we denote its size by N_0. The coarse space is spanned by the columns of a rectangular matrix $\mathbf{R}_0^T$ of size $N \times N_0$ and the preconditioner is defined as follows

$$\mathbf{M}_{\text{ASM,W}}^{-1} := \mathbf{R}_0^T (\mathbf{R}_0 \mathbf{A}_{\text{W}} \mathbf{R}_0^T)^{-1} \mathbf{R}_0 + \sum_{p=1}^{n} \mathbf{R}_p^T (\mathbf{R}_p \mathbf{A}_{\text{W}} \mathbf{R}_p^T)^{-1} \mathbf{R}_p. \tag{5.5}$$

5.3.1 Fictitious space lemma

We formulate the ASM with GenEO coarse space as follows:

Definition 5.11.

- $\mathcal{V}_h \subset \widetilde{H}^{1/2}(\Gamma)$, *for example* $\mathcal{V}_h = \widetilde{\mathcal{S}}^l(\mathcal{T}_h)$.

- *$H := \mathbb{R}^N$ is endowed with the standard euclidean scalar product $(\cdot,\cdot)$, and we consider the following bilinear form:*

$$\begin{aligned}\widetilde{a} : \mathbb{R}^N \times \mathbb{R}^N &\to \mathbb{R}\\ (\mathbf{u}_h, \mathbf{v}_h) &\mapsto (\mathbf{A}_{\mathrm{W}}\mathbf{u}_h, \mathbf{v}_h) = a_{\mathrm{W}}(u_h, v_h),\end{aligned}$$

with $u_h = P_h\mathbf{u}_h$ and $v_h = P_h\mathbf{v}_h$.

- *$H_D := \prod_{p=0}^n \mathbb{R}^{N_p}$ is endowed with the standard euclidean scalar product, and we consider the following bilinear form:*

$$\begin{aligned}\widetilde{b} : \prod_{p=0}^n \mathbb{R}^{N_p} \times \prod_{p=0}^n \mathbb{R}^{N_p} &\to \mathbb{R}\\ (\mathbf{u}_h^p, \mathbf{v}_h^p)_{p=0}^n \mapsto \sum_{p=0}^n (\mathbf{A}_{\mathrm{W}}\mathbf{R}_p^T\mathbf{u}_h^p, \mathbf{R}_p^T\mathbf{v}_h^p) &= (\mathbf{A}_{\mathrm{W}}\mathbf{R}_0^T\mathbf{u}_h^0, \mathbf{R}_0^T\mathbf{v}_h^0)\\ &\quad + \sum_{p=1}^n a_{\mathrm{W}}(R_p^T u_h^p, R_i^T v_h^p),\end{aligned}$$

with $u_h^p = P_{h,p}\mathbf{u}_h^p$ and $v_h^p = P_{h,p}\mathbf{v}_h^p$ for $1 \le p \le n$. Then, $B : H_D \to H_D$ is defined by

$$\forall\, \mathbf{U}_h = (\mathbf{u}_h^p)_{p=0}^n \in \prod_{p=0}^n \mathbb{R}^{N_p}, \quad B(\mathbf{U}_h) := (\mathbf{R}_p\mathbf{A}_{\mathrm{W}}\mathbf{R}_p^T\mathbf{u}_h^p)_{0\le p\le n}.$$

Notice that B is a block diagonal operator, so that

$$\forall\, \mathbf{U}_h = (\mathbf{u}_h^p)_{p=0}^n \in \prod_{p=0}^n \mathbb{R}^{N_p}, \quad B^{-1}(\mathbf{U}_h) := ((\mathbf{R}_p\mathbf{A}_{\mathrm{W}}\mathbf{R}_p^T)^{-1}\mathbf{u}_h^p)_{0\le p\le n}.$$

- *The linear operator $\mathcal{R}_{\mathrm{ASM},2}$ is defined as follows*

$$\begin{aligned}\mathcal{R}_{\mathrm{ASM},2} : H_D &\to H\\ (\mathbf{u}_h^p)_{0\le p\le n} &\mapsto \sum_{p=0}^n \mathbf{R}_p^T\mathbf{u}_h^p.\end{aligned}$$

Before we introduce our two-level preconditioners, we formulate some general remarks. The first hypothesis of Theorem 3.9 is satisfied by definition of $\mathcal{R}_{\mathrm{ASM},2}$, identically to what is done in Lemma 3.19. The second hypothesis of Theorem 3.9 corresponds to the continuity of $\mathcal{R}_{\mathrm{ASM},2}$ and the third one is usually referred to as *stable decomposition*. Using the notations from Definition 5.11 of this theorem, these last two hypotheses can be rewritten as follows

(i) There exists a constant $c_R > 0$ such that, for all $(\mathbf{u}_h^p)_{p=0}^n \in H_D$

$$a_{\mathrm{W}}\left(\sum_{p=0}^n P_h\mathbf{R}_p^T\mathbf{u}_h^p, \sum_{p=0}^n P_h\mathbf{R}_p^T\mathbf{u}_h^p\right) \leqslant c_R \sum_{p=0}^n a_{\mathrm{W}}(P_h\mathbf{R}_p^T\mathbf{u}_h^p, P_h\mathbf{R}_p^T\mathbf{u}_h^p).$$

(ii) There exists a constant $c_T > 0$ such that, for all $\mathbf{u}_h \in \mathbb{R}^N$, there exists $\mathbf{U}_h = (\mathbf{u}_h^p)_{p=0}^n \in H_D$

with $\mathbf{u}_h = \mathcal{R}_{\mathrm{ASM},2}(\mathbf{U}_h)$, and

$$c_T \sum_{p=0}^{n} a_{\mathrm{W}}(P_h \mathbf{R}_p^T \mathbf{u}_h^p, P_h \mathbf{R}_p^T \mathbf{u}_h^p) \leqslant a_{\mathrm{W}}(P_h \mathbf{u}_h, P_h \mathbf{u}_h).$$

We deduce that these two hypotheses are related to how we can localize the bilinear form a_{W}, and since it is symmetric positive definite, it is also related to how we can localize the $\widetilde{H}^{1/2}$-norm.

5.3.2 Continuity of $\mathcal{R}_{\mathrm{ASM},2}$

The first hypothesis of Theorem 3.9 is actually satisfied without any precise definition of the coarse space. We have Lemma 5.7 to localize the $\widetilde{H}^{1/2}$-norm, which yields an upper bound for the global energy on $\widetilde{H}^{1/2}(\Gamma)$ by a sum of local energies, so that we obtain the first hypothesis required by Theorem 3.9.

Lemma 5.12. *Using the notations of Theorem 3.9, let* $\mathbf{U}_h = (\mathbf{u}_h^p)_{p=0}^n \in H_D$*, we have*

$$(\mathbf{A}_{\mathrm{W}} \mathcal{R}_{\mathrm{ASM}}(\mathbf{U}_h), \mathcal{R}_{\mathrm{ASM}}(\mathbf{U}_h)) \leqslant 2 \max\left(1, \frac{5}{2} N_c \frac{C_{\mathrm{W}}^+}{C_{\mathrm{W}}^-}\right) (B\mathbf{U}_h, \mathbf{U}_h)_D.$$

Proof. Using the fact that $\mathbf{A}_{\mathrm{W}}$ is symmetric positive definite, we have

$$\begin{aligned}(\mathbf{A}_{\mathrm{W}} \mathcal{R}_{\mathrm{ASM}}(\mathbf{U}_h), \mathcal{R}_{\mathrm{ASM}}(\mathbf{U}_h)) &= \left(\mathbf{A}_{\mathrm{W}} \sum_{p=0}^{n} \mathbf{R}_p^T \mathbf{u}_h^p, \sum_{p=0}^{n} \mathbf{R}_p^T \mathbf{u}_h^p\right) \\ &\leqslant 2 \left(\left(\mathbf{A}_{\mathrm{W}} \mathbf{R}_0^T \mathbf{u}_h^0, \mathbf{R}_0^T \mathbf{u}_h^0\right) + \left(\mathbf{A}_{\mathrm{W}} \sum_{p=1}^{n} \mathbf{R}_p^T \mathbf{u}_h^p, \sum_{p=1}^{n} \mathbf{R}_p^T \mathbf{u}_h^p\right) \right).\end{aligned}$$

The last term on the right-hand side can be rewritten

$$\begin{aligned}\left(\mathbf{A}_{\mathrm{W}} \sum_{p=1}^{n} \mathbf{R}_p^T \mathbf{u}_h^p, \sum_{p=1}^{n} \mathbf{R}_p^T \mathbf{u}_h^p\right) &= a_{\mathrm{W}} \left(\sum_{p=1}^{n} P_h \mathbf{R}_p^T \mathbf{u}_h^p, \sum_{p=1}^{n} P_h \mathbf{R}_p^T \mathbf{u}_h^p\right) \\ &= a_{\mathrm{W}} \left(\sum_{p=1}^{n} R_p^T u_h^p, \sum_{p=1}^{n} R_p^T u_h^p\right),\end{aligned}$$

with $u_h^p = P_{h,p} \mathbf{u}_h^p \in \mathcal{V}_{h,p} \subset \widetilde{H}^{1/2}(\widetilde{\Gamma}_p)$ (see diagram Figure 3.5). Finally, using the equivalence relation from Equation (5.2), Lemma 5.7 and the definition of the $\widetilde{H}^{1/2}$-norm, we have

$$\begin{aligned}a_{\mathrm{W}} \left(\sum_{p=1}^{n} R_p^T u_h^p, \sum_{p=1}^{n} R_p^T u_h^p\right) &\leqslant C_{\mathrm{W}}^+ \left\| \sum_{p=1}^{n} R_p^T u_h^p \right\|_{\widetilde{H}^{1/2}(\Gamma)}^2 \leqslant \frac{5}{2} N_c C_{\mathrm{W}}^+ \sum_{p=1}^{n} \|u_h^p\|_{\widetilde{H}^{1/2}(\widetilde{\Gamma}_p)}^2 \\ &\leqslant \frac{5}{2} N_c C_{\mathrm{W}}^+ \sum_{p=1}^{n} \left\| R_p^T u_h^p \right\|_{\widetilde{H}^{1/2}(\Gamma)}^2 \leqslant \frac{5}{2} N_c \frac{C_{\mathrm{W}}^+}{C_{\mathrm{W}}^-} \sum_{p=1}^{n} a_{\mathrm{W}}(R_p^T u_h^p, R_p^T u_h^p) \\ &\leqslant \frac{5}{2} N_c \frac{C_{\mathrm{W}}^+}{C_{\mathrm{W}}^-} \sum_{p=1}^{n} (\mathbf{A}_{\mathrm{W}} \mathbf{R}_p^T \mathbf{u}_h^p, \mathbf{R}_p^T \mathbf{u}_h^p).\end{aligned}$$

□

Remark 5.13 (Norm definitions)**.** *We would like to point out that, although one can obtain similar results to Lemma 5.7 with other definitions for the fractional Sobolev norms (with interpolation for example), we use in the last lemma the fact that* $\|u_h^p\|_{\widetilde{H}^{1/2}(\Gamma_p)} = \|R_p^T u_h^p\|_{\widetilde{H}^{1/2}(\Gamma)}$ *which is not a priori guaranteed for other norms. And even if an equivalence relation can be proved, extra care must be taken to show that the constants in the equivalence relation are independent of the size of the subdomain* Γ_p *which is related to the number of subdomains. As a consequence, the choice of the norm* $\|\cdot\|_{\widetilde{H}^{1/2}(\Gamma)}$ *plays a key role in the present analysis.*

5.3.3 Stable decomposition — GenEO concept

We now focus on the second hypothesis of Theorem 3.9. The goal is to find a way to decompose global finite element functions so that we can bound the global energy from below by the sum of the local energies. To do so, let us present how to apply the GenEO approach to BEM matrices. First, using the results from the preceding section, we have the following lemma for a generic coarse space:

Lemma 5.14. *Using the notations of Theorem 3.9, let* $\mathbf{u}_h \in \mathbb{R}^N$ *and* $\mathbf{U}_h = (\mathbf{u}_h^p)_{p=0}^n \in \prod_{p=0}^n \mathbb{R}^{N_p}$ *such that* $\mathbf{u}_h = \mathcal{R}_{\mathrm{ASM}}(\mathbf{U}_h)$*. Then, we have*

$$(B\mathbf{U}_h, \mathbf{U}_h)_D \leqslant 2(\mathbf{A}_{\mathrm{W}}\mathbf{u}_h, \mathbf{u}_h) + (1 + 5N_c\frac{C_{\mathrm{W}}^+}{C_{\mathrm{W}}^-})\sum_{p=1}^n (\mathbf{A}_{\mathrm{W}}\mathbf{R}_p^T\mathbf{u}_h^p, \mathbf{R}_p^T\mathbf{u}_h^p).$$

Proof. Using the definition of B and the Cauchy-Schwarz inequality,

$$\begin{aligned}
(B\mathbf{U}_h, \mathbf{U}_h)_D &= (\mathbf{A}_{\mathrm{W}}\mathbf{R}_0^T\mathbf{u}_h^0, \mathbf{R}_0^T\mathbf{u}_h^0) + \sum_{p=1}^n (\mathbf{A}_{\mathrm{W}}\mathbf{R}_p^T\mathbf{u}_h^p, \mathbf{R}_p^T\mathbf{u}_h^p) \\
&= \left(\mathbf{A}_{\mathrm{W}}\left(\mathbf{u}_h - \sum_{p=1}^n \mathbf{R}_p^T\mathbf{u}_h^p\right), \mathbf{u}_h - \sum_{p=1}^n \mathbf{R}_p^T\mathbf{u}_h^p\right) + \sum_{p=1}^n (\mathbf{A}_{\mathrm{W}}\mathbf{R}_p^T\mathbf{u}_h^p, \mathbf{R}_p^T\mathbf{u}_h^p) \\
&\leqslant 2\left((\mathbf{A}_{\mathrm{W}}\mathbf{u}_h, \mathbf{u}_h) + \left(\mathbf{A}_{\mathrm{W}}\sum_{p=1}^n \mathbf{R}_p^T\mathbf{u}_h^p, \sum_{p=1}^n \mathbf{R}_p^T\mathbf{u}_h^p\right)\right) + \sum_{p=1}^n (\mathbf{A}_{\mathrm{W}}\mathbf{R}_p^T\mathbf{u}_h^p, \mathbf{R}_p^T\mathbf{u}_h^p).
\end{aligned}$$

Then, we apply the equivalence relation from Equation (5.2) and Lemma 5.7 to the second term in the right-hand side to obtain the desired result. □

The last lemma is insufficient to prove a spectral estimate of the ASM preconditioner. There remains to bound the last term, corresponding to the sum of local energies, by the global energy $a(P_h\mathbf{u}_h, P_h\mathbf{u}_h)$. Let us assume there exists $(\mathbf{B}_p)_{p=1}^n \in \prod_{p=1}^n \mathbb{R}^{N_p \times N_p}$ and a constant $C_{\mathrm{loc}} > 0$ independent of h and n such that

$$C_{\mathrm{loc}} \sum_{p=1}^n (\mathbf{B}_p\mathbf{R}_p\mathbf{u}_h, \mathbf{R}_p\mathbf{u}_h) \leqslant a_{\mathrm{W}}(P_h\mathbf{u}_h, P_h\mathbf{u}_h). \tag{5.6}$$

To obtain the second hypothesis of Theorem 3.9, and using Lemma 5.14 with Equation (5.6), one can see that a sufficient condition would be to find a decomposition $\mathbf{U}_h = (\mathbf{u}_h^p)_{p=0}^n \in$

$\prod_{p=0}^{n} \mathbb{R}^{N_p}$ of a given $\mathbf{u}_h \in \mathbb{R}^N$ such that

$$(\mathbf{A}_{\mathrm{W}} \mathbf{R}_p^T \mathbf{u}_h^p, \mathbf{R}_p^T \mathbf{u}_h^p) \leqslant \tau(\mathbf{B}_p \mathbf{R}_p \mathbf{u}_h, \mathbf{R}_p \mathbf{u}_h), \tag{5.7}$$

for some fixed user-defined parameter $\tau > 0$ and $\mathbf{u}_h = \mathcal{R}_{\mathrm{ASM}}(\mathbf{U}_h)$. A natural choice for $(\mathbf{u}_h^p)_{p=0}^n$ might be $\mathbf{u}_h^p = \mathbf{D}_p \mathbf{R}_p \mathbf{u}_h$, but generally it does not satisfy Equation (5.7). The idea of the GenEO coarse space is to filter out the part of $\mathbf{D}_p \mathbf{R}_p \mathbf{u}_h$ that does not satisfy Equation (5.7) using the following local generalized eigenvalue problem: find $(\mathbf{v}_{h,k}^p, \lambda_k^p)$ such that

$$\mathbf{D}_p \mathbf{R}_p \mathbf{A}_{\mathrm{W}} \mathbf{R}_p^T \mathbf{D}_p \mathbf{v}_{h,k}^p = \lambda_k^p \mathbf{B}_p \mathbf{v}_{h,k}^p. \tag{5.8}$$

Then, we can define the local contribution to the coarse space

$$Z_{p,\tau} := \ker(\mathbf{B}_p) \cup \mathrm{Span}(\mathbf{v}_{h,k}^p \,|\, \text{ for every } k \text{ s.t. } \lambda_k^p > \tau), \tag{5.9}$$

and the local projection π_p on $Z_{p,\tau}$ parallel to $\mathrm{Span}(\mathbf{v}_{h,k}^p \,|\, \text{ for every } k \text{ s.t. } \lambda_k^p \leqslant \tau)$. Using the projection π_p, we can filter out the part of $\mathbf{D}_p \mathbf{R}_p \mathbf{u}_h$ that does not satisfy the sufficient condition given by Equation (5.7).

Lemma 5.15 ([45, Lemma 7.15])**.** *For every $1 \leqslant p \leqslant n$ and $\mathbf{v}_{h,k} \in \mathbb{R}^{N_p}$, we have*

$$(\mathbf{R}_p^T \mathbf{D}_p (\mathbf{I}_d - \pi_p) \mathbf{v}_h^p)^T \mathbf{A}_{\mathrm{W}} (\mathbf{R}_p^T \mathbf{D}_p (\mathbf{I}_d - \pi_p) \mathbf{v}_h^p) \leqslant \tau (\mathbf{v}_h^p)^T \mathbf{B}_p \mathbf{v}_h^p.$$

The last relation is similar to Equation (5.7), we define the decomposition such that, for $1 \leqslant p \leqslant n$

$$\mathbf{u}_h^p = \mathbf{D}_p (\mathbf{I}_d - \pi_p) \mathbf{R}_p \mathbf{u}_h. \tag{5.10}$$

It remains to define the coarse space $\mathcal{V}_{h,0}$ and the associated coarse component $\mathbf{u}_h^0$ such that $\mathbf{u}_h = \mathcal{R}_{\mathrm{ASM},2}(\mathbf{U}_h)$ with $\mathbf{U}_h = (\mathbf{u}_h^p)_{p=0}^n$.

Definition 5.16. *The GenEO coarse space is defined as the sum of the local contributions to the coarse space weighted with the partition of unity. We define*

$$\mathcal{V}_{h,0} = \mathrm{Span}(\mathbf{R}_p^T \mathbf{D}_p \mathbf{v}_h^p \,|\, 1 \leq p \leq N,\ \mathbf{v}_h^p \in Z_{p,\tau})$$

Let $\mathbf{Z}_\tau \in \mathbb{R}^{N \times N_0}$ be a column matrix so that $\mathcal{V}_{h,0}$ is spanned by its columns and $N_0 = \dim(\mathcal{V}_{h,0})$. We denote its transpose by $\mathbf{R}_0 := \mathbf{Z}_\tau^T$.

Lemma 5.17 (GenEO coarse component)**.** *Assuming there exists $(\mathbf{B}_p)_{p=1}^n$ such that Equation (5.6) is true, we can define the coarse space as in Definition 5.16, and the coarse component as*

$$\mathbf{u}_h^0 := (\mathbf{R}_0 \mathbf{R}_0^T)^{-1} \mathbf{R}_0 \left(\sum_{p=1}^{n} \mathbf{R}_p^T \mathbf{D}_p \pi_p \mathbf{R}_p \mathbf{u}_h \right), \tag{5.11}$$

then $\mathbf{U}_h = (\mathbf{u}_h^p)_{p=0}^n \in H_D$ defined by Equations (5.10) and (5.11) is a stable decomposition of $\mathbf{u}_h$, that is to say

$$\mathcal{R}_{\mathrm{ASM},2}(\mathbf{U}_h) = \mathbf{u}_h \quad \text{and} \quad c_T \sum_{p=0}^{n} a_{\mathrm{W}}(P_h \mathbf{R}_p^T \mathbf{u}_h^p, P_h \mathbf{R}_p^T \mathbf{u}_h^p) \leqslant a_{\mathrm{W}}(P_h \mathbf{u}_h, P_h \mathbf{u}_h),$$

with $c_T^{-1} = 2 + \left(1 + 5N_c \frac{C_{\mathrm{W}}^+}{C_{\mathrm{W}}^-}\right) \frac{\tau}{C_{\mathrm{loc}}}$.

Proof. Let $\mathbf{u}_h \in \mathbb{R}^N$ and $\mathbf{U}_h = (\mathbf{u}_h^p)_{0 \leqslant p \leqslant n} \in H_D$ defined by Equations (5.10) and (5.11). First notice that we have

$$\mathbf{w}_0 = \mathbf{R}_0^T (\mathbf{R}_0 \mathbf{R}_0^T)^{-1} \mathbf{R}_0 \mathbf{w}_0$$

for every $\mathbf{w}_0 \in \mathcal{V}_{h,0}$ since $\mathbf{R}_0^T (\mathbf{R}_0 \mathbf{R}_0^T)^{-1} \mathbf{R}_0$ is a projector on $\mathcal{V}_{h,0}$. Then, we have

$$\begin{aligned}
\mathcal{R}_{\mathrm{ASM},2}(\mathbf{U}_h) &= \sum_{p=0}^{n} \mathbf{R}_p^T \mathbf{u}_h^p = \mathbf{R}_0^T \mathbf{u}_h^0 + \sum_{p=1}^{n} \mathbf{R}_p^T \mathbf{u}_h^p \\
&= \mathbf{R}_0^T (\mathbf{R}_0 \mathbf{R}_0^T)^{-1} \mathbf{R}_0 \underbrace{\left(\sum_{p=1}^{n} \mathbf{R}_p^T \mathbf{D}_p \pi_p \mathbf{R}_p \mathbf{u}_h \right)}_{\in \mathcal{V}_{h,0}} + \sum_{p=1}^{n} \mathbf{R}_p^T \mathbf{u}_h^p \\
&= \sum_{p=1}^{n} \mathbf{R}_p^T \mathbf{D}_p \pi_p \mathbf{R}_p \mathbf{u}_h + \sum_{p=1}^{n} \mathbf{R}_p^T \mathbf{D}_p (\mathbf{I}_d - \pi_p) \mathbf{R}_p \mathbf{u}_h \\
&= \sum_{p=1}^{n} \mathbf{R}_p^T \mathbf{D}_p \mathbf{R}_p \mathbf{u}_h = \mathbf{u}_h,
\end{aligned}$$

where we used the fact that $\mathbf{D}_p$ defines a partition of unity in the last line. Then, using Lemma 5.14 and Lemma 5.15 with $\mathbf{v}_h^p = \mathbf{R}_p \mathbf{u}_h$ for $1 \leqslant p \leqslant n$, we have

$$\begin{aligned}
\sum_{p=0}^{n} a(P_h \mathbf{R}_p^T \mathbf{u}_h^p, P_h \mathbf{R}_p^T \mathbf{u}_h^p) &= \sum_{p=0}^{n} (\mathbf{A}_{\mathrm{W}} \mathbf{R}_p^T \mathbf{u}_h^p, \mathbf{R}_p^T \mathbf{u}_h^p) \\
&\leqslant 2(\mathbf{A}_{\mathrm{W}} \mathbf{u}_h, \mathbf{u}_h) + (1 + 5N_c \frac{C_{\mathrm{W}}^+}{C_{\mathrm{W}}^-}) \sum_{p=1}^{n} (\mathbf{A}_{\mathrm{W}} \mathbf{R}_p^T \mathbf{u}_h^p, \mathbf{R}_p^T \mathbf{u}_h^p) \\
&\leqslant 2(\mathbf{A}_{\mathrm{W}} \mathbf{u}_h, \mathbf{u}_h) + (1 + 5N_c \frac{C_{\mathrm{W}}^+}{C_{\mathrm{W}}^-}) \tau \sum_{p=1}^{n} (\mathbf{B}_p \mathbf{R}_p \mathbf{u}_h, \mathbf{R}_p \mathbf{u}_h).
\end{aligned}$$

Finally, assuming Equation (5.6) holds, we deduce

$$\sum_{p=0}^{n} a_{\mathrm{W}}(P_h \mathbf{R}_p^T \mathbf{u}_h^p, P_h \mathbf{R}_p^T \mathbf{u}_h^p) \leqslant \left(2 + \left(1 + 5N_c \frac{C_{\mathrm{W}}^+}{C_{\mathrm{W}}^-}\right) \frac{\tau}{C_{\mathrm{loc}}}\right) a_{\mathrm{W}}(P_h \mathbf{u}_h, P_h \mathbf{u}_h).$$

□

Remark 5.18. *As pointed out in Remark 3.28, it should be noted that the decomposition* $(\mathbf{u}_h^p)_{p=0}^n$ *is necessary for the analysis, but it does not need to be computed in practice. To use this coarse space, it is sufficient to compute* $\mathbf{Z}_\tau$ *and then* $(\mathbf{R}_0 \mathbf{A}_{\mathrm{W}} \mathbf{R}_0^T)^{-1}$ *to be able to apply* $\mathbf{P}_{ASM}$ *(see definition given in Equation (3.12)). And to do so, each local contribution to the coarse space defined in Equation (5.9) can be computed independently in parallel, so that computing* $\mathbf{Z}_\tau$ *can be done efficiently. Then, to compute* $\mathbf{R}_0 \mathbf{A}_{\mathrm{W}} \mathbf{R}_0^T$*, one can use the fact that* $\mathbf{Z}_\tau$ *is sparse and distributed (since the eigenvectors are computed in parallel). Finally, we need to invert*

$(\mathbf{R}_0\mathbf{A}_{\mathrm{W}}\mathbf{R}_0^T) \in \mathbb{R}^{N_0\times N_0}$ *and that is why a good coarse space should be of minimum size while still containing relevant information for the convergence.*

5.3.4 Concrete coarse spaces

Using Lemmas 5.12 and 5.17, it remains to find a sequence of local operators $(\mathbf{B}_p)_{p=1}^n$ that satisfies Equation (5.6) to be able to apply Theorem 3.9. The goal is to find a coarse space whose size is as small as possible for a given τ because we need to invert $\mathbf{R}_0\mathbf{A}_{\mathrm{W}}\mathbf{R}_0^T \in \mathbb{R}^{N_0\times N_0}$. However, the choice of the tuple $(\mathbf{B}_p)_{p=1}^n$ is not a priori unique, and we can use Lemmas 5.8 to 5.10 to define various generalized eigenproblems as Equation (5.8). Besides, we do not know the size of the coarse space in advance for a given τ, so that numerical tests are necessary to see which one is the most efficient. A first simple way to obtain a decomposition $(\mathbf{B}_p)_{p=1}^n$ satisfying Equation (5.6) is to simply use Lemma 5.9.

Lemma 5.19. *Let $u \in \widetilde{H}^{1/2}(\Gamma)$, we have*

$$\frac{C_{\mathrm{inj}}C_{\mathrm{W}}^-}{k_1}\sum_{p=1}^n \|u|_{\Gamma_p}\|^2_{L^2(\Gamma_p)} \leqslant a_{\mathrm{W}}(u,u),$$

where k_1 is defined in Definition 5.5, C_{inj} is continuity constant of the injection of $\widetilde{H}^{1/2}(\Gamma)$ in $L^2(\Gamma)$, and they are both independent of h and n.

Proof. Using Lemma 5.9, we obtain

$$a_{\mathrm{W}}(u,u) \geq C_{\mathrm{W}}^-\|u\|^2_{\widetilde{H}^{1/2}(\Gamma)} \geq \frac{C_{\mathrm{inj}}C_{\mathrm{W}}^-}{k_1}\sum_{p=1}^n \|u|_{\Gamma_p}\|^2_{L^2(\Gamma_p)}.$$

□

Using the last lemma, we obtain a similar relation to Equation (5.6) with $C_{\mathrm{loc}} = \frac{C_{\mathrm{inj}}C_{\mathrm{W}}^-}{k_1}$ and $\mathbf{B}_p = \mathbf{M}_p$ for every $1 \leqslant p \leqslant n$ where $\mathbf{M}_p$ is the mass matrix defined as

$$(\mathbf{M}_p)_{i,j} := \int_{\Gamma_p} \varphi_{\sigma_p(i)}\varphi_{\sigma_p(j)}\,\mathrm{d}\gamma, \tag{5.12}$$

where Γ_p has been defined so that $\mathbf{M}_p \in \mathbb{R}^{N_p\times N_p}$. Then, we can obtain the following theorem

Theorem 5.20. *We have the following condition number estimate*

$$\kappa(\mathbf{P}_{\mathrm{ASM,inj}}\mathbf{A}_{\mathrm{W}}) \leqslant 2\max\left(1, N_c\frac{5}{2}\frac{C_{\mathrm{W}}^+}{C_{\mathrm{W}}^-}\right)\left(2+\left(1+5N_c\frac{C_{\mathrm{W}}^+}{C_{\mathrm{W}}^-}\right)\frac{\tau k_1}{C_{\mathrm{inj}}C_{\mathrm{W}}^-}\right),$$

where $\mathbf{P}_{\mathrm{ASM,inj}}$ is defined with the GenEO coarse space and the localization from Lemma 5.19.

Proof. Using Theorem 3.9 and its notations, we have to satisfy its two hypotheses.

(i) The first hypothesis is obtained using Lemma 5.12 with $c_R = 2\max(1, N_c\frac{5C_{\mathrm{W}}^+}{2C_{\mathrm{W}}^-})$.

(ii) The second hypothesis is obtained using Lemma 5.17 with $\mathbf{B}_p = \mathbf{M}_p$ for every $1 \leqslant p \leqslant n$ and Lemma 5.19 so that $c_T^{-1} = 2 + \left(1 + 5N_c\frac{C_{\mathrm{W}}^+k_1}{C_{\mathrm{W}}^-}\right)\frac{\tau k_1}{C_{\mathrm{inj}}C_{\mathrm{W}}^-}$.

□

We will refer to the coarse space based on Theorem 5.20 as *GenEO mass coarse space*. Another approach to build a coarse space is to use Lemma 5.10 to obtain the next lemma.

Lemma 5.21. *Let $u_h \in \widetilde{S}^l(\mathcal{T}_h)$, we have*

$$\frac{C_{\mathrm{ad}}C_{\mathrm{W}}^-}{k_1}\sum_{p=1}^{n}\|h_{\mathcal{T}}^{1/2}\nabla_\Gamma u_h|_{\Gamma_p}\|^2_{L^2(\Gamma_p)} \leqslant a(u_h,u_h),$$

where C_{ad} depends only on Γ, the shape-regularity of $\mathcal{T}_h$ and l.

Proof. According to Lemma 5.10 and the equivalence relation Equation (5.2), we have

$$\begin{aligned} a(u_h,u_h) &\geq C_{\mathrm{W}}^-\|u_h\|^2_{\widetilde{H}^{1/2}(\Gamma)} \geq C_{\mathrm{ad}}C_{\mathrm{W}}^-\|h_{\mathcal{T}}^{1/2}\nabla_\Gamma u_h\|^2_{L^2(\Gamma)} \\ &\geq \frac{C_{\mathrm{ad}}C_{\mathrm{W}}^-}{k_1}\sum_{p=1}^{n}\|h_{\mathcal{T}}^{1/2}\nabla u_h|_{\Gamma_p}\|^2_{L^2(\Gamma_p)}. \end{aligned}$$

□

Using the last lemma, we obtain a similar relation to Equation (5.6) with $C_{\mathrm{loc}} = \frac{C_{\mathrm{ad}}C_{\mathrm{W}}^-}{k_1}$ and $\mathbf{B}_p = \mathbf{K}_p$ for every $1 \leqslant p \leqslant n$ where $\mathbf{K}_p$ is the stiffness matrix defined as

$$(\mathbf{K}_p)_{i,j} := \int_{\Gamma_p} h_{\mathcal{T}}\nabla_\Gamma\varphi_{\sigma_p(i)}\nabla_\Gamma\varphi_{\sigma_p(j)}\,\mathrm{d}\gamma, \tag{5.13}$$

where again, $\mathbf{K}_p \in \mathbb{R}^{N_p\times N_p}$ due to the definition of Γ_p. Then, we can obtain the following theorem

Theorem 5.22. *We have the following condition number estimate*

$$\kappa(\mathbf{P}_{\mathrm{ASM,ad}}\mathbf{A}_{\mathrm{W}}) \leqslant 2\max\left(1, N_c\frac{5}{2}\frac{C_{\mathrm{W}}^+}{C_{\mathrm{W}}^-}\right)\left(2+\left(1+5N_c\frac{C_{\mathrm{W}}^+}{C_{\mathrm{W}}^-}\right)\frac{\tau k_1}{C_{\mathrm{ad}}C_{\mathrm{W}}^-}\right),$$

where $\mathbf{P}_{\mathrm{ASM,ad}}$ is defined with the GenEO coarse space and the localization from Lemma 5.21.

Proof. Using Theorem 3.9 and its notations, we have to satisfy its two hypotheses.

(i) The first hypothesis is obtained using Lemma 5.12 with $c_R = 2\max(1, N_c\frac{5C_{\mathrm{W}}^+}{2C^-})$.

(ii) The second hypothesis is obtained using Lemma 5.17 with $\mathbf{B}_p = \mathbf{K}_p$ for every $1 \leqslant p \leqslant n$ and Lemma 5.21 so that $c_T^{-1} = 2 + \left(1 + 5N_c\frac{C_{\mathrm{W}}^+k_1}{C_{\mathrm{W}}^-}\right)\frac{\tau k_1}{C_{\mathrm{ad}}C_{\mathrm{W}}^-}$.

□

We will refer to the coarse space based on Theorem 5.22 as *GenEO stiffness coarse space*. A third approach is to use Lemma 5.8

Lemma 5.23. *For $u \in H^{1/2}(\Gamma)$, there holds*

$$\frac{C_{\mathrm{W}}^-}{k_1}\sum_{p=1}^{n}\|u|_{\Gamma_p}\|^2_{H^{1/2}(\Gamma_p)} \leqslant a_{\mathrm{W}}(u,u),$$

where k_1 is defined in Definition 5.5.

Proof. According to Lemma 5.8 and the equivalence relation Equation (5.2), we have

$$a_{\mathrm{W}}(u,u) \geq C_{\mathrm{W}}^{-}\|u\|^2_{\widetilde{H}^{1/2}(\Gamma)} \geq \frac{C_{\mathrm{W}}^{-}}{k_1}\sum_{p=1}^{n}\|u|_{\Gamma_p}\|^2_{H^{1/2}(\Gamma_p)}.$$

□

Using the last lemma, we obtain a similar relation to Equation (5.6) with $C_{\mathrm{loc}} = \frac{C_{\mathrm{W}}^{-}}{k_1}$ and $\mathbf{B}_p = \mathbf{H}_p$ for every $1 \leqslant p \leqslant n$ where $\mathbf{H}_p$ is the matrix associated with the scalar product of $H^{1/2}(\Gamma)$ defined as

$$\begin{aligned}(\mathbf{H}_p)_{i,j} := &\int_{\Gamma_p} \varphi_{\sigma_p(i)}\varphi_{\sigma_p(j)}\,\mathrm{d}\sigma(\mathbf{x})\,\mathrm{d}\sigma(\mathbf{y}) + \\ &\int_{\Gamma_p\times\Gamma_p} \frac{(\varphi_{\sigma_p(i)}(\mathbf{x}) - \varphi_{\sigma_p(i)}(\mathbf{y}))(\varphi_{\sigma_p(j)}(\mathbf{x}) - \varphi_{\sigma_p(j)}(\mathbf{y}))}{|\mathbf{x}-\mathbf{y}|^{d+1}}\,\mathrm{d}\sigma(\mathbf{x})\,\mathrm{d}\sigma(\mathbf{y}),\end{aligned} \tag{5.14}$$

where again, $\mathbf{H}_p \in \mathbb{R}^{N_p\times N_p}$ due to the definition of Γ_p. Then, we can obtain the following theorem

Theorem 5.24. *We have the following condition number estimate*

$$\kappa(\mathbf{P}_{\mathrm{ASM,slo}}\mathbf{A}_{\mathrm{W}}) \leqslant 2\max\left(1, N_c\frac{5}{2}\frac{C_{\mathrm{W}}^{+}}{C_{\mathrm{W}}^{-}}\right)\left(2 + \left(1 + 5N_c\frac{C_{\mathrm{W}}^{+}}{C_{\mathrm{W}}^{-}}\right)\frac{\tau k_1}{C_{\mathrm{W}}^{-}}\right),$$

where $\mathbf{P}_{\mathrm{ASM,slo}}$ is defined with the GenEO coarse space and the localization from Lemma 5.23.

Proof. Using Theorem 3.9 and its notations, we have to satisfy its two hypotheses.

(i) The first hypothesis is obtained using Lemma 5.12 with $c_R = 2\max(1, N_c\frac{5C_{\mathrm{W}}^{+}}{2C_{\mathrm{W}}^{-}})$.

(ii) The second hypothesis is obtained using Lemma 5.17 with $\mathbf{B}_p = \mathbf{H}_p$ for every $1 \leqslant p \leqslant n$ and Lemma 5.23 so that $c_T^{-1} = 2 + \left(1 + 5N_c\frac{C_{\mathrm{W}}^{+}k_1}{C_{\mathrm{W}}^{-}}\right)\frac{\tau k_1}{C_{\mathrm{W}}^{-}}$.

□

We will refer to the coarse space based on Theorem 5.24 as *GenEO Slobodeckij*. Now that we have introduced these three coarse spaces, we can make the following remarks:

Remark 5.25. *The GenEO Stiffness coarse space is inexpensive to compute since it is sparse and similar to the stiffness matrix of a Laplace problem weighted by the piecewise constant function $h_{\mathcal{T}}$. However the constant in Theorem 5.22 may depend on the degree of the finite element functions.*

Remark 5.26 (Computational difficulties for GenEO Slobodeckij coarse space)**.** *The GenEO Slobodeckij coarse space requires to compute the matrix associated with the $H^{1/2}$ scalar product. It induces new practical difficulties compared to usual BEM matrices. To see that, notice that the norm of a function φ with compact support given by a BIO is usually of the type $\int_{\Gamma\times\Gamma} K(\mathbf{x},\mathbf{y})\varphi(\mathbf{x})\varphi(\mathbf{y})\,\mathrm{d}\sigma(\mathbf{x})\,\mathrm{d}\sigma(\mathbf{y})$ where $K(\mathbf{x},\mathbf{y})$ is a kernel depending on the BIO. Even if the*

kernel is non-local, the previous integral reduces to a double integral on $\mathrm{supp}(\varphi)$*, the computation remains local. This is not the case for the Sobolev-Slobodeckij semi-norm which reads*

$$\int_{\Gamma\times\Gamma} \frac{|\varphi(\mathbf{x})-\varphi(\mathbf{y})|^2}{|\mathbf{x}-\mathbf{y}|^{d+1}}\,\mathrm{d}\sigma(\mathbf{x})\,\mathrm{d}\sigma(\mathbf{y}) = 2\int_{\mathrm{supp}(\varphi)\times\Gamma} \frac{|\varphi(\mathbf{x})|^2}{|\mathbf{x}-\mathbf{y}|^{d+1}}\,\mathrm{d}\sigma(\mathbf{x})\,\mathrm{d}\sigma(\mathbf{y}) - 2\int_{\mathrm{supp}(\varphi)\times\mathrm{supp}(\varphi)} \frac{\varphi(\mathbf{x})\varphi(\mathbf{y})}{|\mathbf{x}-\mathbf{y}|^{d+1}}\,\mathrm{d}\sigma(\mathbf{x})\,\mathrm{d}\sigma(\mathbf{y}),$$

so that there is an integral over the whole domain in any case. In other words, the Sobolev-Slobodeckij norm is "more non-local" than usual BIO, which makes the computation of $\mathbf{H}_p$ *difficult, see Appendix A for more practical details.*

Due to the practical difficulties in computing $\mathbf{H}_p$ explained in Remark 5.26, another possibility is to approximate the $H^{1/2}$-norm in Lemma 5.23. It is natural to introduce the weakly singular operator $\mathrm{V} : \widetilde{H}^{-1/2}(\Gamma) \to H^{1/2}(\Gamma)$ defined in Section 2.3.2, whose range is the right space under consideration. We suppose that V is symmetric positive definite, which is true when we assume that the hypersingular operator W also has this property. Then, we can define the local weakly singular operator V_p as

$$\langle \mathrm{V}_p\, u_p, v_p\rangle_{H^{1/2}(\Gamma_p)\times\widetilde{H}^{-1/2}(\Gamma_p)} := \langle \mathrm{V}\,\mathrm{E}_{\Gamma_p}(u_p), \mathrm{E}_{\Gamma_p}(v_p)\rangle_{H^{1/2}(\Gamma)\times\widetilde{H}^{-1/2}(\Gamma)}$$

for every u_p, $v_p \in \widetilde{H}^{-1/2}(\Gamma_p)$. Now, we can define a norm on $H^{1/2}(\Gamma_p)$ using the inverse of the local weakly singular operator and equivalent to $\|u\|^2_{H^{1/2}(\Gamma_p)}$. Unfortunately, we have not been able to fully analyse this preconditioner, and this is mainly related to the lack of control on the equivalence constants between $\|u_p\|^2_{\widetilde{H}^{-1/2}(\Gamma_p)}$ and $\|\mathrm{E}_p(u_p)\|^2_{\widetilde{H}^{-1/2}(\Gamma)}$, we refer to Appendix B for a more detailed discussion.

That is why, we only define the associated discrete local operator $\mathbf{B}_p = \widetilde{\mathbf{C}}_p := \mathbf{M}_p\mathbf{V}_p^{-1}\mathbf{M}_p$, for every $1 \leqslant p \leqslant n$ where $\mathbf{V}_p$ is the discretization of the single layer defined as

$$\mathbf{V}_p(i,j) := \langle \mathrm{V}_p(\varphi_{\sigma_p(j)}|_{\Gamma_p}), \varphi_{\sigma_p(i)}|_{\Gamma_p}\rangle_{H^{1/2}(\Gamma_p)\times\widetilde{H}^{-1/2}(\Gamma_p)}, \tag{5.15}$$

where again, $\widetilde{\mathbf{C}}_p \in \mathbb{R}^{N_p\times N_p}$ due to the definition of Γ_p. The choice of $\widetilde{\mathbf{C}}_p$ comes from the fact that we need to approximate the discretization V_p^{-1} by the inverse of the discretization of V_p, which is classical in Calderón preconditioning, see [124, 146]. We will refer to the coarse space as *GenEO single layer*.

In conclusion, we have introduced four coarse spaces and Theorems 5.20, 5.22 and 5.24 showed that at least three of them lead to a bounded condition number for the preconditioned linear systems independently of the mesh size and the number of subdomains. The size of the coarse spaces will adapt to the user-defined parameter τ, which is one of the advantages of this approach. The downside, though, is that we do not know these sizes a priori. For example, if one has to take all the eigenvectors in the local generalized eigenproblem defined in Equation (5.8), the size of the coarse space will be the size of the global matrix, and thus inverting $\mathbf{R}_0\mathbf{A}_{\mathrm{W}}\mathbf{R}_0^T$ will be as costly as using a direct method. We have to verify how relevant these coarse spaces are numerically, and which one retains the most of relevant information. In other words, we have to check numerically that the sizes of the coarse spaces are small compared to the size of the global problem, and which one is the smallest while improving the convergence.

The only intuition we can have a priori is about GenEO single layer and it will be discussed in Section 5.4.2.

5.4 Numerical study of GenEO coarse spaces for the hypersingular operator

To see if the coarse spaces introduced in the previous section are well-adapted in practice, we need to verify that they improve *scalability*. Several definitions of scalability can be given, see Definitions 3.16 and 3.17. For *strong scalability* tests, we study how the number of iterations varies when we increase the number of subdomains for a given problem size. By contrast, for *weak scalability* tests the local problem size is kept constant, in other words doubling the number of subdomains means doubling the global problem size. In both cases, we say that the method is scalable if the number of iterations does not increase with the number of subdomains. In a HPC context, where we usually assign one subdomain per core, this yields robust methods in terms of computing times (for example, constant computing time for weak scalability tests). Indeed, the computing time should be proportional to the size of the subdomains and the number of iterations, provided the coarse problem $(\mathbf{R}_0\mathbf{A}_h\mathbf{R}_0^T)$ is small enough to be solved efficiently with a direct method.

According to Theorems 5.20, 5.22 and 5.24, we see that the condition number of $\mathbf{A}_{\mathrm{W}}$ preconditioned by the ASM defined in Equation (5.5) with the coarse spaces we introduced, is independent of the number or size of subdomains. The numerical experiments in this section will illustrate Theorems 5.20, 5.22 and 5.24 with up to 512 subdomains and show that the proposed methods are scalable, as the resulting coarse problems remain small enough for a direct solver.

Note that the preconditioners we introduced are independent of compression techniques, but one has to implement them efficiently in conjunction with a compression technique to be able to run tests of reasonable size. That is why we developed Htool, an open source C++ library for hierarchical matrices that offers with HPDDM a way to use DDM preconditioners with MPI and OpenMP. For more details, we refer to Chapter 6.

In particular, it allows numerical illustrations of the efficiency of the previous coarse spaces and their costs. We use BemTool[1] to compute the coefficients associated with the interaction between two degrees of freedom and a $\mathbb{P}_1$-Lagrange discretization, Htool[2] to compress the matrices using hierarchical matrices as described in [12, 75, 132] and HPDDM [104] which provides iterative solvers. The setup of the DDM preconditioners for BEM matrices is shared between HPDDM and Htool.

Concerning compression, hierarchical matrices are built upon a hierarchical decomposition of the degrees of freedom in clusters that is usually called cluster tree. Each block of the global matrix can be seen as the interaction between two nodes of this tree. Then, we define the block tree as the product of this cluster tree with itself, so that each node corresponds to a block in the global matrix. Because of the nature of the integral kernels, some of these blocks can be considered as *admissible* meaning that they correspond to far interactions so that they can be well approximated using low-rank matrices, while others are not admissible because they are associated with close interactions. In the latter case, one has to look at the subblocks via the block tree or the blocks need to be built as dense blocks.

To determine if a block is admissible, we use the admissibility condition described in [132, (3.15)]. A block is considered to be admissible if the minimum of the diameters of both clusters over the distance separating them is lower than a user-defined parameter η. Then, we use *Adaptive Cross Approximation* with partial pivoting similarly to [132, Algorithm 3.9] to compress the admissible blocks. The stopping criterion of this algorithm looks at the Frobenius norm between two consecutive approximations and if it is lower than a given threshold ϵ. In our

[1] https://github.com/xclaeys/BemTool
[2] https://github.com/PierreMarchand20/htool

numerical tests, we take $\eta = 10$ and $\epsilon = 0.01$. Note that the compression may depend on how the hierarchical matrix is parallelized, but in the case of the strong scaling, we made sure that the compression was the same to have a fair comparison with different subdomains.

Our domain decomposition is actually defined by the cluster tree. More precisely, one of the levels in the cluster tree defines a partition without overlap, to which we add overlap. Notice that the algorithm used to build the cluster tree tries to obtain a balanced decomposition between the clusters of nodes to have a better compression and load-balanced decomposition. We refer to Chapter 6 for more details about hierarchical matrices and our implementation.

We would like to point out that we are using GMSH [60] to create all the meshes in our test cases and LAPACK [6] to solve generalized eigenproblems. Besides, we use right preconditioning, we set the tolerance for the iterative solvers in HPDDM to 10^{-6}, and we have not used any restart with GMRes.

5.4.1 Test cases

Let $\Omega_{\text{out}} = [-2,2] \times [-2,2] \subset \mathbb{R}^2$ and $\Omega_{\text{in}} = [-1,1] \times [-1,1] \subset \mathbb{R}^2$. In this first numerical experiment, we want to solve the following equation

$$-\Delta u + \kappa^2 u = 0, \quad \text{in } \Omega := \overline{\Omega_{\text{out}} \setminus \Omega_{\text{in}}} \tag{5.16}$$

where $\kappa = 0.1$ with the Neumann condition $\gamma_N(u) = f(x,y)$ such that

$$f(x,y) = \begin{cases} 100 * (x+1.5)^2 & \text{if } x > 1.5 \\ 0 & \text{otherwise.} \end{cases}$$

This problem can be reformulated as a BIE using Equation (2.12) with the hypersingular operator W from Section 2.3.2. More precisely, we use the analytical expression of the hypersingular operator from Section 2.3.2 with $\mathbf{A} = \mathbf{I}_{\text{d}}$, $\mathbf{b} = \mathbf{0}$ and $c = \kappa^2$. The bilinear form a is symmetric positive definite according to Lemma 2.29.

In 3D, we consider a similar test case with cubes instead of squares. To be more precise, we have now $\Omega_{\text{out}} = [-2,2] \times [-2,2] \times [-2,2] \subset \mathbb{R}^3$ and $\Omega_{\text{in}} = [-1,1] \times [-1,1] \times [-1,1] \subset \mathbb{R}^3$ as described in Figure 5.4, but we still consider Equation (5.16) with the same boundary condition. The main difference, besides the geometry, is the Green function (see Equation (2.8)).

5.4.2 Spectrum

There are no theoretical results that establishes relevance of the GenEO coarse space, i.e. that it contains relevant information to ensure that it will be of small size while improving the convergence. This is the downside of having a black-box method that adapts the size of the coarse space to the problem, we cannot know in advance this size. But to see if we can expect a coarse space of small size, we can look at the distribution of the eigenvalues in Equation (5.8) and check that we can discriminate some high eigenvalues that we will put in the coarse space.

Using the 2D test case described in Section 5.4.1 with eight subdomains and $h = 0.1$, we obtain the eigenvalue distributions given in Figure 5.5 for the four coarse spaces. We observe that GenEO single layer, GenEO stiffness and GenEO Slobodeckij have a few high eigenvalues. This is not the case of GenEO mass so that we expect this coarse space to not have the relevant information to improve the convergence. Notice that it is not worth comparing the values of the eigenvalues between the different coarse spaces since they all have different constants appearing in the final bound on the condition number.

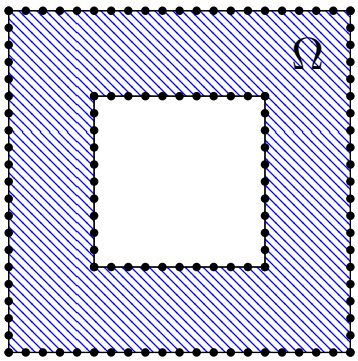


Figure 5.3 – 2D geometry used to test scalability of the DDM preconditioners for the hypersingular operator.

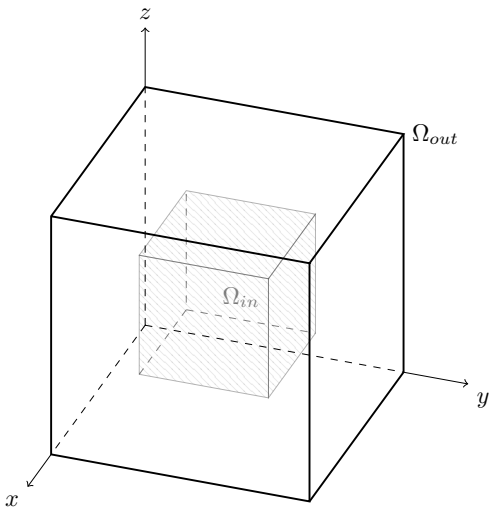


Figure 5.4 – 3D geometry used to test scalability of the DDM preconditioners for the hypersingular operator.

In the case of GenEO single layer, we can actually expect such a distribution of eigenvalues. Indeed, looking at the associated local generalized eigenproblem given by Equation (5.8) where $\mathbf{B}_p = \mathbf{M}_p\mathbf{V}_p^{-1}\mathbf{M}_p$ with $\mathbf{B}_p$ defined in Equation (5.15), it can be rewritten as a local eigenproblem

$$\mathbf{M}_p^{-1}\mathbf{V}_p\mathbf{M}_p^{-1}\mathbf{D}_p\mathbf{R}_p\mathbf{A}_{\mathrm{W}}\mathbf{R}_p^T\mathbf{D}_p\mathbf{v}_{h,k}^p = \lambda_k^p\mathbf{v}_{h,k}^p. \tag{5.17}$$

It is actually really close to a local Calderón preconditioning apart from $\mathbf{D}_p$. The subdomains usually correspond to screens and it is a well-known fact that Calderón preconditioning does not work well in this case. In practice, we observe that the eigenvalues are distributed as observed in [98, Figures 4.1 and 4.2], with a few high eigenvalues so that we expected to be able to discriminate a few eigenvectors for this coarse space.

5.4.3 Results in 2D

We solve the linear system associated with our test case and preconditioned with the coarse spaces introduced in the previous section. We only add one layer of elements for the overlap. To have a fair comparison between these coarse spaces, we take $\tau = 60$ for the GenEO stiffness coarse space, and we take the same number of eigenvectors in the local eigenproblems to build the coarse component of the other coarse spaces, so that the sizes of the local contributions to the coarse space and the size of the global coarse space are the same for a given number of subdomains with all the coarse spaces.

To test the strong scaling, we solve the considered problem with a mesh size $h = 0.001$ so that the number of degrees of freedom N is equal to 24 000 and we use several numbers of subdomains: $\{8, 16, 32, 64, 128\}$. The resulting numbers of iterations using CG and GMRes are given in Figure 5.6. Since τ is fixed, the condition number is bounded independently of the number of subdomains according to Theorem 5.22 for GenEO stiffness, so that we expect the number of iterations to be constant contrary to the case without the coarse component.

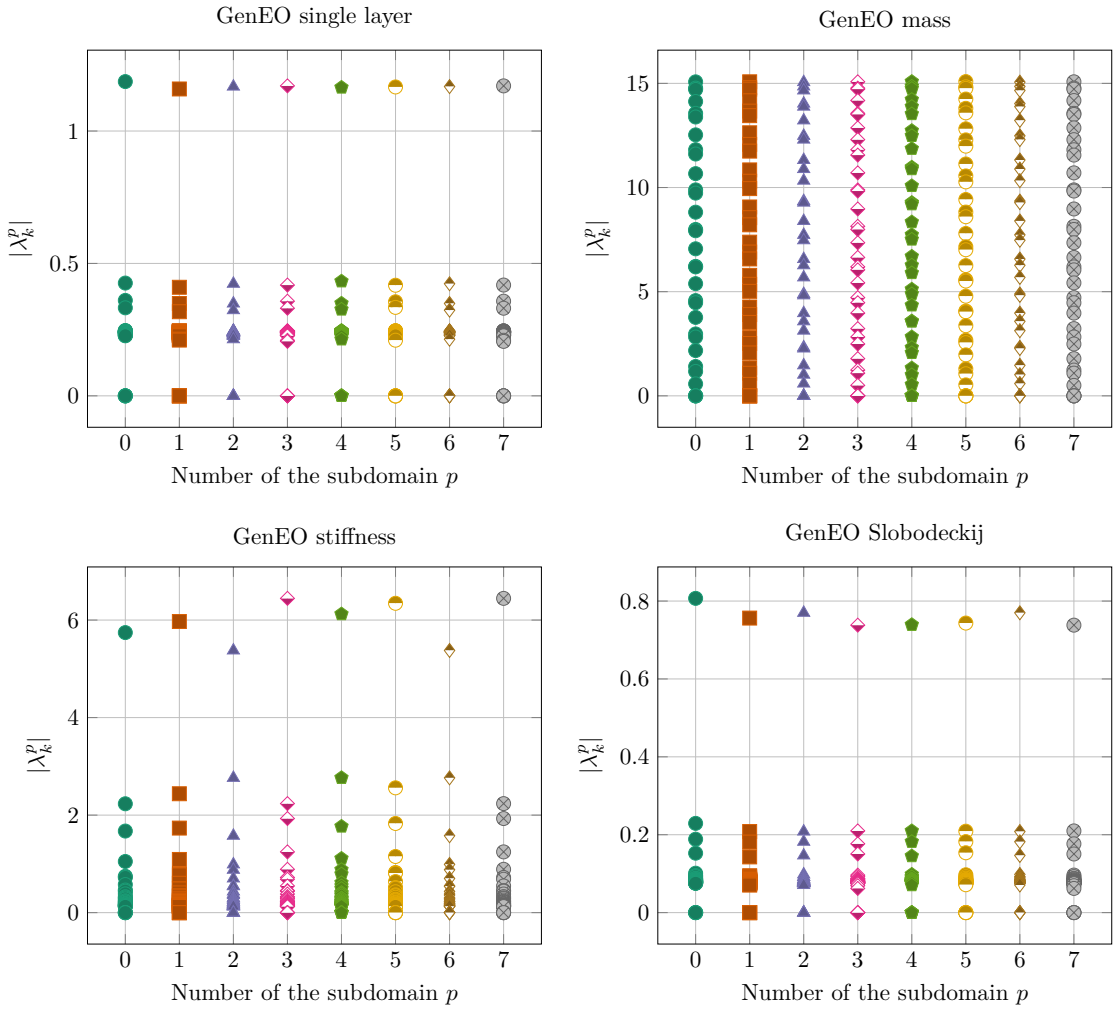


Figure 5.5 – Eigenvalue distributions for the generalized eigenproblem defined Equation (5.8) with eight subdomains and the hypersingular operator.

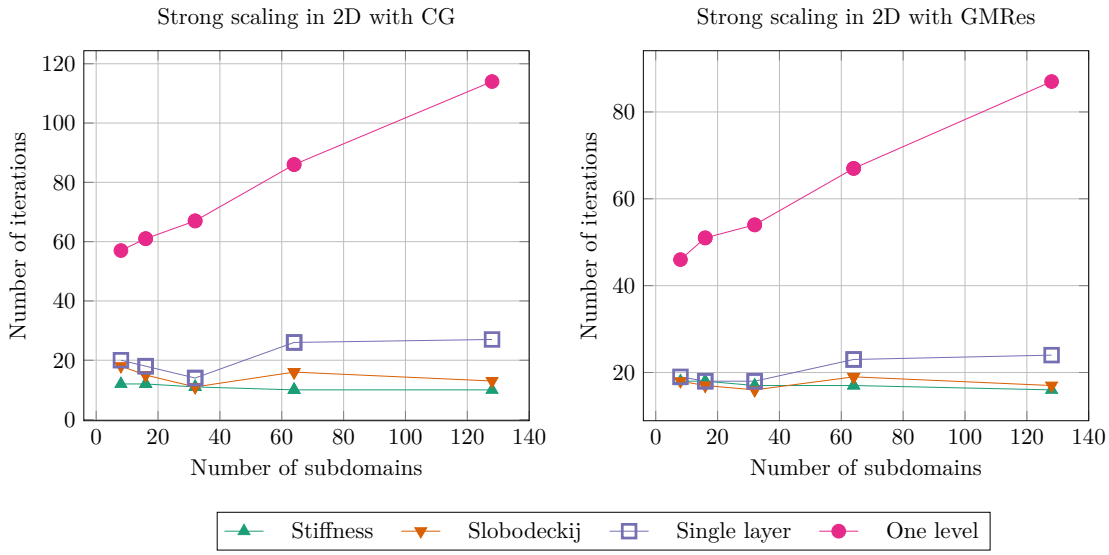


Figure 5.6 – Strong scaling to solve Equation (5.16) with the 2D geometry described in Figure 5.3 and the hypersingular operator.

The numbers of iterations without preconditioner are 656 for CG and 450 for GMRes, and they do not depend on the number of subdomains so that they are not shown in Figure 5.6. We first observe that the preconditioner without coarse component greatly reduces the number of iterations since it is approximately between 40 and 120 iterations instead of several hundreds without preconditioner. But as expected, the one-level preconditioner does not scale with the number of subdomains. It goes from 57 iterations for 8 subdomains to 114 iterations for 128 subdomains with CG for example. We do not show the results for GenEO mass because the results are at best close to the one-level, as expected from Section 5.4.2. We conclude that this coarse space needs to be much larger to scale in terms of iterations. GenEO Slobodeckij, GenEO stiffness and GenEO single layer scale, their number of iterations stagnates respectively between 10 and 27 iterations for CG. Note that the number of iterations for the latter is always greater than the other two. It shows that the size of the GenEO single layer coarse space needs to be greater to maintain the same amount of information. We call the number of eigenvectors taken to build $Z_{p,\tau}$, local contribution to the global coarse space associated with the pth subdomain and we show Figure 5.7 how the mean of these local contributions and the size of the global coarse space vary with the number of subdomains. It is interesting to see how the mean local contribution decreases with the number of subdomains so that the size of the global coarse space does not increase more than linearly.

Finally, we also did the experiment with a Calderón preconditioning. In this case, the preconditioner is defined as [146, (5.15)]

$$\mathbf{P}_{\text{Calderón},\mathbf{W}_h} := \mathbf{M}_h^{-1}\mathbf{V}_h\mathbf{M}_h^{-1}$$

where $\mathbf{M}_h$ is the global mass matrix. Using $\mathbf{P}_{\text{Calderón},\mathbf{W}_h}$ to precondition the linear system associated with the considered test case, we obtain 12 iterations with CG and GMRes.

To conclude about strong scaling, we were able to obtain a number of iterations comparable

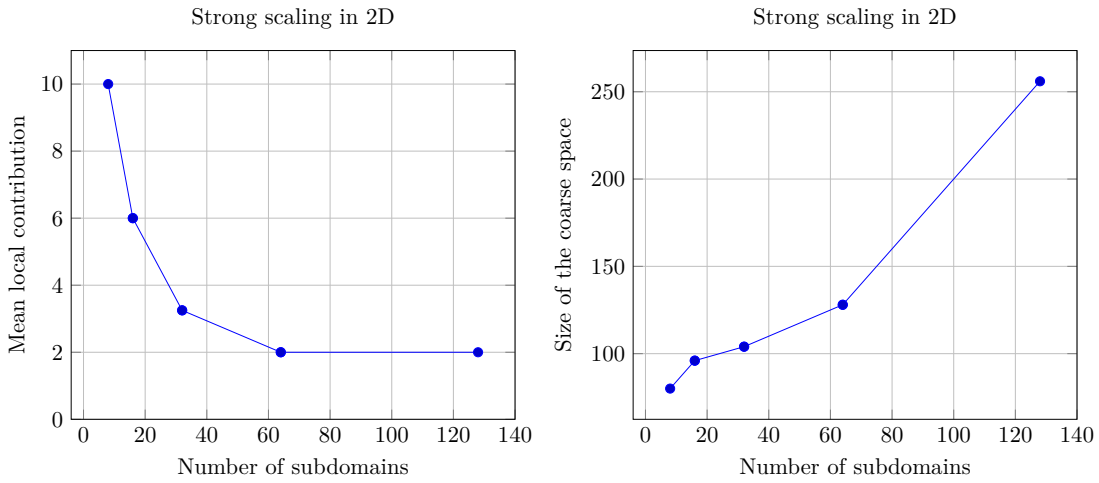


Figure 5.7 – Mean local contribution to coarse space and size of the latter for 2D strong scaling.

to Calderón preconditioning with the GenEO stiffness coarse space and GenEO single layer coarse space, while maintaining a good behavior when increasing the number of subdomains. In this case, DDM preconditioners are interesting alternatives to Calderón preconditioning because they are less expensive to compute since only local computations are needed while Calderón preconditioning requires a global mass matrix, its inversion and a matrix-vector product with another BEM matrix. They are also more likely to scale better thanks to their parallel nature, and they give similar performances in terms of iterations according to our results.

We now conduct a 2D test about weak scaling. We consider the following array of mesh sizes: $h = [0.0005, 0.001, 0.002, 0.004, 0.008]$ that will be used to discretize the geometry shown in Figure 5.3. For each mesh size $h[i]$, we obtain a number of degrees of freedom $N[i]$ with $N = [48\,000, 24\,000, 12\,000, 6000, 3000]$. We associate a number of subdomains $n[i]$ to each mesh such that $N[i]/n[i] = 750$. The resulting array of number of subdomains is $n = [64, 32, 16, 8, 4]$. The results about the number of iterations are given Figure 5.8. The same remarks as for the strong scaling can be formulated here.

Following these two tests, the best coarse space is the GenEO stiffness coarse space, it generally yields fewer iterations and it is also less expensive to compute than the others due to the fact that its associated local matrix $\mathbf{K}_p$ is sparse.

5.4.4 Results in 3D

We only add one layer of elements to the subdomains for the overlap. As for the 2D strong scaling, we take a fixed τ, here equal to 5, for the GenEO stiffness coarse space, and we take the same number of eigenvectors in the local eigenproblems to build the coarse component of the other coarse spaces.

To test the strong scaling, we take a mesh size of 0.039 so that we obtain a number of degrees of freedom N equal to 104 738. We solve the linear system associated with our test case and preconditioned with the coarse spaces we introduced earlier using several numbers of subdomains: $\{32, 64, 128, 256, 512\}$. The resulting numbers of iterations are shown in Figure 5.10. We observe the same behavior as in 2D. Note that here the number of iterations without preconditioner is 156 with CG and 101 with GMRes.

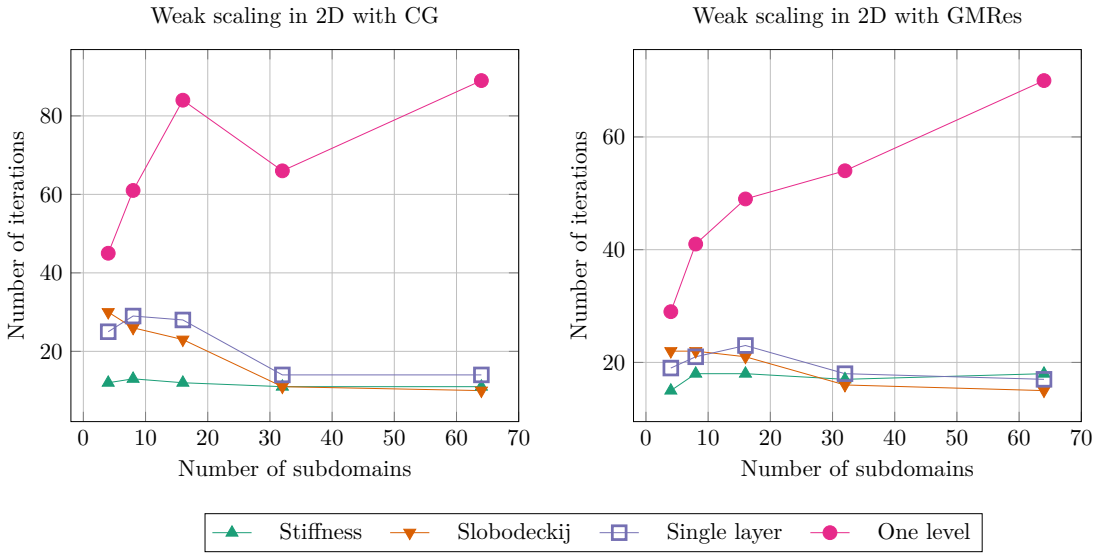


Figure 5.8 – Weak scaling to solve Equation (5.16) with the 2D geometry described in Figure 5.3 and the hypersingular operator.

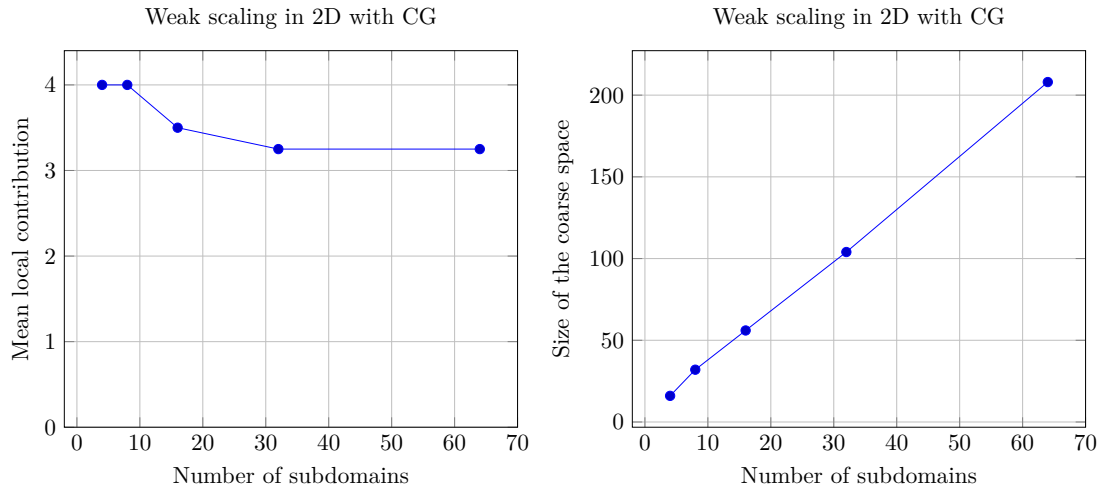


Figure 5.9 – Mean local contribution to coarse space and size of the latter for 2D weak scaling.

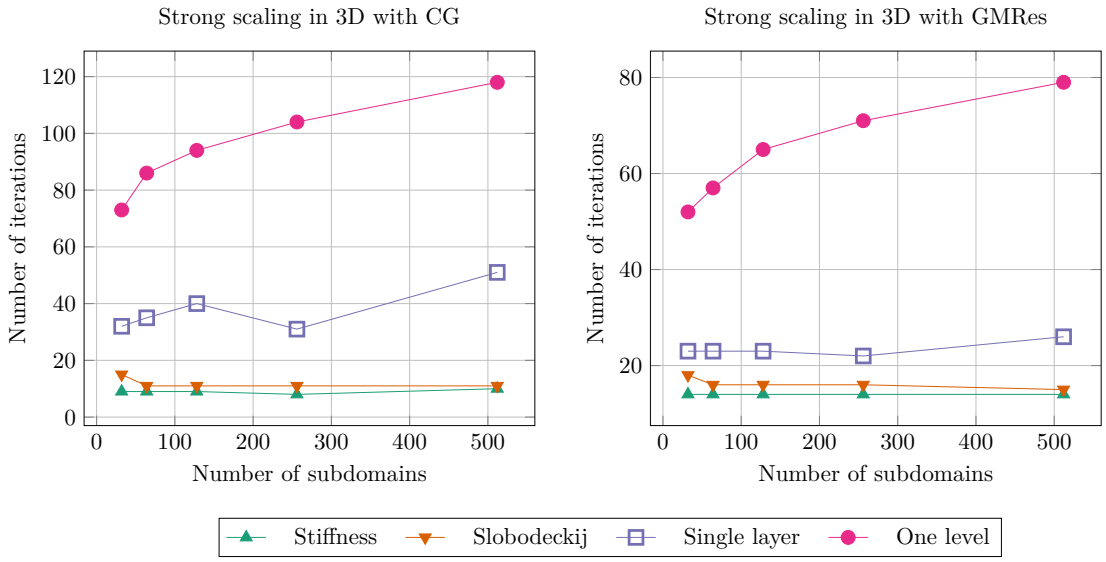


Figure 5.10 – Strong scaling to solve Equation (5.16) with the 3D geometry described in Figure 5.4 and the hypersingular operator.

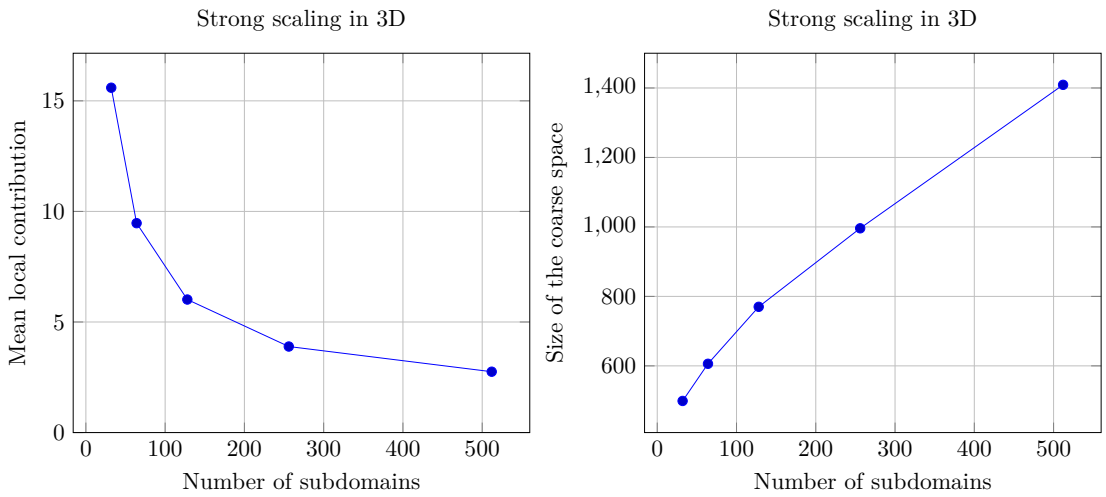


Figure 5.11 – Mean local contribution to coarse space and size of the latter for 3D strong scaling.

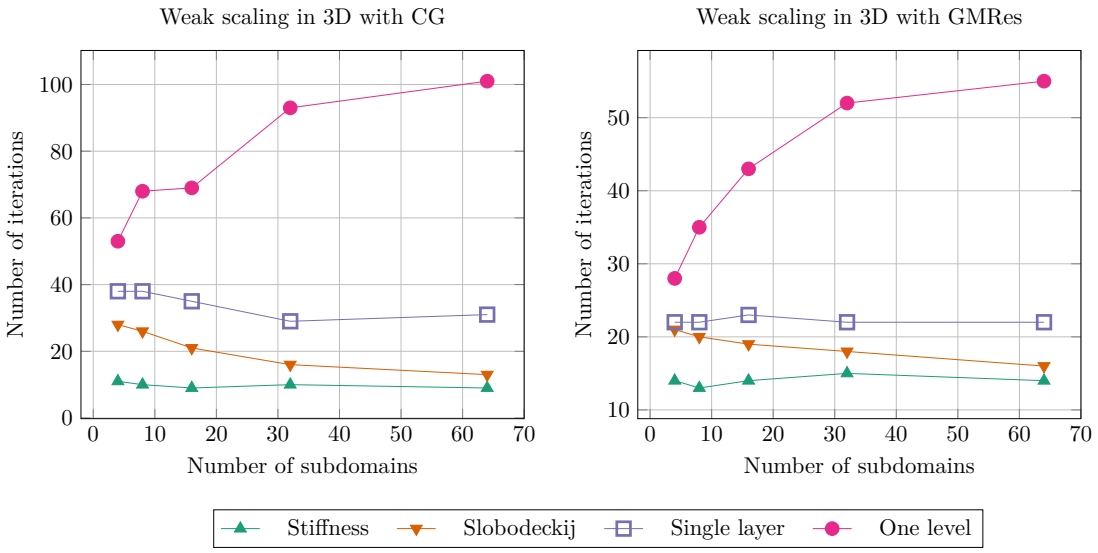


Figure 5.12 – Weak scaling to solve Equation (5.16) with the 3D geometry described in Figure 5.4 and the hypersingular operator.

We now present a 3D test about weak scaling, we consider the following mesh sizes: $h = [0.05, 0.07, 0.1, 0.15, 0.2]$. For each mesh size $h[i]$, we obtain a number of degrees of freedom $N[i]$ with $N = [63\,360, 33\,200, 15\,922, 7201, 3846]$. We associate a number of subdomains $n[i]$ to each mesh such that $N[i]/n[i] \simeq 1000$. Thus, the array of number of subdomains is $n = [64, 32, 16, 8, 4]$. The resulting number of iterations are given in Figure 5.12. The same remarks as for the strong scaling hold here. Notice that the 3D weak scaling is approximated contrary to the 2D case.

We observe that the GenEO stiffness coarse is again the most efficient, it generally yields fewer iterations and it is less expensive to compute because its associated local matrix $\mathbf{K}_p$ is sparse.

5.5 Concluding remarks and perspectives

We have adapted the GenEO approach to precondition the BEM approximation of the hypersingular operator for symmetric positive definite cases. We introduced several coarse spaces, and we have tested them numerically with Dissipative Helmholtz on problems of reasonable size (up to 100 000 degrees of freedom) on a closed surface. The most efficient coarse space from our numerical experiments is the GenEO stiffness coarse space defined by Theorem 5.22.

As a result of this work, several developments can be carried out. Some interesting numerical experiments have not been realized because of the time it would have taken to implement them. For example, screen problems for the hypersingular operator implies that we need to be able to remove the nodes on the boundary of the surface Γ so that $\mathcal{V}_h \subset \widetilde{H}^{1/2}(\Gamma)$. Another alternative would be to "virtually" remove them as homogeneous Dirichlet conditions in FEM, using pseudo-elimination or exact penalization. But it is not clear how the former will interact with hierarchical matrices, while the latter is difficult to handle in the coarse space. A study of these different approaches would be appealing for numerical applications.

Another possible development would be to adapt our work to vector valued problems such

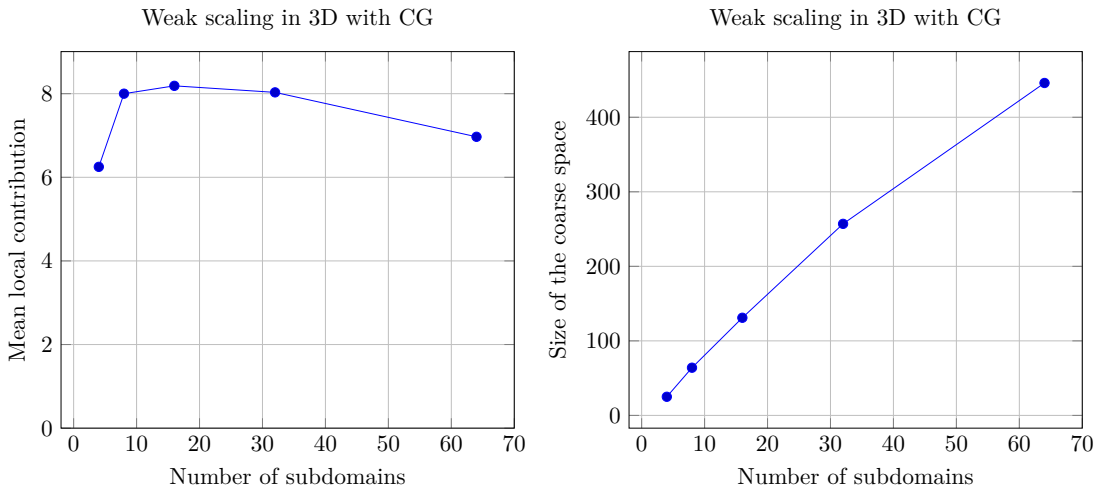


Figure 5.13 – Mean local contribution to coarse space and size of the latter for 3D weak scaling.

as linear elasticity (see [81]). It seems reasonable to think that we could extend our analysis as the authors of [157] extended theirs to Lamé crack problems. We focused on scalar problems so far mainly because we had not the tools to test numerically vector valued problems.

We also only considered the h-version of BEM, i.e. the degree of the polynomials is fixed but we can refine the mesh, mainly besause we only had $\mathbb{P}_1$ Lagrange discretization. Another popular approach is the hp-version where the degree of the polynomials also varies. This is particularly useful to correctly approximate the solution on open surfaces despite its singularity near the boundary. We refer to [56, Section 2.5] for more details about these singularities. In any case, notice that the bounds given in Theorems 5.20 and 5.24 are independent of the discretization, while, as stated in Remark 5.25, the bound given in Theorem 5.22 depends on the polynomial degree via the inverse inequality Lemma 5.21. But recent results [8, Equation (3.3)] gave a more precise inverse inequality where the dependence on the polynomial degree is explicit, so that we could just change the weight in the definition of $\mathbf{K}_p$ to take into account the varying polynomial degree. Then, it would remain to observe how the size of the coarse space changes.

Finally, we make two general remarks, first on the preconditioning of the weakly singular operator, and then on preconditioning the Helmholtz equation.

5.5.1 Preconditioner for the weakly singular operator

Naturally, after adapting the GenEO approach to the hypersingular operator, we wanted to do the same for the weakly singular operator. The analysis is actually more difficult because the energy space is then $\widetilde{H}^{-1/2}(\Gamma)$.

Concerning the continuity of $\mathcal{R}_{\mathrm{ASM},2}$, notice that the main tool is how we upper bounded the $\widetilde{H}^{1/2}(\Gamma)$ in Lemma 5.12 using Lemma 5.7 and the fact that the local term $\|u_h^p\|_{\widetilde{H}^{1/2}(\Gamma_p)}$ is equal to $\|\mathrm{R}_p^T u_h\|_{\widetilde{H}^{1/2}(\Gamma)}$. Then, we can use several norm definitions for $H^{1/2}$ to define the dual norm on $\widetilde{H}^{-1/2}$:

- with the Sobolev-Slobodeckij norm as we did so far, we can prove a similar inequality to Lemma 5.7 for the $\widetilde{H}^{-1/2}(\Gamma)$ norm using duality and Lemma 5.8. But then, we do not

know how to relate $\|u_h^p\|_{\widetilde{H}^{-1/2}(\Gamma_p)}$ and $\|\mathrm{R}_p^T u_h\|_{\widetilde{H}^{-1/2}(\Gamma)}$ independently of the size of the subdomain.

- with interpolation theory, we can prove a similar lemma to Lemma 5.8, so that again, we can prove a similar inequality to Lemma 5.7 by duality. But again, we were not able to relate $\|u_h^p\|_{\widetilde{H}^{-1/2}(\Gamma_p)}$ and $\|\mathrm{R}_p^T u_h\|_{\widetilde{H}^{-1/2}(\Gamma)}$ independently of the size of the subdomain.

In other words, with our choice of norms, we need to know how to relate a local norm $\|u_h^p\|^2_{\widetilde{H}^{-1/2}(\Gamma_p)}$ with the global norm of its extension by zero $\|\mathrm{E}_p(u_h^p)\|^2_{\widetilde{H}^{-1/2}(\Gamma)}$ to be able to use the positive definiteness of a. Thus, the difficulty is similar to the one in the analysis of GenEO single layer (see Appendix B).

An important tool to prove the stable decomposition was based on how we lower bound the global norm by a sum of local terms. In the case of the hypersingular operator, we introduced several approaches that we can try to adapt to the weakly singular operator.

- Lemma 5.8 comes from a direct computation on the Sobolev-Slobodeckij norm. By duality, we can prove a similar lemma for $-1/2$ instead of $1/2$ using Lemma 5.7. But then there is no analytical expression for $\|u|_{\Gamma_p}\|_{H^{-1/2}(\Gamma_p)}$ to compute the associated matrix.

- Lemma 5.9 relies on the continuous injection between $L^2(\Gamma)$ and $\widetilde{H}^{1/2}(\Gamma)$. There is nothing similar for $\widetilde{H}^{-1/2}(\Gamma)$.

- Similarly to GenEO single layer, we could approximate $\|u|_{\Gamma_p}\|_{H^{-1/2}}$ using the inverse of the local hypersingular operator $\mathrm{W}_p : \widetilde{H}^{1/2}(\Gamma_p) \to H^{-1/2}(\Gamma_p)$. But then, we would need a dual mesh to compute the discretization of its inverse because usually if $\mathcal{V}_{h,p} \subset \widetilde{H}^{-1/2}(\Gamma_p)$ then $\mathcal{V}_{h,p} \not\subseteq \widetilde{H}^{1/2}(\Gamma_p)$.

- There is a similar result to Lemma 5.10 for the $H^{-1/2}$ norm given [65] for the h-version and generalized in [8, Equation (3.4)] for the hp-version.

Thus, the most practical localization technique is the one given by the following lemma:

Lemma 5.27. *Let $u_h \in \mathcal{P}^m(\mathcal{T}_h)$, we have*

$$C'_{\mathrm{ad}} \sum_{p=1}^{n} \|h_{\mathcal{T}}^{1/2} u_h|_{\Gamma_p}\|^2_{L^2(\Gamma_p)} \leqslant k_1 \|u_h\|^2_{\widetilde{H}^{-1/2}(\Gamma)},$$

where C'_{ad} depends only on Γ, the shape-regularity of $\mathcal{T}_h$ and m.

Proof. According to [65], we have

$$\|u_h\|^2_{\widetilde{H}^{1/2}(\Gamma)} \geq C'_{\mathrm{ad}} \|h_{\mathcal{T}}^{1/2} u_h\|^2_{L^2(\Gamma)}.$$

Then, the L^2-norm is local, so that we can decompose it like in Lemma 3.22. □

This last lemma motivates a GenEO-type preconditioner as introduced in Section 5.3 with $(\mathbf{B}_p)_{i,j} = \int_{\Gamma_p} h_{\mathcal{T}} \varphi_{\sigma_p(i)} \varphi_{\sigma_p(j)}$, similarly to Equation (5.13).

Because of the technical difficulties mention before, we cannot carry the full analysis of a GenEO-type preconditioner for the weakly singular operator. But we can still test numerically its efficiency. Using the same test case as described in Section 5.4.1 but where f is used as a Dirichlet condition. We first compute the spectra given in Figure 5.14 from the generalized

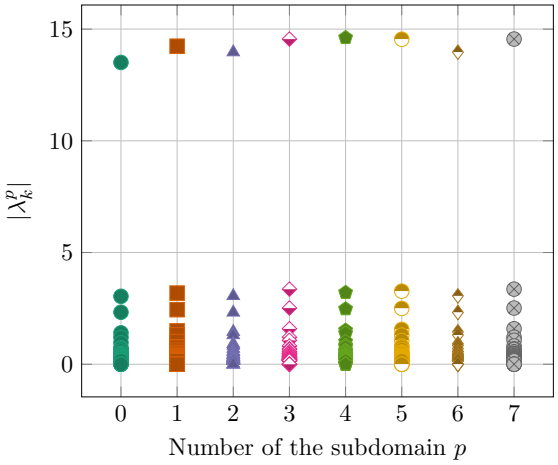


Figure 5.14 – Eigenvalue distributions for the generalized eigenproblem defined Equation (5.8) with eight subdomains and the weakly singular operator.

eigenproblem defined with $\mathbf{B}_p$ and eight subdomains. We observe that we can discriminate a few eigenvalues to be put in the coarse space.

We carry a strong scaling test similarly to the one done in Section 5.4.3, the results are given in Figures 5.15 and 5.16. We also added results where the coarse space is not used additively as in Equation (5.5) for the hypersingular operator:

$$\mathbf{R}_0^T(\mathbf{R}_0\mathbf{A}_{\mathrm{V}}\mathbf{R}_0^T)^{-1}\mathbf{R}_0 + (\mathbf{I}_{\mathrm{d}} - \mathbf{P}_0)\sum_{p=1}^{n}\mathbf{R}_p^T(\mathbf{R}_p\mathbf{A}_{\mathrm{V}}\mathbf{R}_p^T)^{-1}\mathbf{R}_p(\mathbf{I}_{\mathrm{d}} - \mathbf{P}_0) \qquad \text{Balanced,} \tag{5.18}$$

$$\mathbf{R}_0^T(\mathbf{R}_0\mathbf{A}_{\mathrm{V}}\mathbf{R}_0^T)^{-1}\mathbf{R}_0 + \sum_{p=1}^{n}\mathbf{R}_p^T(\mathbf{R}_p\mathbf{A}_{\mathrm{V}}\mathbf{R}_p^T)^{-1}\mathbf{R}_p(\mathbf{I}_{\mathrm{d}} - \mathbf{P}_0) \qquad \text{Deflated,} \tag{5.19}$$

where $\mathbf{P}_0$ is the a_{V}-orthogonal projection on the coarse space. It can be shown that $\mathbf{P}_0 = \mathbf{R}_0^T(\mathbf{R}_0^T\mathbf{A}_{\mathrm{V}}\mathbf{R}_0)^{-1}\mathbf{R}_0\mathbf{A}_{\mathrm{V}}$. We refer to [45, Section 7.5] and [152] for more details on alternatives to Equation (5.5).

The number of iterations without preconditioner is about 420 with CG and 128 with GMRes. The results are given in Figure 5.15. We see that the difference of efficiency between the one-level and the coarse space suggested by the localisation in Lemma 5.27 is smaller that in the tests we did with the hypersingular. With the hypersingular operator, the GenEO stiffness coarse space and CG, the number of iterations for the strong scaling 2D test from Section 5.4.3 was around 11, while it increased from 57 to 114 with the one-level preconditioner. So the number of iterations was at least 4.75 lower using the GenEO stiffness coarse space. For a similar test case with the weakly singular operator, the two curves obtained with CG in Figure 5.15 are relatively close to each other, the one-level is even slightly more efficient for 64 and 128 subdomains, but in any case the difference is of the order of the unit. The additive approach is quite close to the one-level, and we recover a quasi-constant number of iterations with Balanced and Deflated. In this case, Calderón preconditioning becomes $\mathbf{P}_{\text{Calderón},\mathbf{V}_h} = \mathbf{M}_h^{-1}\mathbf{W}_h\mathbf{M}_h^{-1}$ and then, we obtain 17 iterations with CG and 10 iterations with GMRes. We obtain a few iterations more with the one-level and additive preconditioners, but it is still of the same order, and one iteration of $\mathbf{P}_{\text{Calderón},\mathbf{V}_h}$ implies two matrix vector products, so we expect it to be more expensive. We

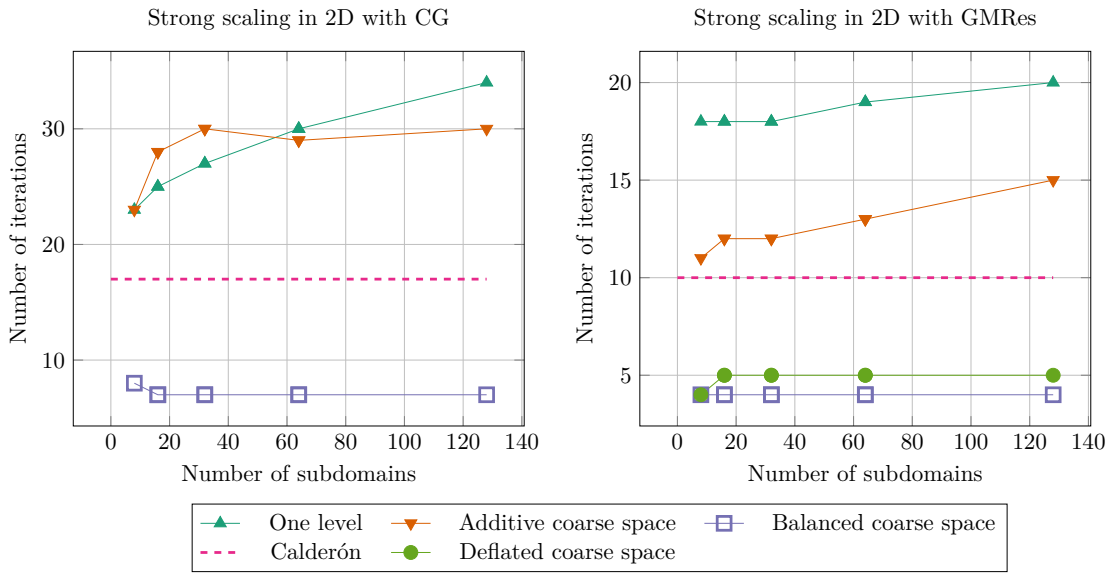


Figure 5.15 – Number of iterations for a strong scaling to solve Equation (5.16) with the 2D geometry described in Figure 5.3 and the weakly singular operator.

show in Figure 5.17 the number of matrix vector products for this strong scaling experiment. We see that the DDM preconditioners usually require less matrix vector products than Calderón preconditioning.

We also tried a more difficult test case solving the Laplace equation in 3D with a Dirichlet condition equal to one and the geometry described in Figure 5.18. We used 121 996 degrees of freedom and we refined around the boundary. The number of iterations without preconditioner is 471 with GMRes and it does not converge in less than 1000 iterations with CG. The results are shown in Figures 5.19 and 5.20 and even with this more difficult test case, it seems that the one-level is already a good preconditioner and there is not a lot of room for improvement contrary to Neumann problems. In both test cases, the variations in the number of iterations for the one-level are small (no more than 10 at worst), so that it may not worth adding a coarse space. We did not try Calderón preconditioning here because it is known to not work very well on screens and requires a special treatment with special BIO, see [97, 98, 99].

5.5.2 Preconditioner for Helmholtz equation

Another natural question which comes to mind after Section 5.3 is how to precondition problems that are not positive definite such as the Helmholtz equation. In this case, the GenEO coarse space is no longer relevant in FEM or in BEM, because the analysis that relies on norm properties does not hold anymore. Other approaches have been introduced, for example solving a coarse problem on a coarse mesh or special conditions at the interface of the subdomains (see [66] and [68] for recent results and a literature review).

We were interested in how the one-level behaved for BEM matrices. Let $u_{\text{inc}} = e^{i\kappa \mathbf{x}\cdot\mathbf{d}}$ be a plane wave with $\mathbf{d}$ the direction of propagation and $\kappa = \frac{2\pi k}{c_0}$ where $c_0 = 299\,792\,548\,\text{m}\,\text{s}^{-1}$ is the speed of light and k is the frequency.

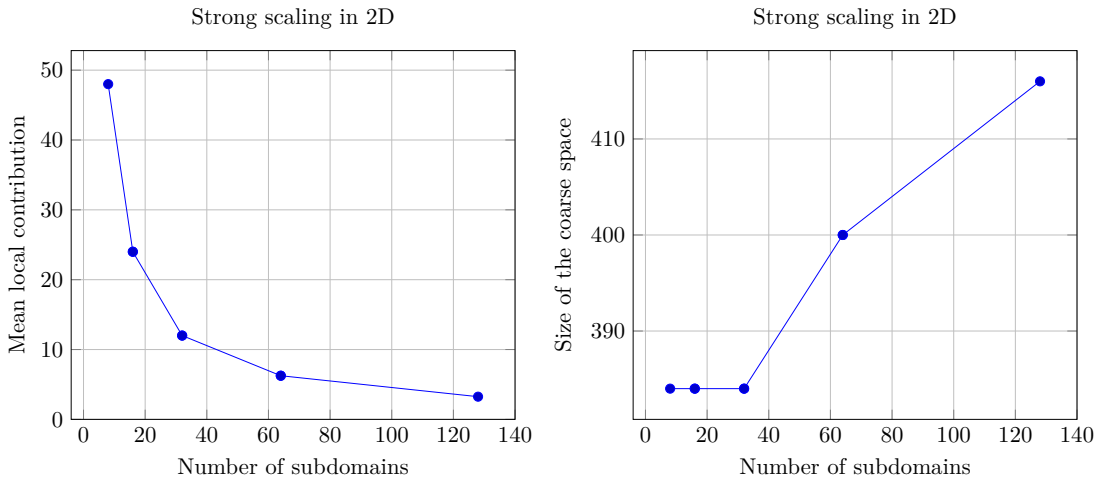


Figure 5.16 – Mean local contribution to coarse space and size of the latter for 2D strong scaling.

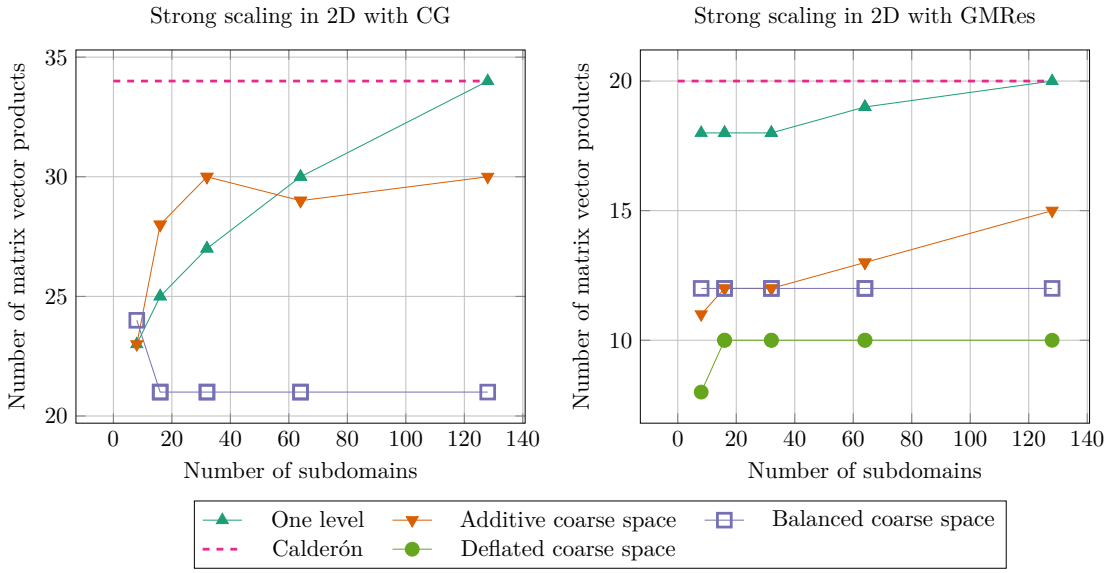


Figure 5.17 – Number of matrix vector products for a strong scaling to solve Equation (5.16) with the 2D geometry described in Figure 5.3 and the weakly singular operator.

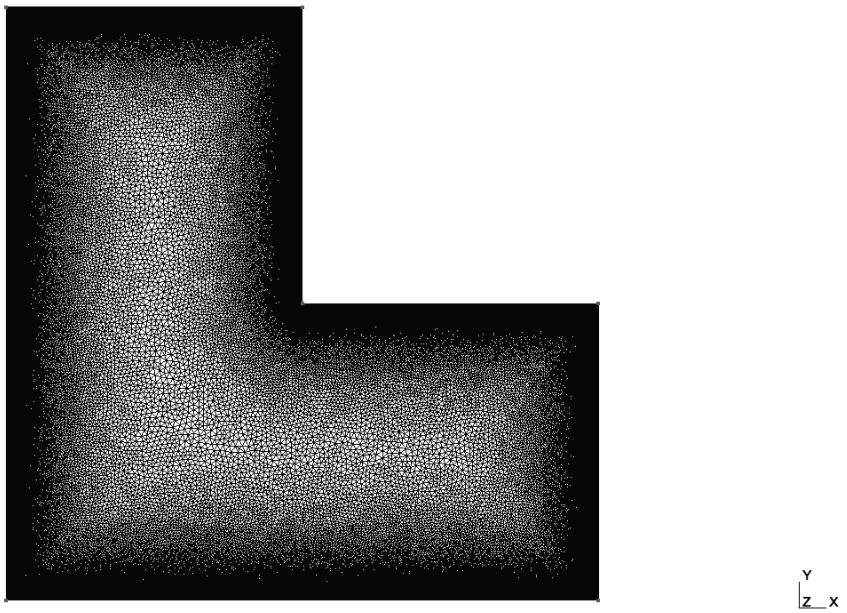


Figure 5.18 – Mesh of a L-shaped screen in 3D with 121 996 nodes and refined near its boundary.

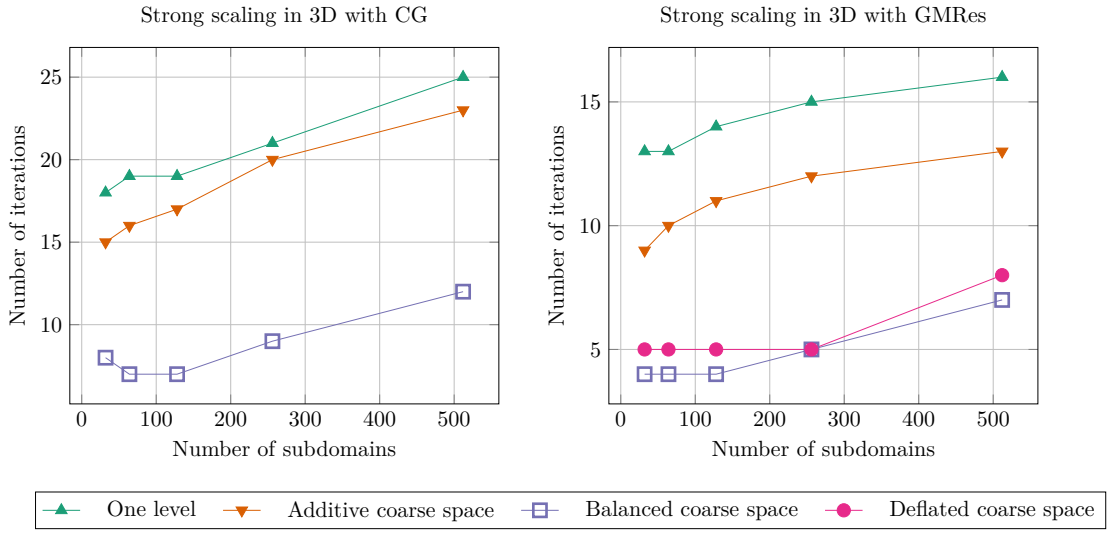


Figure 5.19 – Strong scaling to solve Equation (5.16) with the 3D geometry described in Figure 5.18 and the weakly singular operator.

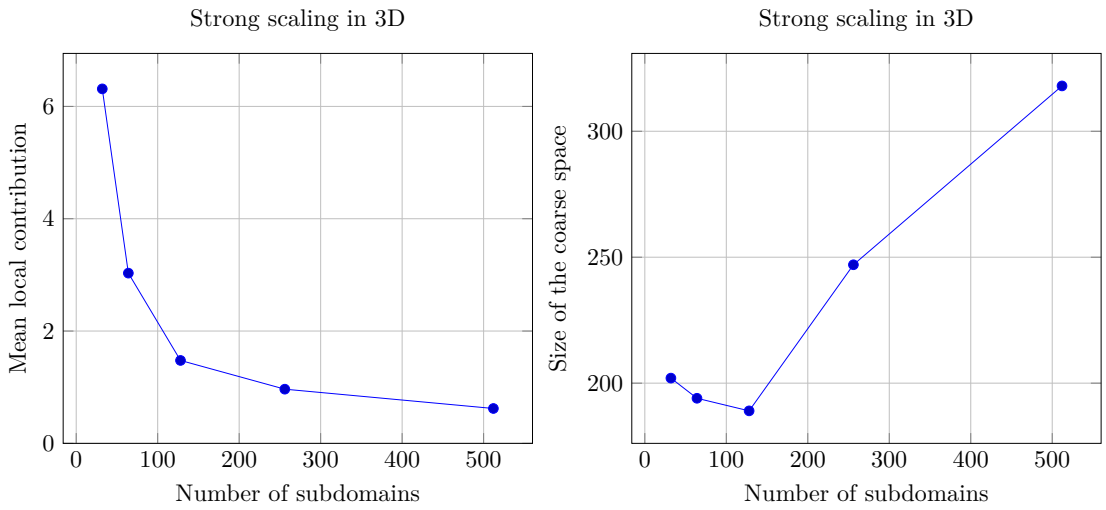


Figure 5.20 – Mean local contribution to coarse space and size of the latter for 3D strong scaling.

First, we solved a 2D problem with homogeneous Neumann condition on the total field. Similarly to Equation (2.12), the equation reads: Find $v \in \widetilde{H}^{1/2}(\Gamma)$ such that

$$\langle \mathrm{W}_{\mathrm{L_H}}(v), w\rangle_{H^{-1/2}(\Gamma)\times\widetilde{H}^{1/2}(\Gamma)} = -\langle \gamma_N(u_{inc}), w\rangle_{H^{-1/2}(\Gamma)\times\widetilde{H}^{1/2}(\Gamma)}, \quad \forall w \in \widetilde{H}^{1/2}(\Gamma), \tag{5.20}$$

where Γ is an object with a cavity as described in Figure 5.21, $\mathrm{L_H}(u) = -\Delta u + \kappa^2 u$ and $\mathbf{d} = (-1/\sqrt{2}, -1/\sqrt{2})$. Figure 5.21 shows the geometry and the real part of the computed solution for $k = 1\,\mathrm{GHz}$. We have $h = 0.001$ so that there are 16 000 degrees of freedom. Figure 5.22 shows the results of a strong scaling with this test case. For Calderón preconditioning, we obtain 111 iterations with $k = 1\,\mathrm{GHz}$, 355 iterations with $k = 3\,\mathrm{GHz}$, 536 iterations with $k = 5\,\mathrm{GHz}$, and we do not show the number of iterations without preconditioner when it does not converge in less than 1000 iterations. We see that the number of iterations with the one-level preconditioner is increasing, and this increase is more important when the frequency is greater. But in any case, in terms of matrix vector product, the efficiency of Calderón preconditioning, shown by multiplying by two the number of iterations, seems to deteriorate faster.

Then, we take a classical geometry, namely the COBRA cavity described in [114] and shown in Figure 5.23. We solved a screen problem with homogeneous Dirichlet condition as described in Equation (2.18). This time $\mathbf{d} = (1, 0, 0)$ so that u_{inc} is a plane wave normally incident upon the cavity aperture. The equation then reads: Find $q \in \widetilde{H}^{-1/2}(\Gamma)$ such that

$$\langle \mathrm{V}_{\mathrm{L_H}}(q), r\rangle_{H^{1/2}(\Gamma)\times\widetilde{H}^{-1/2}(\Gamma)} = -\langle \gamma_D(u_{inc}), r\rangle_{H^{1/2}(\Gamma)\times\widetilde{H}^{-1/2}(\Gamma)}, \quad \forall r \in \widetilde{H}^{-1/2}(\Gamma), \tag{5.21}$$

where Γ is the COBRA cavity. We always keep 10 nodes per wavelength in our geometry, see Figure 5.23. We do a strong scaling test for $k = 5\,\mathrm{GHz}$, $k = 10\,\mathrm{GHz}$ and $k = 17.5\,\mathrm{GHz}$. The results are given in Figure 5.24 and the total field $\mathrm{SL}(q) + u_{\mathrm{inc}}$ for $k = 5\,\mathrm{GHz}$ is shown in Figure 5.25.

We observe that the variation in the number of iterations for the one-level is small compared to what is expected from FEM, similarly to what was observed in the previous section for Dissipative Helmholtz. For $k = 5\,\mathrm{GHz}$ ($N = 3346$), it increases by about 20 iterations, it increases by about

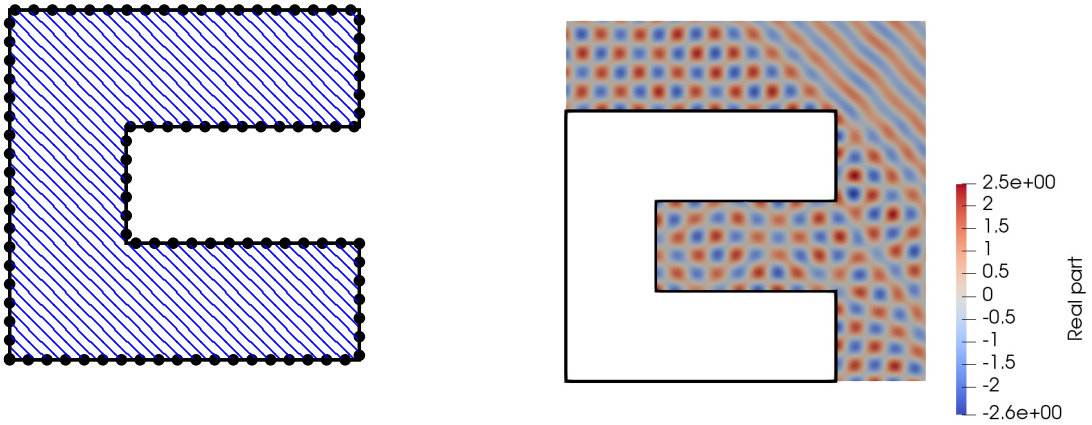


Figure 5.21 – 2D geometry used to solve Equation (5.20) with the hypersingular operator and real part of the computed solution.

100 iterations for $k = 10\,\text{GHz}$ ($N = 16\,266$) and by about 200 iterations for $k = 17.5\,\text{GHz}$ ($N = 92\,664$). Of course, the problem is more difficult when the frequency increases, but we note that the one-level preconditioner is quite cheap and parallel, so that it can be already efficient enough for a range of frequencies. Like the 3D test case in Section 5.5.1, we do not make the comparison with Calderón preconditioning because for screens it requires a special treatment.

We would like to point out that this case is more difficult to analysis and to solve numerically compared to symmetric positive definite problems for the following reasons:

- When k increases, the solution becomes more oscillatory which put a constraint on the mesh to be able to represent accurately such a solution.
- When k increases, the problem becomes more and more indefinite, so that usual preconditioners for symmetric positive definite problems are less and less efficient. It is exactly what we have seen in this section with the one-level DDM preconditioner.
- Because the equation is indefinite, CG is not relevant, and GMRes is used instead. But the convergence proof is then more difficult.

In a FEM context, one of the first approach to adapt Schwarz method to precondition indefinite problems was introduced in [20] (see also [153, Chapter 11]) and the analysis relies on considering the Helmholtz equation as a symmetric positive definite operator with a compact perturbation. This approach has been adapted to BEM in [149, 85, 150].

Since then, other approaches have been developed for usual PDE. There seems to be several classes of methods. One idea introduced in [10] was to precondition the Helmholtz equation by solving the Laplace equation. A variant proposed in [117, 116] was to properly perturb the associated matrix to define a preconditioner. These approaches were then pushed forward in [109, 52] where they solved another Helmholtz equation where κ^2 was modified to be positive definite to precondition the original problem. Then, this approach was usually called “shifted Laplacian” and has been studied in [51, 50, 62, 54] for example.

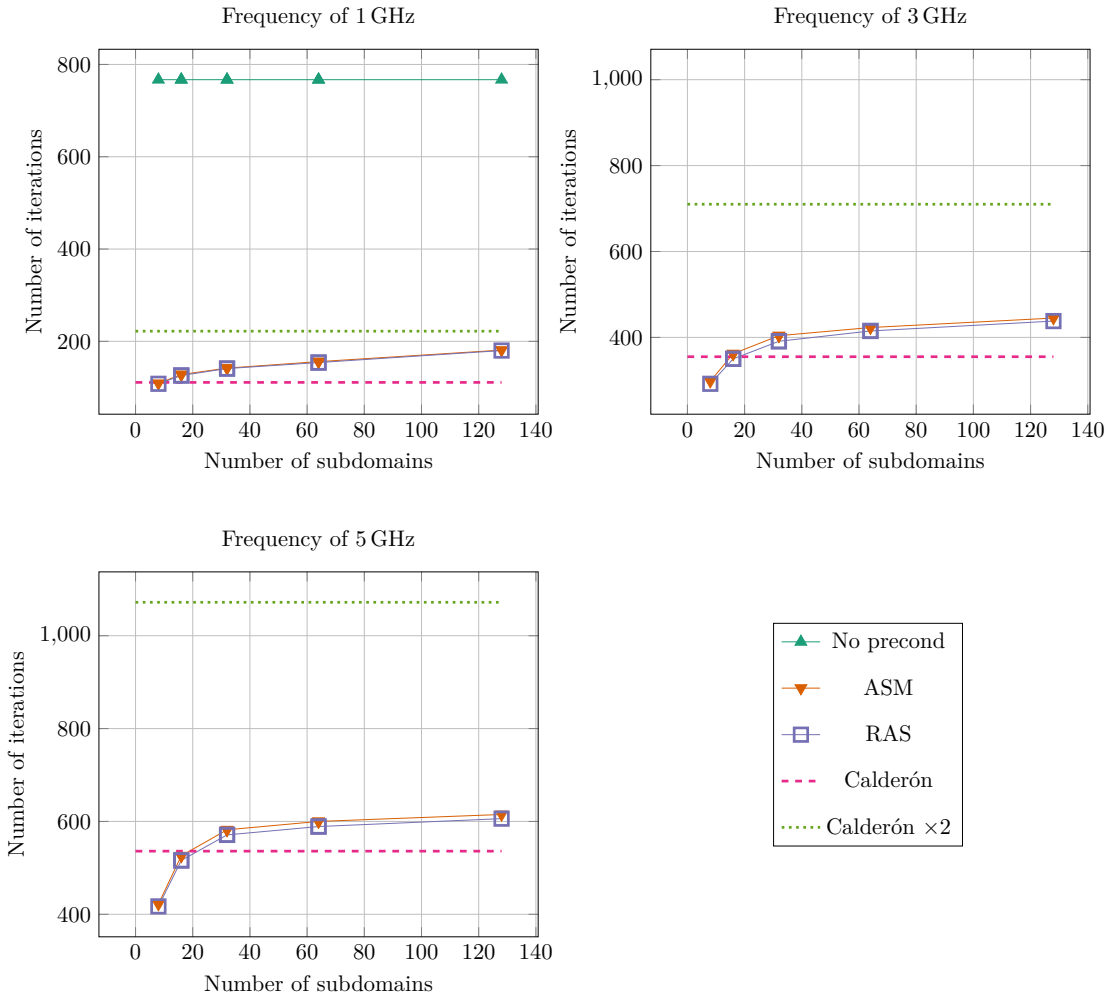


Figure 5.22 – Strong scaling to solve Equation (5.20) with the 2D geometry described in Figure 5.21 and the hypersingular operator.

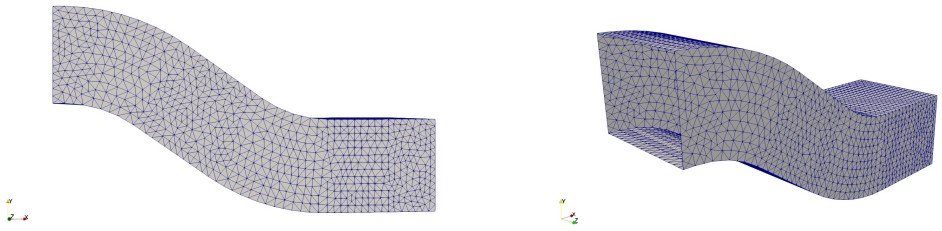


Figure 5.23 – Mesh of the COBRA cavity for 5 GHz with ten points per wavelength.

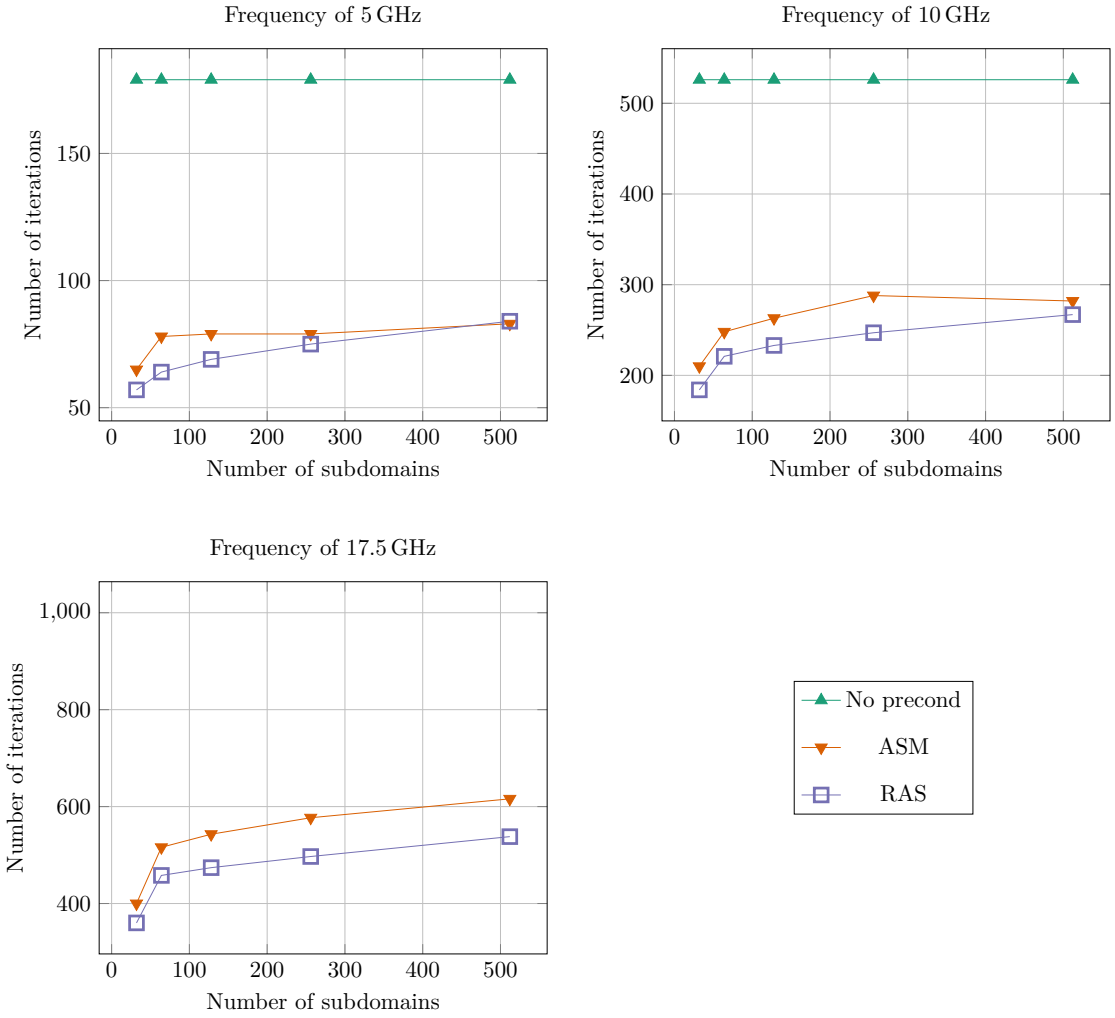


Figure 5.24 – Strong scaling to solve Equation (5.21) with the geometry described in Figure 5.23 and the weakly singular operator.

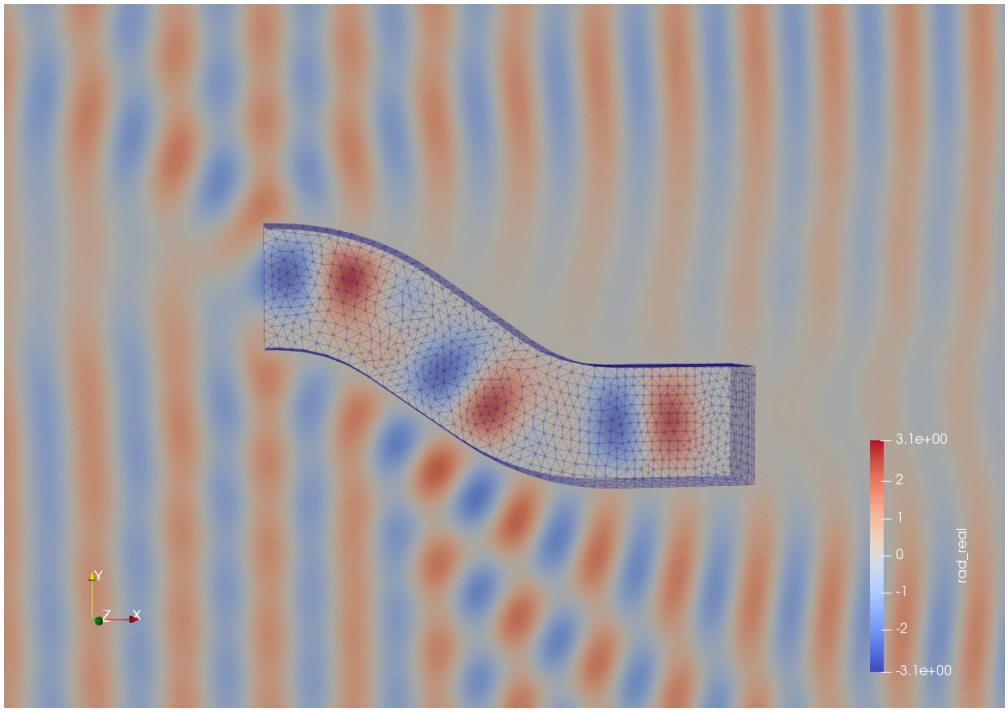


Figure 5.25 – Total field solution of Equation (5.21) for 5 GHz with five points per wavelength.

Another approach is to use DDM methods where the subproblems are solved sequentially, namely sweeping preconditioners [48, 49], single trace layer [151], the source transfer methods [24] and polarized trace method [164], which are all related to the approximation of the Optimal Schwarz Method presented in Section 3.1.2 (see [59] for a unified presentation).

Finally, DDM preconditioners have also been proposed in [55, 58, 106, 107, 40, 103, 66] with special conditions at the interface and/or a coarse space. Convergence proofs for these methods are usually an open problem but recent developments have been made in [66, 67, 68].

An interesting development would be to explore if ideas or techniques from the methods developed in a FEM context as the ones we cited previously can be adapted to BEM matrices, like we adapted GenEO preconditioners to BIE.

CHAPTER 6

Hierarchical matrices and Htool

Outline of the current chapter

As we have seen in Section 2.4.5, BEM matrices are dense due to the non-local nature of the integral kernels introduced in Section 2.3. This means that using classical direct solvers is too expensive for BEM matrices. Besides, the storage and the cost of assembling dense BEM matrices are both quadratic. Iterative solvers should also be expensive because the complexity associated with a matrix-vector product is quadratic.

To tackle these issues, several techniques have been developed: *panel clustering* [78, 79], *Fast Multipole Method* (FMM) [134, 70, 42, 43], *hierarchical matrices* ($\mathcal{H}$ and $\mathcal{H}^2$-matrices) [74, 76, 77, 17, 12, 75] *Adaptive Cross Approximation* (ACA) [11, 13, 18, 12], *Sparse Cardinal Sine Decomposition* (SCSD) [4] and *wavelet compression techniques* [5]. All these techniques give an approximate representation of the matrix that allows storage and matrix-vector product with almost linear complexity. Let us emphasize that they do not require the original dense matrix, in practice the approximated representation of a BEM matrix is directly assembled, so that we never need to perform a step whose complexity is more than quasi-linear.

With these alternative representations of BEM matrices, iterative solvers such as CG and GMRes can be a good choice to solve the linear systems associated with BEM since the matrix-vector product has a reasonable cost. Another advantage of iterative solvers is that they are independent of the underlying structure of the approximate matrix in theory, unlike techniques using direct methods and compression like hierarchical LU decomposition [75] and the fast direct solver described in [121, 71].

Another issue comes from the fact that the number of iterations needed in iterative solvers is strongly linked to the spectral properties of these matrices. For example, in the case of the single layer for the Laplace problem, the condition number $\kappa(\mathbf{A}_{\mathrm{V},h}) := \|\mathbf{A}_{\mathrm{V},h}\|_2\|\mathbf{A}_{\mathrm{V},h}^{-1}\|_2$ typically deteriorates like $O(h^{-1})$ [139, Section 4.5], and we have the same behavior for the hypersingular operator. This means that the number of iterations when using CG will increase when refining the mesh size as the square root of h^{-1}.

That is why we introduced in Chapter 5 DDM preconditioners for symmetric positive definite BEM matrices that bound the condition numbers of these matrices, so that iterative solvers should be robust to variations of the mesh size h and the number of subdomains n. DDM preconditioners are particularly well-suited for distributed memory parallel environments, which is usually the paradigm used for HPC. If, in theory, preconditioning is independent of the compression techniques, in practice we still need a distributed memory parallel implementation of a compression technique adapted to DDM preconditioners. For example, it seems appropriate that the parallelism of the compression techniques should be linked to the domain decomposition to lower the exchange of information between the processors.

With this in mind, we decided to use hierarchical matrices because it is a black-box method. The compression algorithm is independent of the nature of the underlying linear operator, unlike FMM for BEM matrices for example. This feature goes well with GenEO preconditioners that are also black-box, in the sense they do not depend on the underlying equation as long as it is symmetric positive definite. The resulting solver should be easy to use for the end user.

This led us to implement Htool[1], an open source C++ library for hierarchical matrices that offers a way to use DDM preconditioners with MPI and OpenMP. It depends on several libraries

- a BLAS [15] implementation to perform algebraic operations (dense matrix-matrix or matrix-vector operations),
- LAPACK [6] to solve dense generalized eigenvalue problems and perform LU-decomposition,
- HPDDM [104] to use iterative solvers.

The user needs to provide a function that generates the coefficient $\mathbf{A}_{i,j}$ from the pair of indices (i, j), where $\mathbf{A}$ is the matrix to be compressed. Then, Htool can assemble the hierarchical matrix associated with $\mathbf{A}$ and do parallel $\mathcal{H}$-matrix-vector and $\mathcal{H}$-matrix-matrix products. It also allows solving linear systems associated with the approximation of $\mathbf{A}$ using DDM preconditioners. The setup of the DDM preconditioners for BEM matrices is shared between HPDDM and Htool.

[1] https://github.com/PierreMarchand20/htool

We would like to point that several C libraries implementing hierarchical matrices already existed, for example HLIB[2] and its successor H2Lib[3], but they usually do not support distributed memory parallelism, which is more relevant for DDM preconditioners. Besides, our implementation is by no means the most optimal, years of R&D led to the development of very efficient implementations (see [115] for example), but they are proprietary softwares. Htool has several advantages from our point of view.

- It is portable, it has been used on laptops and supercomputers. Actually, the library is automatically tested on Linux and macOS using g++ and clang compilers on every change.
- It is lightweight, it focuses on hierarchical matrices and how to solve the associated linear systems. We preferred to focus on its interoperability to facilitate its integration in other codes.
- It is black-box, in the sense that it can be easily embedded in other libraries as an independent module. It is already in Freefem++ [82] for instance.
- Its structure is modular, so that additions of new low-rank approximations should be easy to integrate.
- It works in parallel using MPI and OpenMP.
- It is free and open source under the MIT Licence.

In this chapter, we present the main ideas of hierarchical matrices and how we implemented them to be sufficiently efficient with DDM preconditioners for our test cases. We refer to [75, 132, 145, 115] for more details on the presentation and analysis of hierarchical matrices. We first present low-rank approximation in Section 6.1. Since BEM matrices do not have a low rank structure because of the singularity of the Green kernel (see Equation (2.8)), we introduce in Section 6.2 how to partition these matrices in admissible blocks, i.e. blocks that can be well-approximated by low-rank matrices. In Section 6.3, we introduce the library Htool and we present how we implemented the methods and algorithms presented in this manuscript. Finally, we conclude with some perspectives for Htool.

We want to acknowledge that Htool was created following the CEMRACS[4] summer school in 2016, where we had the opportunity to work on hierarchical matrices for a particular application from IFPEN (the French Institute of Petroleum). This led to the code called ElastoPhi[5] and the publication [61]. Htool was then written following what we learned from this project and parallelized with the collaboration of Pierre-Henri Tournier, CNRS research engineer at Laboratoire Jacques-Louis Lions.

6.1 Low-rank approximation

6.1.1 Low-rank matrix

Let $\mathbf{B} \in \mathbb{C}^{M \times N}$ be a dense matrix with $\mathbf{B} = (\mathbf{B}_{j,k})_{1 \leq j \leq M, 1 \leq k \leq N}$. Without any assumption, the cost of storage and the complexity of a matrix-vector product are both quadratic. Assume that

[2] http://www.hlib.org
[3] http://www.h2lib.org
[4] http://smai.emath.fr/cemracs/cemracs16/
[5] https://github.com/xclaeys/ElastoPhi

$\mathbf{B}$ can be written as follows

$$\mathbf{B} = \sum_{j=1}^{r} \mathbf{u}_j \mathbf{v}_j^T, \tag{6.1}$$

where $r \leq \min(M, N)$, $\mathbf{u}_j \in \mathbb{C}^M$,$\mathbf{v}_j \in \mathbb{C}^N$ with $1 \leq j \leq r$. Then, the cost of storage and the complexity of a matrix-vector product can be reduced to $O(r(M + N))$, which is lower that $O(MN)$ provided $r < MN/(M + N)$. In this case, we say that $\mathbf{B}$ has a *low rank* structure.

6.1.2 Singular value decomposition

Matrices encountered in applications rarely have this feature. Nevertheless, a matrix can always be written as a sum of rank-one matrices through its Singular Value Decomposition (SVD)

$$\mathbf{B} = \sum_{j=1}^{P} \sigma_j \mathbf{u}_j \mathbf{v}_j^T \tag{6.2}$$

such that the spectrum of $\mathbf{B}^*\mathbf{B}$ is equal to $(\sigma_j^2)_{j=1}^r$ with $P = \min(M, N)$ while $(\mathbf{u}_j)_{j=1}^M$ and $(\mathbf{v}_j)_{j=1}^N$ are orthogonal bases of respectively $\mathbb{C}^M$ and $\mathbb{C}^N$. By convention, we have $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_P$. Then, a natural idea is to truncate the SVD Equation (6.2) to obtain a low-rank approximation of $\mathbf{B}$. This should be a good approximation in the case $(\sigma_j)_{j=1}^P$ quickly decreases. The following theorem gives a more precise result.

Theorem 6.1 ([63, Sections 2.5.3 and 2.5.6] and [75, Lemma C.5])**.** *Let* $\mathrm{B} \in \mathbb{C}^{M\times N}$*, then*

(i) there exist unitary matrices $\mathbf{U} \in \mathbb{C}^{M\times M}$ *and* $\mathbf{V} \in \mathbb{C}^{N\times N}$ *such that*

$$\mathbf{B} = \mathbf{USV}$$

where $\mathbf{S} \in (\mathbb{R}^+)^{P\times P}$ *is a rectangular diagonal matrix with* $P = \min(M, N)$*. We denote its diagonal elements* $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_P \geq 0$*. If* $M \leq N$*, we have*

$$\mathbf{S} = \begin{pmatrix} \sigma_1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \ddots & \vdots & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & \sigma_P & 0 & \cdots & 0 \end{pmatrix},$$

where $(\sigma_j)_{j=1}^P$ *are called singular values of* $\mathbf{B}$*.*

(ii) The spectral norm of $\mathbf{B}$ *has the value* $\|\mathbf{B}\|_2 = \sigma_1$*, where* $\|\cdot\|_2$ *refers to the matrix norm induced by the vector norm* $\|\mathbf{u}\|_2^2 = (\sum_{j=1}^N |u_j|^2)$ *for* $\mathbf{u} = (u_j)_{j=1}^N \in \mathbb{C}^N$*.*

(iii) The Frobenius norm of $\mathbf{M}$ *has the value* $\|\mathbf{B}\|_{\mathrm{F}}^2 = \sum_{j=1}^P \sigma_j^2$*, where* $\|\cdot\|_{\mathrm{F}}$ *refers to the Frobenius norm given by* $\|\mathrm{B}\|_{\mathrm{F}}^2 = \sum_{j,k=1\ldots n} |\mathrm{B}_{j,k}|^2$*.*

(iv) $\mathrm{rank}(\mathbf{B})$ *is equal to* $\max\{j \,|\, \sigma_j > 0\}$*.*

We obtain directly the following corollary

Corollary 6.2. *Let* $\mathbf{B}^{(r)} \in \mathbb{C}^{M\times N}$*, the matrix obtained by truncating the SVD of* $\mathbf{B}$ *at rank* $r \leq P$*, i.e.* $\mathbf{B}^{(r)} = \sum_{j=1}^{r} \sigma_j \mathbf{u}_j \mathbf{v}_j^T$ *where* $\mathbf{u}_j$ *and* $\mathbf{v}_j$ *are respectively the* j*th column of* $\mathbf{U}$ *and* $\mathbf{V}$ *from Theorem 6.1. Then, we have*

$$\|\mathbf{B} - \mathbf{B}^{(r)}\|_2^2 = \sigma_{r+1}^2 \quad \textit{and} \quad \|\mathbf{B} - \mathbf{B}^{(r)}\|_{\mathrm{F}}^2 = \sum_{j=r+1}^{n} \sigma_j^2.$$

We deduce that, if $\mathbf{B}$ has exponentially decreasing singular values, i.e. $\sigma_r < q^r$ for some $0 < q < 1$, then we can obtain an error of ε in Frobenius norm with $r \simeq \log_q(\varepsilon)$ approximating $\mathbf{B}$ by $\mathbf{B}^{(r)}$. Another important result states that the truncated SVD is actually the best low-rank approximation possible.

Theorem 6.3 (Eckart–Young–Mirsky theorem, [46] and [63, Theorem 2.5.3])**.** *Using the notation from Theorem 6.1 and Corollary 6.2, for* $r < \mathrm{rank}(\mathbf{B})$*, we have*

$$\min_{\mathrm{rank}(C)=r} \|\mathbf{B} - \mathbf{C}\|_2 = \|\mathbf{B} - \mathbf{B}^{(r)}\|_2 = \sigma_{r+1},$$

$$\min_{\mathrm{rank}(C)=r} \|\mathbf{B} - \mathbf{C}\|_{\mathrm{F}} = \|\mathbf{B} - \mathbf{B}^{(r)}\|_{\mathrm{F}} = \sqrt{\sum_{j=r+1}^{n} \sigma_j^2}.$$

These results have been extended to every unitarily invariant norms in [126].

6.1.3 Adaptive cross approximation

Unfortunately, the SVD is quite expensive. State of the art SVD algorithms require at least $O(N^3 + M^2N + MN^2)$ operations (see [63, Section 5.4.5]) and we still need to compute the whole original matrix. Thus, it is still expensive in terms of storage and computation, even if we then obtain a cheaper matrix-vector product.

6.1.3.1 Pseudo-skeleton approximation

Another approach to obtain low-rank approximations of $\mathbf{B} \in \mathbb{C}^{M\times N}$ is to use columns and rows of the original matrix. Let us denote $\mathbf{B}(\sigma, \tau) \in \mathbb{C}^{|\sigma|,|\tau|}$ the submatrix of $\mathbf{B}$ consisting of the rows $\sigma \subseteq \{1, \ldots, M\}$ and the columns $\tau \subseteq \{1, \ldots, N\}$. We will use the notation $\mathbf{B}(\sigma, :)$ (resp. $\mathbf{B}(:, \tau)$) when $\tau = \{1, \ldots, N\}$ (resp. $\sigma = \{1, \ldots, M\}$) and we denote $\mathbf{B}(j, :)$ (resp. $\mathbf{B}(:, k)$) the jth row of $\mathbf{B}$ (resp. the kth column of $\mathbf{B}$).

Theorem 6.4 ([64, Corollary 3.1])**.** *Assume that* $\mathbf{B}$*,* $\mathbf{R} \in \mathbb{R}^{M\times N}$ *such that* $\|\mathbf{B} - \mathbf{R}\|_2 \leq \varepsilon$ *and* $\mathrm{rank}(R) \leq r$*. Then, there exists* $\sigma \subset \{1, \ldots, M\}$*,* $\tau \subset \{1, \ldots, N\}$ *and* $\mathbf{S} \in \mathbb{R}^{r\times r}$ *such that* $|\sigma| = r$*,* $|\tau| = r$ *and*

$$\|\mathbf{B} - \mathbf{B}(:, \tau)\mathbf{S}\mathbf{B}(\sigma, :)\|_2 \leq \varepsilon(1 + 2\sqrt{r}(\sqrt{N} + \sqrt{M})).$$

The matrix $\mathbf{B}(:, \tau)\mathbf{S}\mathbf{B}(\sigma, :)$ *is called* pseudo-skeleton *component of* $\mathbf{B}$*.*

In other words, if there exists a low-rank approximation of rank r for a given error ε, then there exists a pseudo-skeleton approximation.

6.1.3.2 Fully pivoted Adaptive Cross Approximation

The Adaptive Cross Approximation (ACA) algorithm introduced in [11, 13, 12] is an iterative algorithm that provides such an approximation with $\mathbf{S}$ diagonal. As in [132, Section 2.2], we first present the Fully pivoted ACA algorithm with Algorithm 1 and then its variant called Partially pivoted ACA algorithm with Algorithm 2. In our pseudo-code notation, for any vector $\mathbf{w} \in \mathbb{C}^N$, the number $\mathbf{w}(j)$ refers to the jth entry of $\mathbf{w}$.

Algorithm 1 Fully Pivoted ACA

function FULL ACA($\mathbf{B}$) ▷ We need the original dense matrix
  $r = 0$ and $\mathbf{R}^0 = \mathbf{B}$
  while (stopping criterion not satisfied) **do** ▷ see Section 6.1.3.4
    $(j_*^r, k_*^r) = \text{argmax}_{1 \leq j \leq M,\, 1 \leq k \leq N} |\mathbf{R}^r_{j,k}|$
    $w_*^r = \mathbf{R}^r_{j_*^r, k_*^r}$
    if $(|w_*^r| \neq 0)$ **then**
      $\mathbf{v}_r = \mathbf{R}^r(j_*^r, :)$ and $\mathbf{u}_r = (w_*^r)^{-1}\mathbf{R}^r(:, k_*^r)$
      $\mathbf{R}^{r+1} = \mathbf{R}^r - \mathbf{u}_r \mathbf{v}_r^T$
      $r = r + 1$
    else
      break
  return $\widehat{\mathbf{B}}^{(r)} = \sum_{j=0}^{r-1} \mathbf{u}_j \mathbf{v}_j^T$

In other words, at each iteration r we look for the coefficient (j_*^r, k_*^r) of maximal absolute value in the residual $\mathbf{R}^r$ and we define $\mathbf{u}_r$ and $\mathbf{v}_r$ respectively as the k_*^rth column and the j_*^rth rows, with the right normalization for the former so that $\mathbf{R}^{r+1} = \mathbf{R}^r - \mathbf{u}_r \mathbf{v}_r^T$ has zeros for k_*^rth column and j_*^rth row (see Lemma 6.6). The coefficients $(j_*^r, k_*^r)_r$ are usually called the *pivots*. This algorithm has several interesting properties given by the following lemmas from [19, Chapitre 4][6].

Lemma 6.5 (Exact reproduction of rank-r matrices, [19, Lemma 4.4])**.** *Let* $\mathbf{B}$ *be a matrix of rank exactly* r*. Then* $\widehat{\mathbf{B}}^{(r)} = \mathbf{B}$.

Proof. Remark that $\mathbf{R}^r = \mathbf{B} - \sum_{j=0}^{r-1} \mathbf{u}_j \mathbf{v}_j^T$, we will prove that $\mathbf{R}^{r'}$ is of rank $r - r'$ for $0 \leq r' \leq r$ by induction on r'.

For $r' = 0$ the relation is trivial. Suppose now that the relation is true for $0 \leq r' < r$, i.e. $\mathbf{R}^{r'}$ is of rank $r - r'$. Let V denote the image of $\mathbf{R}^{r'}$ of dimension $r - r'$ by induction hypothesis, we define W as the orthogonal complement of V in $\mathbb{R}^N$ so that its dimension is equal to $N - r + r'$. Similarly, we define V' and W' the corresponding spaces for $\mathbf{R}^{r'+1} = \mathbf{R}^{r'} - \mathbf{u}_{r'} \mathbf{v}_{r'}^T$. By definition of Algorithm 1, $\mathbf{u}_{r'}$ is the $k_*^{r'}$th column of $\mathbf{R}^{r'}$ up to a multiplicative constant so that it belongs to V. Therefore $V' \subset V$ and $W \subset W'$.

Let $j_* := j_*^{r'}$, we have

$$\mathbf{e}_{j_*}^T \mathbf{R}^{r'} = \mathbf{v}_{r'} \neq \mathbf{0},$$

because $w_*^r \neq 0$ and

$$\mathbf{e}_{j_*}^T (\mathbf{R}^{r'} - \mathbf{u}_{r'} \mathbf{v}_{r'}) = \mathbf{v}_{r'} - \underbrace{\mathbf{u}_{r'}(j_*)}_{1} \mathbf{v}_{r'} = \mathbf{0},$$

[6] Because the reference may not be easy to find and the proofs can be unclear, we also provide the proofs

using $\mathbf{u}_{r'}(j_*) = (w_*^{r'})^{-1}\mathbf{R}^r(j_*^{r'}, k_*^{r'}) = 1$. We deduce that $\mathbf{e}_{j_*}$ is in the left null space of $\mathbf{R}^{r'+1}$, but not in the one of $\mathbf{R}^{r'}$. In other words, $\mathbf{e}_{j_*} \in W' \setminus W$, so that $\dim(V') < \dim(V)$, or $\dim(V') \leq k - k' - 1$.

We also have $\operatorname{rank}(\mathbf{R}^{r'}) = \operatorname{rank}(\mathbf{R}^{r'+1} + \mathbf{u}_{r'}\mathbf{v}_{r'}^T) \leq \operatorname{rank}(\mathbf{R}^{r'+1}) + 1$, so that $\dim(V') = \operatorname{rank}(\mathbf{R}^{r'+1}) \geq r - r' + 1$

To conclude, we have $\dim(V') = r - r' - 1$, and so $\mathbf{R}^{r'+1}$ is of rank $r - r' - 1$, which proves our initial statement. In particular for $r' = r$, we have $\widehat{\mathbf{B}}^{(r)} = \mathbf{B}$. □

Lemma 6.6 (Interpolation property, [19, Lemma 4.5])**.** *Let $\mathbf{B} \in \mathbb{C}^{M\times N}$ be a matrix of rank at least $r \geq 1$ and $\widehat{\mathbf{B}}^{(r)}$ the cross approximation from Algorithm 1. For any pivot $(j_*^{r'}, k_*^{r'})$ with $0 \leq r' \leq r - 1$, there holds*

$$\widehat{\mathbf{B}}^{(r)}(:, k_*^{r'}) = \mathbf{B}(:, k_*^{r'}) \quad \text{and} \quad \widehat{\mathbf{B}}^{(r)}(j_*^{r'}, :) = \mathbf{B}(j_*^{r'}, :),$$

i.e. $\widehat{\mathbf{B}}^{(r)}$ reproduces the columns and rows associated with the pivots $(j_*^{r'}, k_*^{r'})$.

Proof. We will prove the result by induction on r for $1 \leq r$ and r lower than the exact rank of $\mathbf{B}$. In the case, $\mathbf{B}$ is of rank exactly r, we have actually that $\mathbf{B}$ is equal to $\widehat{\mathbf{B}}^{(r)}$ according to Lemma 6.5.

For $r = 1$, the statement follows from the definition of Algorithm 1, we have

$$\widehat{\mathbf{B}}^{(1)}(:, k_*^1) = \mathbf{B}(:, k_*^1) \quad \text{and} \quad \widehat{\mathbf{B}}^{(1)}(j_*^1, :) = \mathbf{B}(j_*^1, :).$$

Suppose now that the relation is true for $1 \leq r$ and strictly lower that the rank of $\mathbf{B}$, then we have for $0 \leq r' \leq r - 1$

$$\widehat{\mathbf{B}}^{(r)}(:, k_*^{r'}) = \mathbf{B}(:, k_*^{r'}) \quad \text{and} \quad \widehat{\mathbf{B}}^{(r)}(j_*^{r'}, :) = \mathbf{B}(j_*^{r'}, :),$$

or equivalently, $\mathbf{R}^r(:, k_*^{r'}) = \mathbf{0}$ and $\mathbf{R}^r(j_*^{r'}, :) = \mathbf{0}$. We have

$$\mathbf{R}^{r+1} = \mathbf{R}^r - \mathbf{u}_r\mathbf{v}_r^T = \mathbf{R}^r - (w_*^r)^{-1}\mathbf{R}^r(:, k_*^r)\mathbf{R}^r(j_*^r, :)^T$$

so that for $0 \leq r' \leq r - 1$, we have

$$\mathbf{R}^{r+1}(:, k_*^{r'}) = \mathbf{R}^{r+1}\mathbf{e}_{k_*^{r'}} = \mathbf{R}^r(:, k_*^{r'}) - (w_*^r)^{-1}\mathbf{R}^r(:, k_*^r)\mathbf{R}^r(j_*^r, :)^T\mathbf{e}_{k_*^{r'}}.$$

The first term in the right-hand side is equal to zero by induction hypothesis, while the second term is equal to $(w_*^r)^{-1}\mathbf{R}^r(:, k_*^r)\mathbf{R}^r(j_*^r, k_*^{r'})^T$ where $\mathbf{R}^r(j_*^r, k_*^{r'})$ is also equal to zero. We deduce that $\mathbf{R}^{r+1}(:, k_*^{r'}) = 0$ for $0 \leq r' \leq r - 1$ and similarly, we can prove that $\mathbf{R}^{r+1}(j_*^{r'}, :) = 0$. It remains to prove this relation for $r' = r$,

$$\begin{aligned}\mathbf{R}^{r+1}(:, k_*^r) &= \mathbf{R}^{r+1}\mathbf{e}_{k_*^r} = \mathbf{R}^r(:, k_*^r) - (w_*^r)^{-1}\mathbf{R}^r(:, k_*^r)\mathbf{R}^r(j_*^r, :)^T\mathbf{e}_{k_*^r} \\ &= \mathbf{R}^r(:, k_*^r) - (w_*^r)^{-1}\mathbf{R}^r(:, k_*^r)\underbrace{\mathbf{R}^r(j_*^r, k_*^r)^T}_{w_*^r} = 0.\end{aligned}$$

Similarly, we can prove that $\mathbf{R}^{r+1}(j_*^r, :)$, which finishes the proof. □

This last lemma shows that the low-rank approximation is built upon the rows and columns associated with the pivots. Remark that this is not the case a priori for the optimal low-rank approximation $\mathbf{B}^{(r)}$. It can also be proved that ACA can actually be seen as a pseudo-skeleton approximation according to [12, Lemma 3.32] or [19, Lemma 4.6].

Let us denote $\widehat{r}_{\text{final}}$, the final value of r obtained following Algorithm 1. Without taking into account the stopping criterion, the total number of operations of Algorithm 1 is $O(\widehat{r}_{\text{final}}MN)$ because of the search of the argmax of $\mathbf{R}^r$ and the update of the matrix $\mathbf{R}^{r+1}$, respectively in lines 5 and 9 of Algorithm 1. We also need to compute the whole matrix $\mathbf{B}$ to compute the argmax of $\mathbf{R}^r$.

6.1.3.3 Partially pivoted Adaptive Cross Approximation

As we have seen with Lemma 6.6, the resulting low-rank approximation only uses the columns and the rows associated with the pivots, so that if we knew in advance the pivots, we would not need to compute the whole matrix $\mathbf{B}$ and we would not need to update the whole residual $\mathbf{R}^{r+1} \in \mathbb{C}^{M\times N}$. Since we cannot know a priori the pivots because they are computed iteratively, a variant of Algorithm 1 is to restrict the search for the pivots to a part of the matrix.

This variant is given by Algorithm 2, where in each iteration r, we have a given row j_*^r and we look for the coefficient (j_*^r, k_*^r) of maximal absolute value in the residual $\mathbf{R}^r$, but only in the j_*^rth row. This means that we do not need to compute the whole matrix and residual, but only their j_*^rth row. Then, similarly, we look for the coefficient (j_*^{r+1}, k_*^r) of maximal absolute value in the k_*^rth column of the residual $\mathbf{R}^r$, which is never completely computed (this is why it does not appear explicitly in Algorithm 2). As in Algorithm 1, the successive rows and columns of residual associated with the pivots are used to build the low-rank approximation that we denote $\widetilde{\mathbf{B}}^{(r)}$.

Algorithm 2 Partially Pivoted ACA

1: **function** PARTIAL ACA($(j,k) \mapsto \mathbf{B}_{j,k}$) ▷ We do not need the whole matrix
2: **initialize** j_*^0 ▷ see Remark 6.7
3: $r = 0$
4: **while** (stopping criterion not satisfied) **do** ▷ see Section 6.1.3.4
5: $\mathbf{v}_r^T = \mathbf{B}(j_*^r, :) - \sum_{k=0}^{r-1} \mathbf{u}_k(j_*^r)\mathbf{v}_k^T$
6: $k_*^r = \text{argmax}_{k=1\dots N} |\mathbf{v}_r(k)|$
7: $w_*^r = \mathbf{v}_r(k_*^r)$
8: **if** $(w_*^r \neq 0)$ **then**
9: $\mathbf{u}_r = w_*^{-1}(\mathbf{B}(:, k_*^r) - \sum_{k=0}^{r-1} \mathbf{u}_k\mathbf{v}_k^T(k_*^r))$
10: $r = r + 1$
11: $j_*^r = \text{argmax}_{j=1\dots M} |\mathbf{u}_r(j)|$
12: **else**
13: pick any another j_*^r different from the previous pivots
14: **return** $(\mathbf{u}_j)_{j=1}^r$ and $(\mathbf{v}_j)_{j=1}^r$

Let us denote $\widetilde{r}_{\text{final}}$ the final value of r following Algorithm 2. At each iteration r, the number of operations to look for j_*^{r+1} and k_*^r is $O(M+N)$, but the number of operations to compute $\mathbf{v}_r$ and $\mathbf{u}_r$ is $O(r(M+N))$ because we cannot compute the residual iteratively as in Algorithm 1. Without taking into account the initialization step and the stopping criterion, we deduce that the complexity of Algorithm 2 is $O(\widetilde{r}_{\text{final}}^2(M+N))$.

Remark 6.7 (Initialization)**.** *The initial row is sometime presented as picked arbitrarily in the literature, but in [12, Section 3.4.3] a procedure is presented. It uses the geometry of the problem. Usually one can associate a geometric point with each degree of freedom, let us denote Cl_t the cluster of points associated with the rows, usually called targets, Cl_s the cluster of points*

associated with the columns usually called sources. Then, j_*^0 is chosen to be the closest target point to the barycenter of the target points.

6.1.3.4 Stopping criteria

The stopping criterion can be a given rank, for example $r < r_{\text{required}}$, or a given error of the approximation ε. In the latter case, we want to stop when

$$\|\mathbf{B} - \widehat{\mathbf{B}}^{(r)}\|_{\mathrm{F}} \leq \varepsilon \|\mathbf{B}\|_{\mathrm{F}}. \tag{6.3}$$

Since this needs the whole matrix $\mathbf{B}$ and the number of operations is $O(MN)$ at each iteration, we will use it with the fully pivoted ACA algorithm so that the complexity and the cost of storage is not changed. In [115, Section 3.2.2], it is stated that the number of operations is actually $O(r^2MN)$ because it said that we need to compute the vector products in $\widehat{\mathbf{B}}^{(r)} = \sum_{j=0}^{r-1} \mathbf{u}_j \mathbf{v}_j^T$ at each iteration to then compute the error. But remark that in line 8 of Algorithm 1, $\mathbf{R}^{r+1} = \mathbf{B} - \widehat{\mathbf{B}}^{(r)}$ is iteratively computed in $O(MN)$ operations so that we do not need to recompute the error doing all the vector products in $\widehat{\mathbf{B}}^{(r)} = \sum_{j=0}^{r-1} \mathbf{u}_j \mathbf{v}_j^T$.

But the complexity is still quadratic so that in practice, the stopping criterion given in Equation (6.3) is usually estimated by

$$\|\widetilde{\mathbf{B}}^{(r+1)} - \widetilde{\mathbf{B}}^{(r)}\|_{\mathrm{F}} \leq \varepsilon \|\widetilde{\mathbf{B}}^{(r+1)}\|_{\mathrm{F}},$$

which can be rewritten

$$\|\mathbf{u}_r \mathbf{v}_r^T\|_{\mathrm{F}} \leq \varepsilon \|\widetilde{\mathbf{B}}^{(r+1)}\|_{\mathrm{F}}, \tag{6.4}$$

First, notice that

$$\|\widetilde{\mathbf{B}}^{(r+1)}\|_{\mathrm{F}}^2 = \|\widetilde{\mathbf{B}}^{(r)}\|_{\mathrm{F}} + 2\langle \widetilde{\mathbf{B}}^{(r)}, \mathbf{u}_r \mathbf{v}_r^T \rangle_{\mathrm{F}} + \|\mathbf{u}_r \mathbf{v}_r^T\|_{\mathrm{F}}^2, \tag{6.5}$$

where $\langle \cdot, \cdot \rangle_{\mathrm{F}}$ stands for the Frobenius inner product. Let us precise how to compute all these terms

$$\|\mathbf{u}_r \mathbf{v}_r^T\|_{\mathrm{F}}^2 = \mathrm{Tr}(\overline{\mathbf{v}_r \mathbf{u}_r^T} \mathbf{u}_r \mathbf{v}_r^T) = \mathrm{Tr}(\overline{\mathbf{v}_r} \overline{\mathbf{u}_r^T} \mathbf{u}_r \mathbf{v}_r^T) = \|\mathbf{u}_r\|_2^2 \underbrace{\mathrm{Tr}(\overline{\mathbf{v}_r} \mathbf{v}_r^T)}_{\|\mathbf{v}_r\|_2^2}$$

$$\langle \widetilde{\mathbf{B}}^{(r)}, \mathbf{u}_r \mathbf{v}_r^T \rangle_{\mathrm{F}} = \sum_{j=0}^{r-1} \langle \mathbf{u}_j \mathbf{v}_j^T, \mathbf{u}_r \mathbf{v}_r^T \rangle_{\mathrm{F}} = \sum_{j=0}^{r-1} \mathrm{Tr}(\overline{\mathbf{v}_j \mathbf{u}_j^T} \mathbf{u}_r \mathbf{v}_r^T) = \sum_{j=0}^{r-1} \langle \mathbf{u}_j, \mathbf{u}_r \rangle_2 \underbrace{\mathrm{Tr}(\overline{\mathbf{v}_j} \mathbf{v}_r^T)}_{\langle \mathbf{v}_j, \mathbf{v}_r \rangle_2},$$

where $\langle \cdot, \cdot \rangle_2$ stands for the inner product for vectors. The number of operations are respectively $O(M + N)$ and $O(r(M + N))$ at each iteration r. According to Equation (6.5), we see that $\|\widetilde{\mathbf{B}}^{(r+1)}\|_{\mathrm{F}}$ from the criterion given in Equation (6.4) can be computed iteratively using $O(r(M + N))$ operations at each iteration, while computing $\|\mathbf{u}_r \mathbf{v}_r^T\|_{\mathrm{F}}$ requires $O(M + N)$. We deduce that the number of operations for this stopping criterion is $O(r^2(M + N))$, which is linear in the size of $\mathbf{B}$. With this choice, the Algorithm 2 has still a complexity of $O(r^2(M + N))$.

Using the stopping criteria given in Equation (6.3) or Equation (6.4), the rank of the approximation is chosen adaptively to ε.

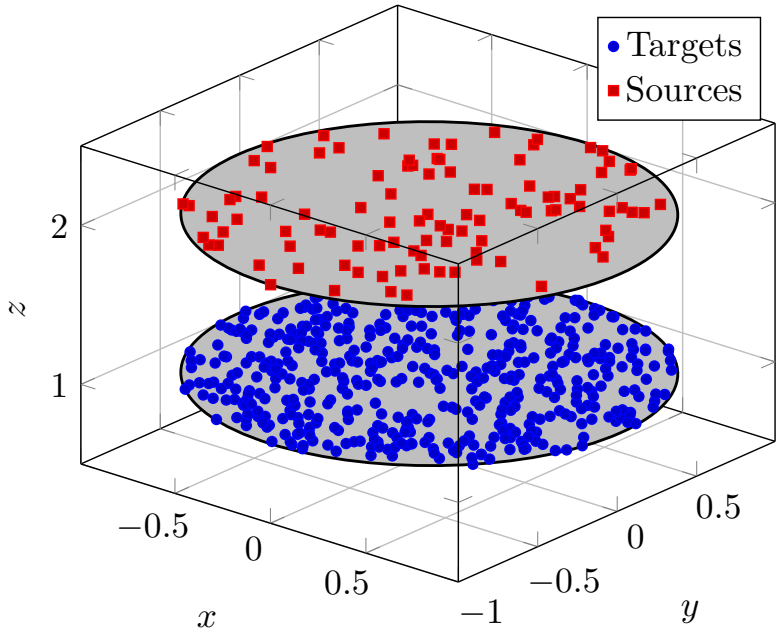


Figure 6.1 – Geometry used to illustrate low-rank approximation with $z_1 - z_2 = 1$.

6.1.4 Numerical illustrations

We now illustrate the approximation properties of the three low-rank approximations introduced in the previous sections: SVD, full ACA and partial ACA, respectively $\mathbf{B}^{(k)}$, $\widehat{\mathbf{B}}^{(k)}$ and $\widetilde{\mathbf{B}}^{(k)}$. Let us define the set of target points $(\mathbf{x}_j)_{j=1}^M$ and source points $(\mathbf{y}_k)_{k=1}^N$. We generate them randomly using a uniform law, such that they are both in a circle of radius 1 included in a plane normal to the z-axis, see Figure 6.1. We denote their respective z-coordinate z_1 and z_2.

We compute the low-rank approximations of the matrix $\mathbf{B} \in \mathbb{R}^{M\times N}$ defined as $\mathbf{B}_{j,k} = 1/|\mathbf{x}_j - \mathbf{y}_k|$ with $M = 500$ and $N = 100$. The errors of approximation in Frobenius norm as a function of the rank are given in Figure 6.2 for $z_1 - z_2 = 1$ and $z_1 - z_2 = 2$. In both cases, we observe an exponential decrease of the error of approximation. The SVD approximation gives always the best approximation, which is consistent with Theorem 6.3. Then, the full ACA approximation is better than the partial ACA approximation, which is expected since there is more freedom in the choice of the pivots in Algorithm 1 than in Algorithm 2. Remark also that when the distance between the two circles increases so that the interaction in farther away from the singularity of $1/|\mathbf{x} - \mathbf{y}|$, the approximation for a given rank is improved.

6.2 $\mathcal{H}$-matrices

In Section 6.1, we presented techniques to compute the low-rank approximations of a matrix $\mathbf{B} \in \mathbb{C}^{M\times N}$. These techniques are especially efficient when $\mathbf{B}$ has very fast decreasing singular values, which is not the case for a general BEM matrix. It is actually not the case because of the singularity of Green functions.

To illustrate this behavior, we consider the $\mathbb{P}_1$-Lagrange discretization of the weakly singular operator and the hypersingular operator on a circle associated with the equation $-\Delta u + u = 0$

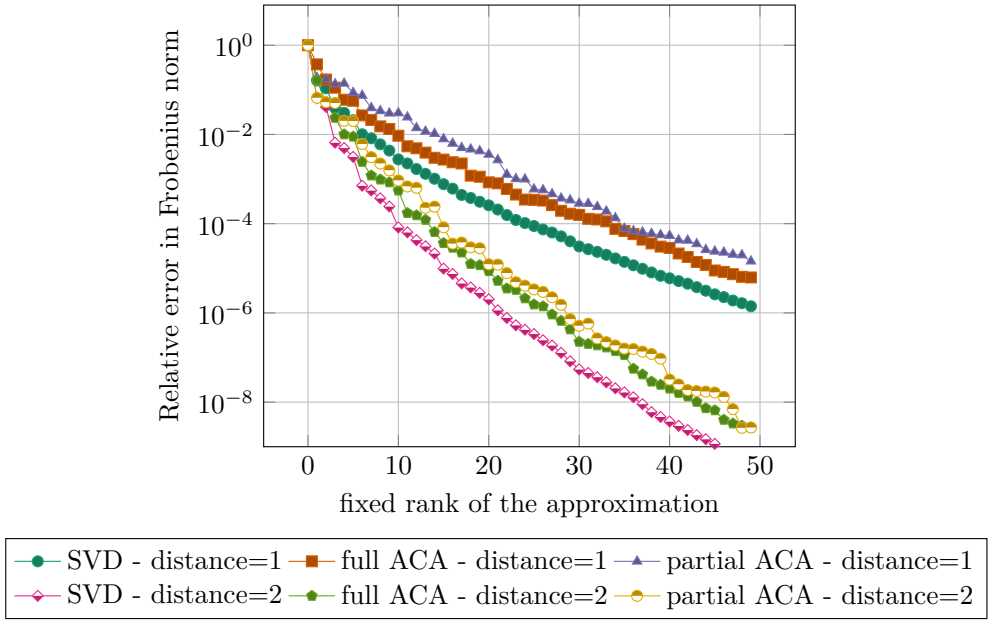


Figure 6.2 – Approximation of the kernel $1/|\mathbf{x}-\mathbf{y}|$ by low-rank matrices for the geometry given in Figure 6.1.

for $d = 2$ and $M = N = 63$. In Figure 6.3, we show the BEM matrices associated with these two operators normalized by their maximal values, and their corresponding singular values. We observe that great variations in the value of the coefficients occur around the diagonal (and also on the top right and bottom left corners by periodicity). This is because the diagonal corresponds to close interactions, and the Green functions are singular in this case. According to Theorem 6.3, we see that even if we took the SVD approximation of these matrices with a rank equal to 62, we could not obtain a better error than 10^{-2} in Frobenius norm.

But we see that apart from the diagonal, the values of the coefficients show smaller variations compared to the diagonal. In this case, the coefficients correspond to far interactions so that the Green function is very regularizing. The idea of hierarchical matrices is based on a hierarchical geometric partition of the domain, called *cluster tree*, that induces a hierarchical representation of the blocks of the matrix, called *block tree*. This allows us to identify the blocks that correspond to far interactions and that can be compressed using low-rank approximation. We develop this idea in the following.

6.2.1 Cluster tree

Let us consider a generic BEM matrix $\mathbf{A} \in \mathbb{C}^{N\times N}$ stemming from the Galerkin discretization of a BIO as presented in Section 2.4.5. Each coefficient corresponds to the interaction of two degrees of freedom φ_j, φ_k so that $\mathbf{A}_{j,k} = a(\varphi_j, \varphi_k)$ where a is the bilinear form of the problem and $1 \leq j, k \leq N$. We denote $\mathcal{I} = (1, \ldots, N)$ the global array of indices, it is completely arbitrary and it is usually defined by the numbering of the degrees of freedom. We can usually associate a degree of freedom φ_j with a geometric point $\mathbf{x}_j$ (center of cell for $\mathbb{P}_0$-Lagrange, node for $\mathbb{P}_1$-Lagrange for example) and some "weight" g_j (size of the cell for $\mathbb{P}_0$-Lagrange and size of the

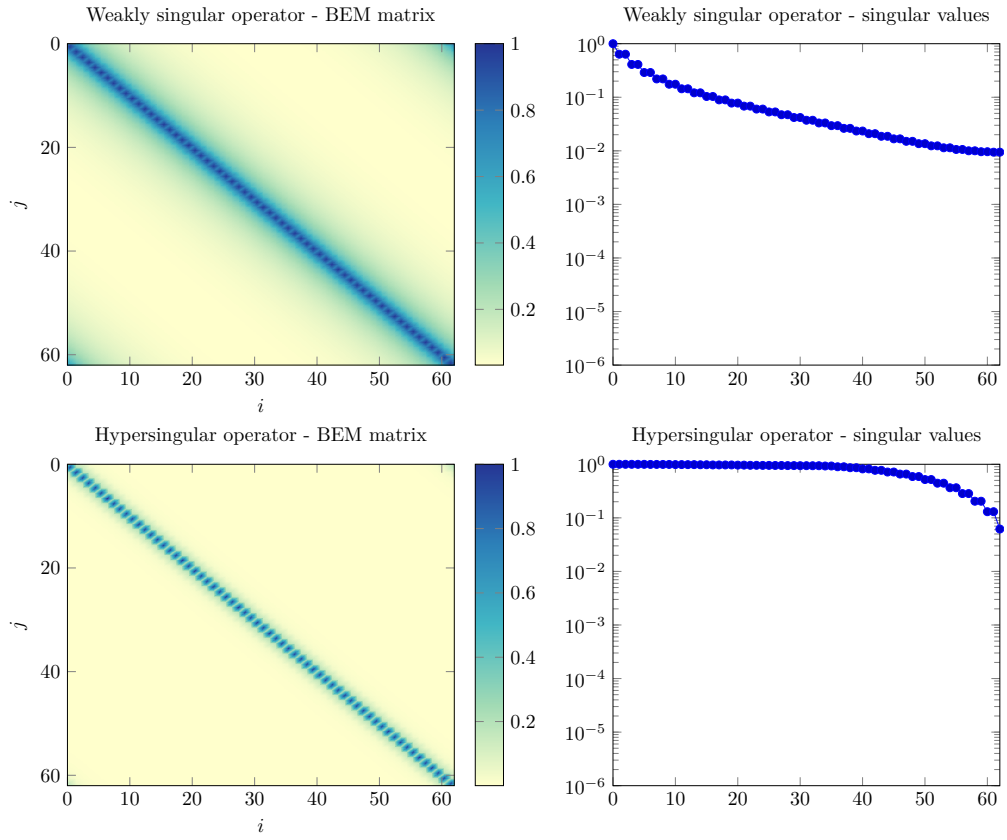


Figure 6.3 – Discretization of the weakly singular operator and the hypersingular operator on a circle associated with the equation $-\Delta u + u = 0$ for $d = 2$ and $M = N = 63$. Normalized values of the coefficients and singular values.

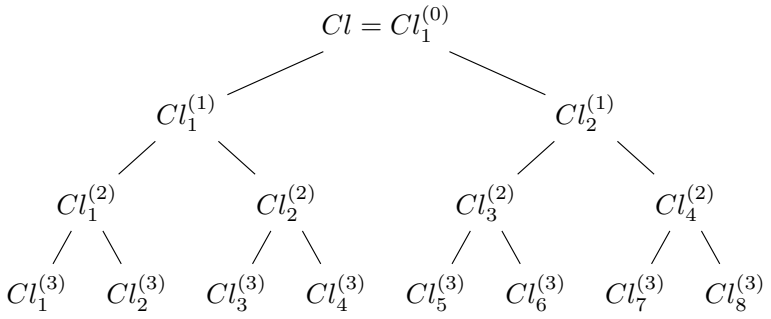


Figure 6.4 – Example of cluster tree for $p = 3$.

support of the shape functions for $\mathbb{P}_1$-Lagrange for example). Thus, we can associate an array of indices with a cluster/set of points, for example

$$Cl = \{(\mathbf{x}_j, g_j) \,|\, j \in \mathcal{I}\}\,.$$

Denoting $Cl_1^{(0)} = Cl$, the goal of the hierarchical geometric partition is to define hierarchically subclusters of $Cl_1^{(0)}$ that group closer points. To do so, we will separate each cluster $Cl_l^{(p)}$ in two sons recursively $Cl_{2(l-1)+1}^{(p+1)}$ and $Cl_{2(l-1)+2}^{(p+1)}$, so that we obtain a binary tree whose nodes correspond to clusters, as shown in Figure 6.4 for example. Then, each subcluster $Cl_l^{(p)}$ can be associated with a subinterval $\mathcal{I}_l^{(p)}$ i.e., a set of integers corresponding to the numbering of the geometric points in $Cl_l^{(p)}$, see Figure 6.5. We define the isomorphism $f(Cl_l^{(p)}) = \mathcal{I}_l^{(p)}$ that gives the associated numbering of a cluster.

To actually separate a cluster in two subclusters, we use the algorithm defined in [132, p. 3.1.2] and given in Algorithm 3. Algorithm 3 is similar to a *principal component analysis*, it computes the covariance matrix associated with the distance of each point to the barycenter. The first eigenvector of this covariance matrix, the first principal component, corresponds to the direction of longest expanse of the cluster. Then, we separate the cluster along a plane perpendicular to this direction and containing $\mathbf{x}_c$. This approach and the fact that we weight the points should provide a balanced partition of the clusters in most cases. We should emphasize that since the covariance matrix is symmetric and in $\mathbb{R}^{3\times 3}$, the associated eigenproblem can be solved explicitly [141].

This cluster tree actually defines a permutation of the global numbering such that the degrees of freedom in a cluster are numbered continuously. We denote p_{final} the depth of the cluster tree, then the permuted numbering $\mathcal{J}_l^{(p)}$ is defined as the concatenation of $(\mathcal{I}_m^{(p_{\text{final}})})_m$ for $0 \leq p \leq p_{\text{final}}$, $1 \leq l \leq 2^p$ and $2^{p_{\text{final}}-p}(l-1)+1 \leq m \leq l2^{p_{\text{final}}-p}$.

In particular, we have $\mathcal{J} = (\mathcal{I}_1^{p_{final}}, \ldots, \mathcal{I}_{2^{p_{final}}}^{p_{final}})$ and we give an example in Figure 6.5, where $\mathcal{J} = (4, 0, 6, 3, 8, 5, 2, 7, 9, 1)$. The array $\mathcal{J}$ allows us to go from the permuted numbering to the original numbering. Notice that a cluster is characterized by the position of its first element in $\mathcal{J}$, also called offset, and its size. In our example, $\mathcal{J}_2^{(2)} = (3, 6, 8)$ can also be defined as the subsequence in $\mathcal{J}$ whose offset is 3 and its size is 3.

Remark 6.8 (Other clustering approaches)**.** *There exist other approaches to define a cluster tree. For example, in [19, Section 2.1], variants are presented where bounding boxes are used with respectively a quadtree in 2D and an octree 3D.*

Algorithm 3 Cluster tree

function CLUSTER TREE(Cl) ▷ Cl can be any cluster

Mass of cluster:

$$G = \sum_{j \in f(Cl)}^{N} g_j,$$

Center of the cluster:

$$\mathbf{x}_c = \frac{1}{G} \sum_{j \in f(Cl)} \mathbf{x}_j,$$

Covariance matrix of cluster:

$$\mathbf{C} = \sum_{j \in f(Cl)} g_j (\mathbf{x}_j - \mathbf{x}_c)(\mathbf{x}_j - \mathbf{x}_c)^T \in \mathbb{R}^{3\times 3},$$

Eigenvalues and eigenvectors:

$$\mathbf{C}\mathbf{v}_l = \lambda_l \mathbf{v}_l, \quad l \in \{1,2,3\}, \quad \lambda_1 \geq \lambda_2 \geq \lambda_3,$$

Separation: Let $Cl_1 = \emptyset$ and $Cl_2 = \emptyset$,
for $j \in f(Cl)$ **do**
  if $(\mathbf{x}_j - \mathbf{x}_c)^T \mathbf{v}_1 \geq 0$ **then**
    $Cl_1 = Cl_1 \cup (\mathbf{x}_j, g_j)$
  else
    $Cl_2 = Cl_2 \cup (\mathbf{x}_j, g_j)$
return Cl_1 and Cl_2

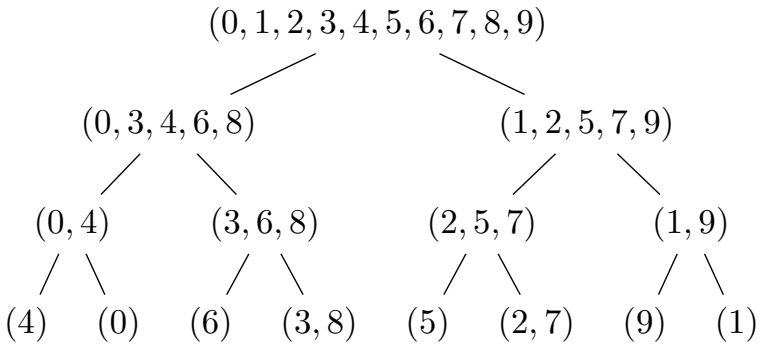


Figure 6.5 – Example of numbering induced by a cluster tree.

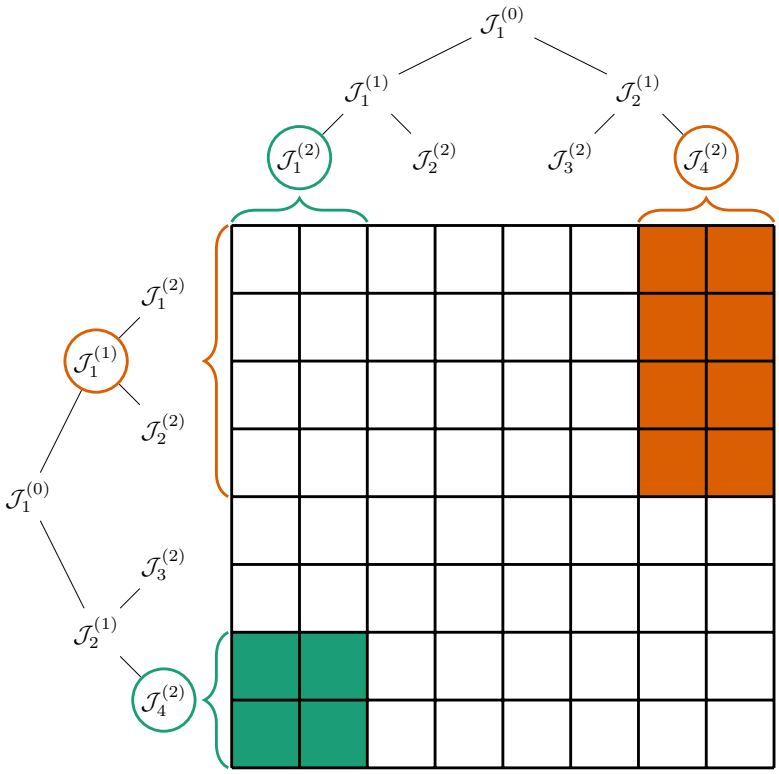


Figure 6.6 – Relation between blocks and cluster trees.

6.2.2 Block tree

Let us denote $\mathcal{I}_l^{(p)}$, $\mathcal{I}_{l'}^{(p')}$ two subintervals defined by the cluster tree. Remark that $(\mathcal{I}_l^{(p)}, \mathcal{I}_{l'}^{(p')})$ actually corresponds to a subblock of $\mathbf{A}$, which may not be contiguous. Similarly, $(\mathcal{J}_l^{(p)}, \mathcal{J}_{l'}^{(p')})$ corresponds to a subblock of $\mathbf{A}$ after permutation so that the block is contiguous. An example is given in Figure 6.6 where the block associated with $(\mathcal{J}_1^{(1)}, \mathcal{J}_4^{(2)})$ and $(\mathcal{J}_4^{(2)}, \mathcal{J}_1^{(2)})$ are shown. The fact that the blocks are contiguous has several advantages in practice, so we will always work on the permuted matrix in the following.

The block tree is then a quadtree where the nodes are pairs $(\mathcal{J}_l^{(p)}, \mathcal{J}_{l'}^{(p')})$ and their sons are defined using the sons of $\mathcal{J}_l^{(p)}$ and $\mathcal{J}_{l'}^{(p')}$. It is actually an alternative representation of the permuted matrix. Starting from the root $(\mathcal{J}, \mathcal{J})$, this hierarchical structure allows going from the coarsest blocks to the smaller blocks while descending in the tree.

To know which block can be compressed, we use an *admissibility condition* based on the geometry of clusters. It is a geometric condition which states that two clusters are sufficiently far apart so that the associated block corresponds to a far interaction. As we have seen at the beginning of the section, far interaction are regular because of the Green kernel so that they should contain little information, or more precisely they should have fast decreasing singular values. In this case, low-rank approximations introduced in Section 6.1 should be relevant.

Current literature provides various admissibility criteria. It should be considered as problem dependent. In our case, we chose the following admissibility criterion (see [132, eq. 3.15]) for

two clusters Cl_t and Cl_s

$$\min\big(\operatorname{diam}(Cl_t), \operatorname{diam}(Cl_s)\big) < \eta \ \operatorname{dist}(Cl_t, Cl_s),$$

where

$$\operatorname{diam}(Cl_s) = \max_{\mathbf{x}_{k_1}, \mathbf{x}_{k_2} \in Cl_s} |\mathbf{x}_{k_1} - \mathbf{x}_{k_2}|, \quad \operatorname{dist}(Cl_s, Cl_t) = \min_{\mathbf{x}_k \in Cl_s, \mathbf{x}_l \in Cl_t} |\mathbf{x}_k - \mathbf{x}_l|,$$

and $\eta > 0$ is a fitting parameter. As suggested in [132] and in order to avoid the quadratic cost of the computation of $\operatorname{diam}(Cl_s)$, the practical implementation makes use of the more restrictive but more easily computable admissibility condition:

$$2 \ \min\big(\max_{\mathbf{x}_k \in Cl_s} |\mathbf{X}_s - \mathbf{x}_k|, \max_{\mathbf{x}_l \in Cl_t} |\mathbf{X}_t - \mathbf{x}_l|\big) < \eta \, |\mathbf{X}_t - \mathbf{X}_s| - (\max_{\mathbf{x}_k \in s} |\mathbf{X}_s - \mathbf{x}_k| + \max_{\mathbf{x}_l \in t} |\mathbf{X}_t - \mathbf{x}_l|), \quad (6.6)$$

where $\mathbf{X}_s$ (resp. $\mathbf{X}_t$) is the center of Cl_s (resp. Cl_t). To summarize, the block tree is traversed recursively calling Algorithm 4 with $(Cl_1^{(0)}, Cl_1^{(0)})$.

Algorithm 4 Block tree

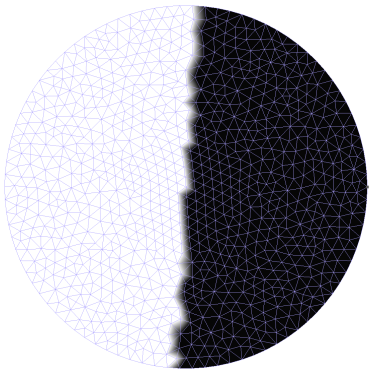
function BLOCK TREE($Cl_l^{(p)}$, $Cl_{l'}^{(p')}$)
  if $(Cl_l^{(p)}, Cl_{l'}^{(p')})$ is admissible **then** ▷ See condition given in Equation (6.6)
    Compression of $\mathbf{A}|_{\mathcal{J}_l^{(p)} \times \mathcal{J}_{l'}^{(p')}}$
  else if $Cl_l^{(p)}$ and $Cl_{l'}^{(p')}$ are both leafs **then**
    Assembling of $\mathbf{A}|_{\mathcal{J}_l^{(p)} \times \mathcal{J}_{l'}^{(p')}}$
  else if $Cl_l^{(p)}$ has more elements than $Cl_{l'}^{(p')}$ **then**
    BLOCK TREE ($Cl_{2(l-1)+1}^{(p)}$,$Cl_{l'}^{(p')}$)
    BLOCK TREE ($Cl_{2(l-1)+2}^{(p)}$,$Cl_{l'}^{(p')}$)
  else
    BLOCK TREE ($Cl_l^{(p)}$,$Cl_{2(l'-1)+1}^{(p')}$)
    BLOCK TREE ($Cl_l^{(p)}$,$Cl_{2(l'-1)+2}^{(p')}$)

6.2.3 Numerical illustrations

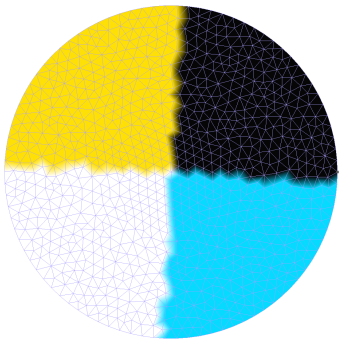
We give a small illustration of the cluster tree and block tree introduced in the previous section. We take as an example a disk with a mesh of $N = 924$ degrees of freedom and we want to assemble the approximation of the weakly singular operator $\mathbf{V}$ for the Laplace equation. We show the results for the cluster tree using GMSH [60].

In Figure 6.7, we show several levels of the cluster tree. Remark that the clusters are well-balanced, this is because we chose an algorithm that always looks for the direction of longest expense of the cluster and cut along the perpendicular plane containing the barycenter.

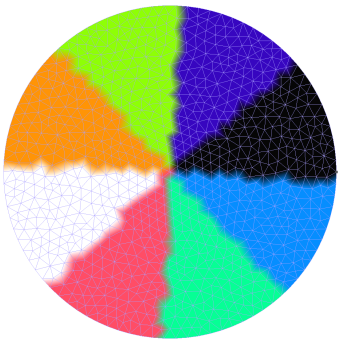
We show the resulting hierarchical matrices in Figure 6.8 for $\varepsilon = 0.01$ and $\eta = 100$ where the level of green describes the level of compression while the red blocks are dense blocks. We also write the rank of the low-rank approximations, when the block is not too small. We define the compression rate of the matrix as $\text{Comp} = 100(1 - N_g/MN)$ where N_g is the number of generated coefficients, so that when Comp is closer to 100, it is more compressed. We obtain the



(a) p=1



(b) p=2



(c) p=3

Figure 6.7 – Cluster tree for a disk with $N = 924$ at several depth p.

following compression rates

$$\text{Comp}_{\text{SVD}} = 74.98\% > \text{Comp}_{\text{fullACA}} = 68.82\% > \text{Comp}_{\text{partialACA}} = 65.93\%.$$

As expected, using SVD approximations gives the best compression rate, but remember that it is also the most costly. Similarly, we see that the full ACA approximation gives a better compression rate than the partial ACA approximation, but again, remember that it requires to compute the whole matrix.

6.3 Implementation

We present here the organization of Htool, a C++ header-only template library for hierarchical matrices parallelized with MPI and OpenMP. Our goal was to make it as modular as possible to allow easy additions (other compression methods for example), and it should be as easy as possible to use. For the latter, we aimed to use black-box methods to not require too many parametrizations from the end-user.

We often use a simple and efficient design called Abstract Base Class (ABC). It consists in a pure abstract class that defines the interface, i.e. all the functions used by the rest of the code and pure virtual functions that needs to be defined by derived classes. This abstract class cannot be instantiated directly. This approach allows for a great factorization of codes and a modular design that makes additions easy to implement.

6.3.1 Clustering

We defined ClusterTree as a class containing the root of the cluster tree and some auxiliary functions. The real cluster tree is defined as a linked list of Cluster objects that contain the radius, the center of the cluster and the pointers to its Cluster sons. It takes the geometry as an argument, in practice an array of 3D points, and optionally an array of weights.

When traversing the tree, we build the array $\mathcal{J}$ implementing the permutation that allows going from the numbering stemming from the clustering to the numbering of the given geometric points (see Figure 6.5 for an example).

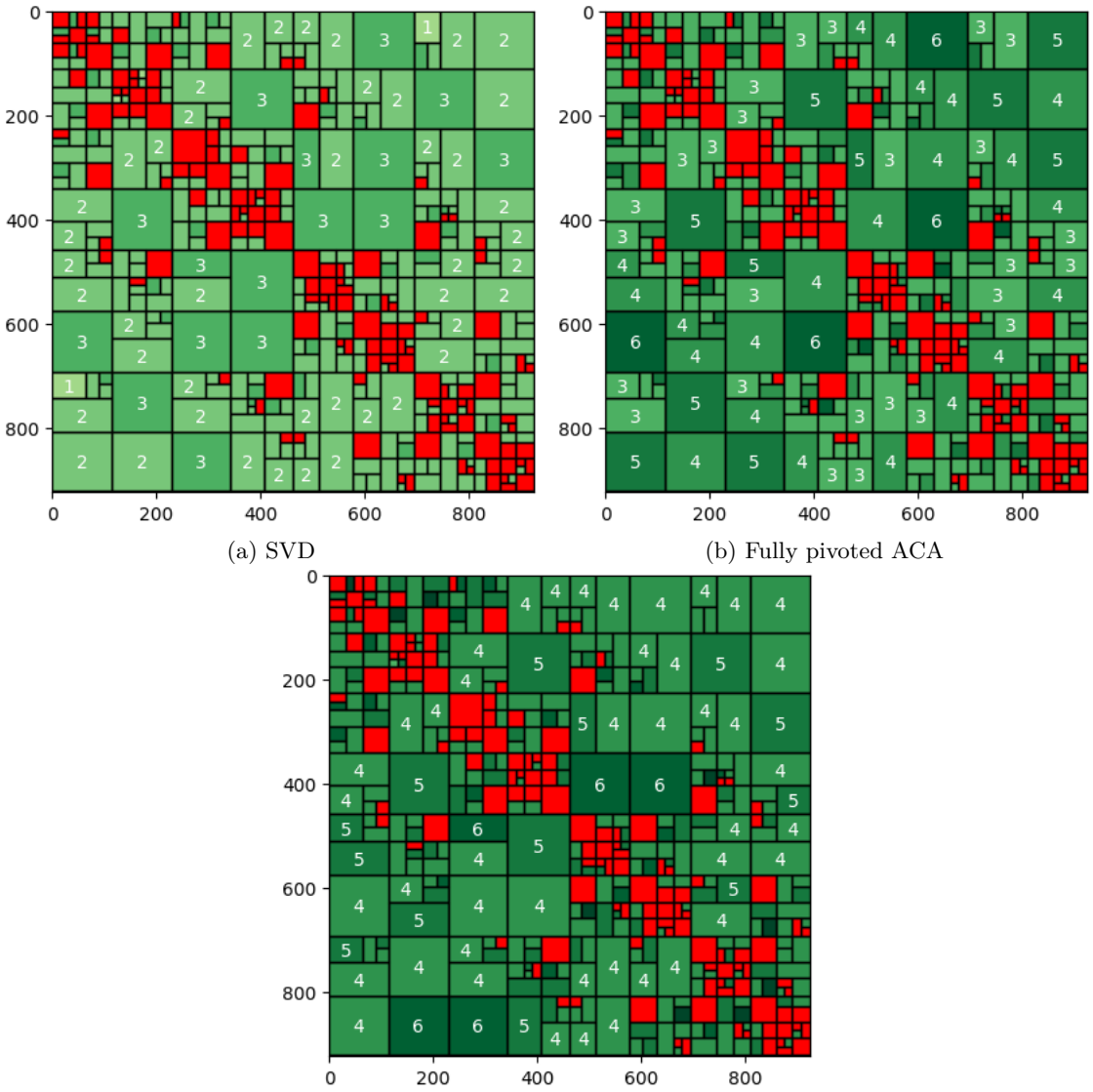
Then, the Cluster objects also contain their offset and the number of associated geometric points, which completely characterizes the cluster with $\mathcal{J}$.

6.3.2 Matrix

We use the ABC pattern to define matrix structures where IMatrix is the abstract class. It only defines the number of rows and columns and their associated accessors. The pure virtual function is get_coef that takes a pair (j, k) and returns the coefficient (j, k). The type of the coefficient is a template argument of the class.

As stated in [158, Section 1.3.1], this type of dynamic polymorphism using pure virtual function can imply performance penalties, but this overhead can be negligible if the actual code of the function is costly. Since, BEM quadratures are usually expensive because of the singularity of the Green kernel, we expect that the overhead is indeed negligible.

IMatrix defines the interface that the user can use to build hierarchical matrices with their own kernel. The user can define a class deriving from IMatrix and its function get_coef that implements how to compute a coefficient (j, k), and we will see that it will be a required argument to our class implementing hierarchical matrices.



(a) SVD

(b) Fully pivoted ACA

(c) Partially pivoted ACA

Figure 6.8 – Hierarchical matrices associated with the geometry from Figure 6.7, $\varepsilon = 0.1$ and $\eta = 10$.

We also define MATRIX as a derived class that inherits from IMATRIX. It implements dense matrices with BLAS operations. And we define SUBMATRIX as a derived class that inherits from MATRIX. It corresponds to a dense block of the global matrix, so it stores the offsets and sizes of the source and target clusters, corresponding respectively to the columns and rows of the global matrix.

Remark 6.9 (GET_SUBMATRIX). *Using Galerkin method, assembling a block looping over elements is usually more efficient, than looping over degrees of freedom. So that IMATRIX also provides a virtual function GET_SUBMATRIX that can be overloaded by the user. By default, it loops over the coefficients of given arrays of source points and target points using GET_COEF. It returns the corresponding block as a SUBMATRIX.*

6.3.3 Low-rank approximation

Following the ABC pattern, we define a class LOWRANKMATRIX to implement low-rank approximations of a dense matrix. It contains all data members and methods needed to use low-rank approximations. The pure virtual function is the BUILD function that needs to be implemented by each class deriving from LOWRANKMATRIX.

Then, we defined the compressors such as the partial ACA approximation as derived classes from LOWRANKMATRIX, defining the BUILD function that implements the particular algorithm to assemble the low-rank approximation. It takes as arguments a IMATRIX to generate the coefficients, a source and target cluster because it will correspond to a block of the global matrix. At the moment, the three algorithms from Section 6.1 are implemented.

The only optimization we made is that we represent the low-rank approximation by two matrices $\mathbf{U} \in \mathbb{C}^{M \times r}$ and $\mathbf{V} \in \mathbb{C}^{r \times N}$ instead of sum of vectors as in Equation (6.1), where r is the rank of the approximation. It allows for matrix-vector products that uses BLAS 2 routines, instead of looping over the vectors.

6.3.4 $\mathcal{H}$-matrices

The HMATRIX class implements hierarchical matrices, it takes two template arguments, the type of the coefficients and the class of the compressor which is expected to follow the interface defined in LOWRANKMATRIX. This design pattern relies on static polymorphism and it can be called policy-based design. Each compressor defines a "policy class" that the host, the HMATRIX class, can take. But policy classes have to follow a policy, an implicit interface. Using the ABC design for the compressors with LOWRANKMATRIX, we actually made this interface explicit, so that new additions just need to derive from this abstract class to follow the needed policy.

The constructor takes as arguments smart shared pointers to a source cluster tree and a target cluster tree. This allows sharing the cluster tree between several HMATRIX. It can be useful when discretizing the direct method in BEM where several operators on the same geometry are necessary, see Equation (2.14). It also requires a IMATRIX to generate the coefficients.

In practice, it goes through the block tree recursively as described in Algorithm 4. But we only define objects of type BLOCK that contain the associated target and source clusters. Then, each MPI process determines which block it needs to compute, following a band decomposition as described in Section 6.3.5. To compute the blocks, the IMATRIX is given to the constructor of the chosen compressor for admissible blocks and to the constructor of SUBMATRIX for dense blocks.

6.3.5 Parallelization

We chose to use both distributed and shared parallelism. Because we want to use DDM preconditioners, it is natural to use distributed parallelism, but then one has to choose how to distribute the data.

There exist several classical approaches to partition the data for a dense matrix. In our case, notice that the diagonal usually contains more dense blocks corresponding to close interactions (see Figure 6.8 for example). Then, we cannot just decompose the data in a regular grid because it would imply that some processes would only have compressed extra-diagonal blocks, which could lead to load-balancing issues.

Since we want to apply a DDM preconditioner, it is relevant to use a row-wise partition. After a matrix-vector product, the resulting vector is local and so are the DDM preconditioners. This means that if we define the subdomain partitioning in accordance with the cluster tree partitioning, we only need to exchange the values in the overlap between MPI processes to be able to apply the preconditioner. That is why we define the domain decomposition as a level of the cluster tree, and then we add overlap.

The issue with a row-wise decomposition is that when increasing the number of processes, we need to traverse the target cluster tree, which may break larger blocks that could be compressed. To be able to scale with more processes but without going too deep in the block tree, we use OpenMP threads to parallelize loops over blocks (when building them and for the matrix-vector product). But as we have said, each row has dense diagonal blocks, compressed extra-diagonal blocks, and their size are usually different. This means that the distribution of the tasks to each thread needs to be done dynamically, which is done by using guided scheduling. Guided scheduling is appropriate when the iterations in the loop are poorly balanced between each other, which is the case for hierarchical matrices stemming from BEM matrices, see Figure 6.8 for example.

The parallelization pattern is summarized in Figure 6.9. In Figure 6.10, we give the mean assembly time for the matrix associated with the test case introduced in Section 6.1.4 with 1 000 000 points in each disk and a distance equal to 0.05. More precisely, we do a strong scaling increasing the number of MPI processes for a fixed number of OpenMP threads. The total number of cores (MPI process and OpenMP threads) goes from 32 to 2048 for each curve. We take $\eta = 100$ and $\varepsilon = 10^{-5}$ so that we always obtain a compression of the order of 99%.

When increasing the number of MPI processes, we see in Figure 6.10 that it always scales well for a smaller number of MPI processes, but we need more OpenMP threads to scale longer, as expected. Adding more OpenMP threads allows not breaking larger compressed blocks and it balances the parallelization pattern.

6.3.6 DDM preconditioners

The iterative solvers and DDM preconditioners are provided through the class DDM that takes an IMatrix, a HMatrix and the information related to the domain decomposition.

Let us denote p the chosen level of parallelization. To build the one-level part of the preconditioner (second term in the right-hand side of Equation (5.5)), in the MPI process numbered m, we need to assemble the diagonal block corresponding to the interaction between $\mathcal{J}_m^{(p)}$ and itself. We densify the relevant compressed blocks and put all the blocks associated with this diagonal block in a dense matrix. Then, we need to generate the coefficients stemming from the degrees of freedom in the overlap using IMatrix. Once we have the diagonal block corresponding to the subdomain associated with the MPI process, we apply a LU factorization using Lapack [6] to the diagonal block for assembling the one-level preconditioner.

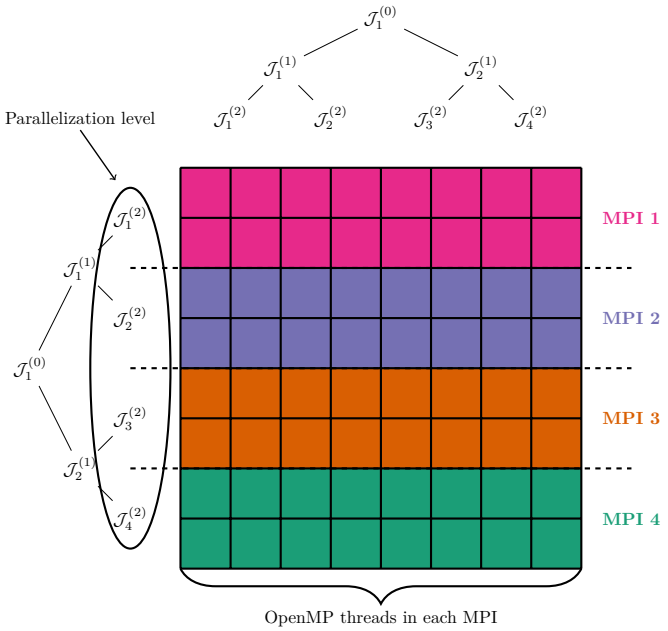


Figure 6.9 – Description of the parallelization pattern.

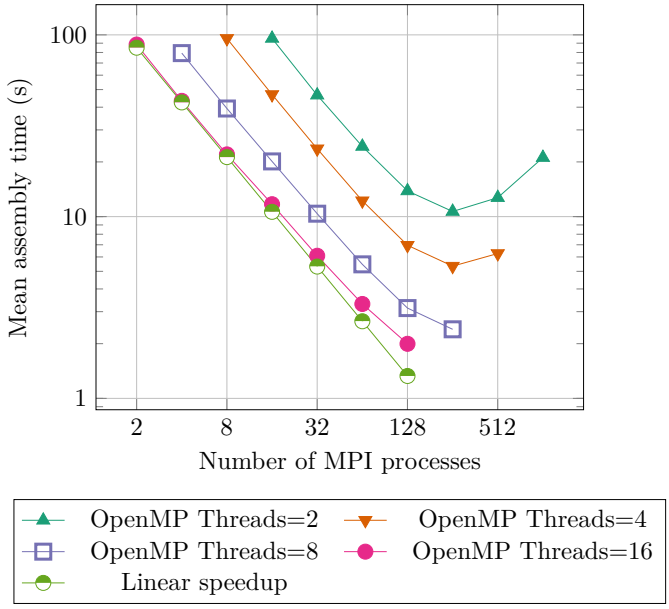


Figure 6.10 – Strong scaling for the assembly of the matrix associated with the kernel $1/|\mathbf{x}-\mathbf{y}|$, the geometry given in Figure 6.1 and 1 000 000 points.

Remark 6.10 (Generating diagonal blocks)**.** *Instead of densifying the diagonal blocks and regenerating the coefficients from the overlap, one could build a hierarchical matrix associated with the subdomain with overlap. Then, a $\mathcal{H}$-LU factorization could be used for the one-level preconditioner for example.*

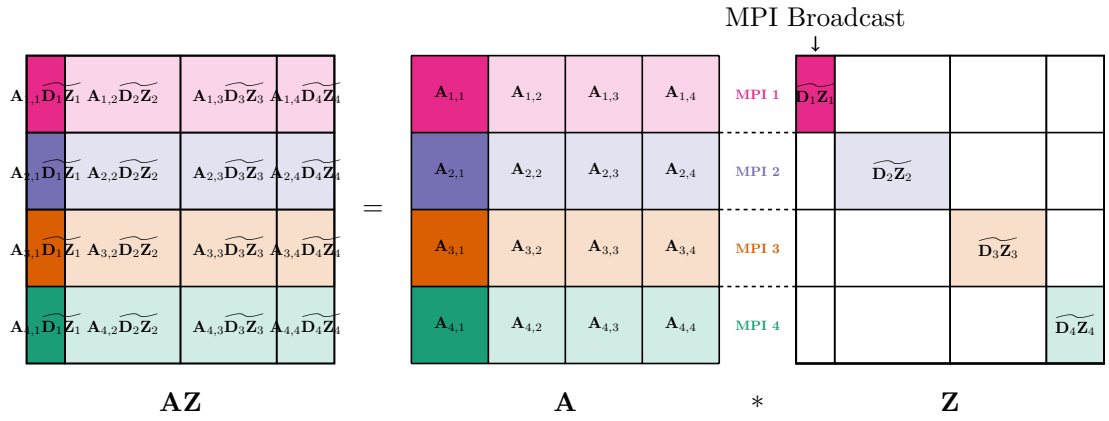
To assemble the coarse component (first term in the right-hand side of Equation (5.5)), DDM has a method called BUILD_COARSE_SPACE that takes the relevant arguments depending on the type of coarse space (see Section 5.3.4). For GenEO Stiffness and GenEO Slobodeckij for example, it just takes a matrix corresponding to $\mathbf{B}_m$ in Equation (5.8), so respectively the stiffness matrix $\mathbf{K}_m$ and the Sobolev-Slobodeckij matrix $\mathbf{H}_m$. For GenEO Single layer, this function is overloaded so that it can take a IMATRIX and the 3D points of the geometry to assemble a local hierarchical matrix corresponding to $\mathbf{V}_m$ that we densify like the diagonal block of $\mathbf{A}$. For this coarse space, the function also takes as input the local mass matrix whose inverse is needed in the definition of GenEO single layer Equation (5.17).

In every case, the function calls Lapack to solve the generalized eigenvalue problem given in Equation (5.8). The number of eigenvectors chosen for the coarse space depends on a user-defined number, or a user-defined threshold on the eigenvalues (we refer to Section 5.3.3 for a presentation of the GenEO coarse space). Each processor stores these eigenvectors as columns of a local dense matrix $\mathbf{Z}_m \in \mathbb{C}^{N_m \times N_{\text{loc},m}}$ where N_m is the number of degrees of freedom in the mth subdomain with overlap and $N_{\text{loc},m}$ is the local number of eigenvectors chosen to assemble the coarse space. The interpolation matrix from the finite element space to the coarse space $\mathbf{R}_0 = \mathbf{Z}^T$ is defined as the transpose of the concatenation of $(\mathbf{R}_m^T \mathbf{D}_m \mathbf{Z}_m)_{m=1}^n \in (\mathbb{C}^{N \times N_{\text{loc}}})_{m=1}^n$.

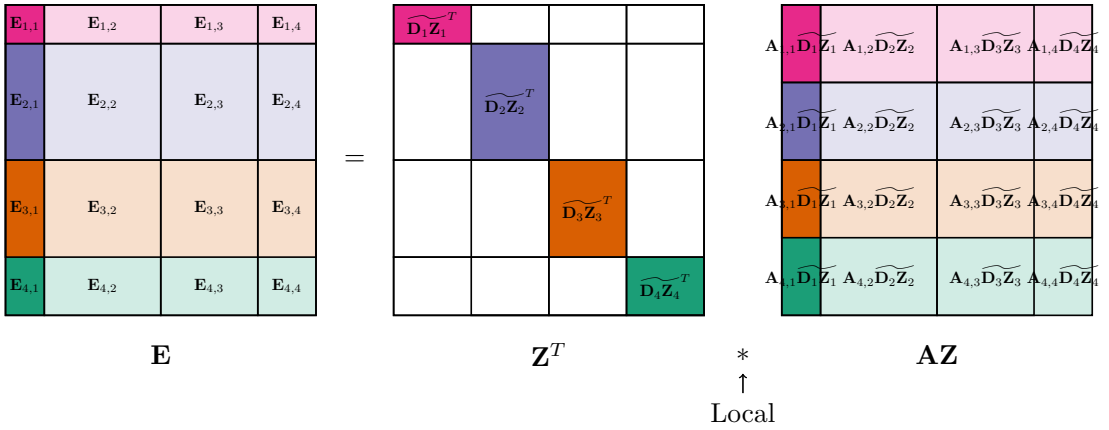
In practice, we define the partition of unity $(\mathbf{D}_m)_{m=1}^n$ such that $\mathbf{D}_m$ is a diagonal matrix whose diagonal coefficients are equal to 1 if they correspond to a degree of freedom in the subdomain without overlap, or 0 otherwise. In this case, we denote $\widetilde{\mathbf{D}_m \mathbf{Z}_m}$ the block $\mathbf{D}_m \mathbf{Z}_m$ stripped of its zero-blocks due to this particular definition of the partition of unity. Then, an example of matrix $\mathbf{Z}$ is given by the top right matrix in Figure 6.11. The assembling procedure to build the coarse component $\mathbf{E} = \mathbf{Z}^T \mathbf{A} \mathbf{Z}$ is described in Figures 6.11 and 6.12. The first figure shows the operations where the block in $\mathbf{Z}$ associated with the first subdomain is involved, while the second figure is similar but for the block in $\mathbf{Z}$ associated with the second subdomain. All the blocks involved in the computation are in brighter colors.

In practice, $\mathbf{E}$ is built by looping over the mth subdomains to execute the following operations. For subdomain m, process m broadcast $\widetilde{\mathbf{D}_m \mathbf{Z}_m}$ to every other MPI process so that they all can do the product $\mathbf{A}_{j,m} \widetilde{\mathbf{D}_m \mathbf{Z}_m}$ for $1 \leq j \leq n$, see Figures 6.11a and 6.12a. Then, each MPI process can locally build $\mathbf{E}_{j,m} = \widetilde{\mathbf{D}_j \mathbf{Z}_j}^T \mathbf{A}_{j,m} \widetilde{\mathbf{D}_m \mathbf{Z}_m}$ as illustrated in Figures 6.11b and 6.12b. Remark that we never store completely $\mathbf{AZ} \in \mathbb{C}^{N \times N_0}$ where $N_0 = \sum_{m=1}^n N_{\text{loc},m}$ is the total number of chosen eigenvectors to span the coarse space. We only compute a block in each row that is directly used to compute a block in $\mathbf{E} \in \mathbb{C}^{N_0 \times N_0}$.

Finally, all the elements of the preconditioner, i.e. the diagonal block associated with the subdomain with overlap, the local contributions to the coarse space $\mathbf{D}_m \mathbf{Z}_m$ and $\mathbf{E}$ are passed to HPDDM where they are used when applying the DDM preconditioner in iterative solvers. It can actually apply several types of Schwartz methods (additive or restricted) using several coarse correction (Additive as in Equation (5.5), Deflated as in Equation (5.19) or Balanced as in Equation (5.18)).

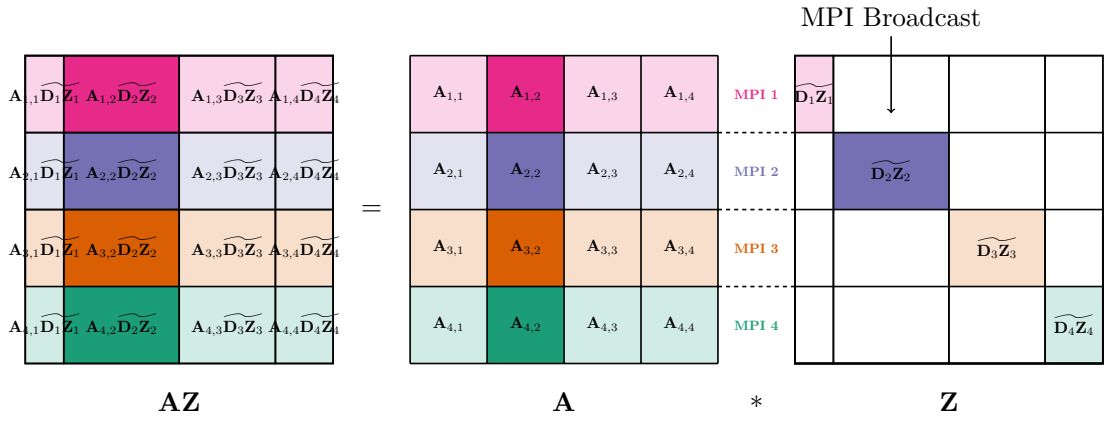


(a) Computation of the blocks of **AZ**

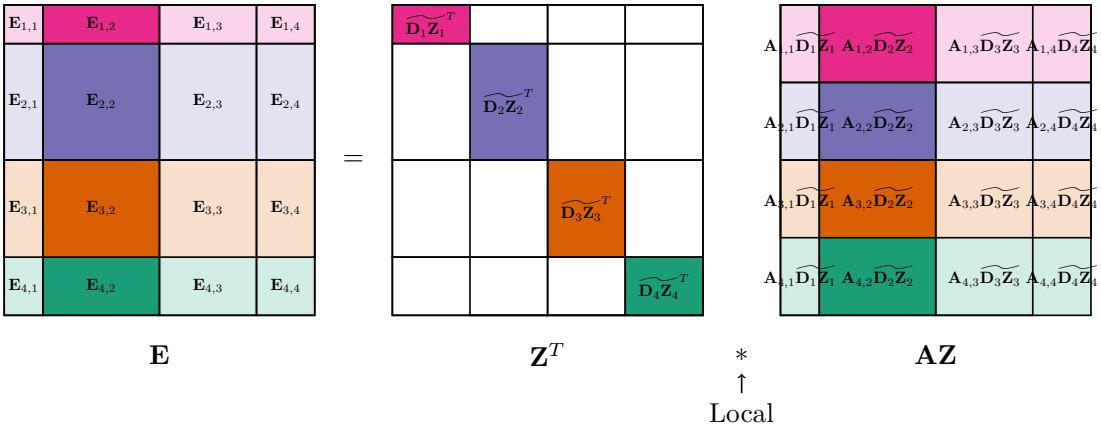


(b) Computation of the blocks of **E**

Figure 6.11 – Building the first block column of the coarse component.



(a) Computation of the blocks of **AZ**



(b) Computation of the blocks of **E**

Figure 6.12 – Building the second block column of the coarse component.

6.4 Perspectives

Htool proved to be efficient enough for our test cases (see Sections 5.4 and 5.5) and quite easy to use (see an example in Appendix C). We mention here some software development perspectives for Htool. Besides all the technical enhancements we could do (better testing, refactoring some parts of the code, adding code coverage, ...), we want to point out some more mathematical perspectives for Htool. As noted in Section 5.5, we could add the possibility to deal with vector valued problems such as linear elasticity or electromagnetics. This means that we should be able to associate one geometric point to several degrees of freedom and the ACA approximation should be modified as in [133, 21]. We would like also to study different approaches to apply a "Dirichlet boundary" type condition on hierarchical matrices using similar techniques to pseudo-elimination and exact penalization. This would be particularly useful when dealing with the hypersingular operator whose energy space is $\widetilde{H}^{1/2}$. Because, in the case of screen problems for the hypersingular operator, there is no degree of freedom on the boundary of the screen.

Finally, we point out improvements that can be made on two parts of the library.

6.4.1 Clustering and block-tree

One point where we could improve Htool would be to change how we handle clustering to offer the same modularity as for low-rank approximations. It would allow adding other clustering techniques using bounding boxes as described in [75, Section 5.4].

We focused on $\mathcal{H}$-matrix-vector and $\mathcal{H}$-matrix-matrix products because we are interested in solving linear systems with iterative solvers, but it could be interesting to develop the algebra of hierarchical matrices. In particular, we could add the possibility to sum or multiply two hierarchical matrices in Htool, but also to factorize a hierarchical matrix using $\mathcal{H}$-LU or $\mathcal{H}$-QR as described in [12, Sections 2.5, 2.9 and 2.10]. Then, the $\mathcal{H}$-LU factorization could be used when building the diagonal blocks associated with the one-level DDM preconditioner, see Remark 6.10.

6.4.2 Low-rank approximation

We can take advantage of the modularity of the class LowRankMatrix to offer other low-rank approximations. In particular, it has been shown in [69] that ACA can have bad performances in certain situations and the author introduced a variant called ACA+ that relies on a different heuristic in the choice of the pivots.

Other approaches have also been introduced using pseudo-skeleton approximations (see Section 6.1.3.1), which is also called CUR decomposition. For example, CUR-GS introduced in [9] provide such an approximation in linear-time and relies on a *Geometric Sampling* to choose the pivots.

There exists also randomized algorithms to provide a low-rank approximation of a given matrix, see [120] for a review on these methods. Htool would be a good framework to compare all these low-rank approximation on a same problem.

Finally, recompression techniques introduced in [69] could be implemented to improve the efficiency of the approximation and lower the storage cost.

Bibliography

Bibliography

[1] Mark Ainsworth and Christian Glusa. "Towards an Efficient Finite Element Method for the Integral Fractional Laplacian on Polygonal Domains". In: *Contemporary Computational Mathematics - A Celebration of the 80th Birthday of Ian Sloan*. Springer International Publishing, 2018, pp. 17–57. DOI: 10.1007/978-3-319-72456-0_2 (cit. on p. 167).

[2] Mark Ainsworth and Benqi Guo. "An additive Schwarz preconditioner for p-version boundary element approximation of the hypersingular operator in three dimensions". In: *Numerische Mathematik* 85.3 (2000), pp. 343–366. ISSN: 0945-3245. DOI: 10.1007/s002110000134. URL: http://dx.doi.org/10.1007/s002110000134 (cit. on p. 91).

[3] Guillaume Alléon, Michele Benzi, and Luc Giraud. "Sparse approximate inverse preconditioning for dense linear systems arising in computational electromagnetics". In: *Numerical Algorithms* 16.1 (1997), pp. 1–15. DOI: 10.1023/a:1019170609950 (cit. on p. 33).

[4] François Alouges and Matthieu Aussal. "The sparse cardinal sine decomposition and its application for fast numerical convolution". In: *Numerical Algorithms* 70.2 (Jan. 2015), pp. 427–448. DOI: 10.1007/s11075-014-9953-6 (cit. on pp. 5, 10, 32, 126).

[5] Bradley Alpert et al. "Wavelet-like bases for the fast solution of second-kind integral equations". In: *SIAM Journal on Scientific Computing* 14.1 (1993), pp. 159–184. DOI: 10.1137/0914010. eprint: http://dx.doi.org/10.1137/0914010. URL: http://dx.doi.org/10.1137/0914010 (cit. on pp. 5, 10, 32, 126).

[6] Edward Anderson. *LAPACK Users' Guide*. Society for Industrial and Applied Mathematics, Jan. 1, 1987. 429 pp. ISBN: 0898714478. URL: https://www.ebook.de/de/product/35214336/e_anderson_lapack_users_guide.html (cit. on pp. 105, 126, 145).

[7] Markus Aurada et al. "Energy norm based error estimators for adaptive BEM for hypersingular integral equations". In: *Applied Numerical Mathematics* 95 (Sept. 2015), pp. 15–35. DOI: 10.1016/j.apnum.2013.12.004 (cit. on p. 94).

[8] Markus Aurada et al. "Local inverse estimates for non-local boundary integral operators". In: *Mathematics of Computation* 86.308 (Apr. 2017), pp. 2651–2686. DOI: 10.1090/mcom/3175 (cit. on pp. 94, 113, 114).

[9] Alan Ayala, Xavier Claeys, and Laura Grigori. *Linear-time CUR approximation of BEM matrices*. Research Report RR-9208. INRIA PARIS, Oct. 2018. URL: https://hal.inria.fr/hal-01893036 (cit. on p. 150).

[10] Alvin Bayliss, Charles I Goldstein, and Eli Turkel. "An iterative method for the Helmholtz equation". In: *Journal of Computational Physics* 49.3 (Mar. 1983), pp. 443–457. DOI: 10.1016/0021-9991(83)90139-0 (cit. on p. 120).

[11] Mario Bebendorf. "Approximation of boundary element matrices". In: *Numerische Mathematik* 86.4 (2000), pp. 565–589. ISSN: 0945-3245. DOI: 10.1007/PL00005410. URL: http://dx.doi.org/10.1007/PL00005410 (cit. on pp. 5, 10, 32, 126, 130).

[12] Mario Bebendorf. *Hierarchical matrices: A Means to Efficiently Solve Elliptic Boundary Value Problems*. Vol. 63. Lecture Notes in Computational Science and Engineering. Springer-Verlag, Berlin, 2008 (cit. on pp. 5, 10, 32, 104, 126, 130–132, 150).

[13] Mario Bebendorf and Sergej Rjasanow. "Adaptive Low-Rank Approximation of Collocation Matrices". In: *Computing* 70.1 (2003), pp. 1–24. ISSN: 1436-5057. DOI: 10.1007/s00607-002-1469-6. URL: http://dx.doi.org/10.1007/s00607-002-1469-6 (cit. on pp. 5, 10, 32, 126, 130).

[14] Jöran Bergh and Jörgen Löfström. *Interpolation Spaces: an Introduction*. Vol. 223. Grundlehren der mathematischen Wissenschaften : A Series of Comprehensive Studies in Mathematics. Springer Berlin Heidelberg, 1976. DOI: 10.1007/978-3-642-66451-9 (cit. on p. 173).

[15] L. Susan Blackford et al. "An updated set of basic linear algebra subprograms (BLAS)". In: *ACM Transactions on Mathematical Software* 28.2 (June 2002), pp. 135–151. DOI: 10.1145/567806.567807 (cit. on p. 126).

[16] Andrea Bonito, Wenyu Lei, and Joseph E Pasciak. "Numerical approximation of the integral fractional Laplacian". In: *arXiv preprint arXiv:1707.04290* (2017) (cit. on p. 167).

[17] Steffen Börm. "Data-sparse approximation of non-local operators by $\mathcal{H}^2$-matrices". In: *Linear Algebra and its Applications* 422.2 (2007), pp. 380–403. ISSN: 0024-3795. DOI: http://dx.doi.org/10.1016/j.laa.2006.10.021. URL: http://www.sciencedirect.com/science/article/pii/S0024379506004745 (cit. on pp. 5, 10, 32, 126).

[18] Steffen Börm and Lars Grasedyck. "Hybrid cross approximation of integral operators". In: *Numerische Mathematik* 101.2 (2005), pp. 221–249. ISSN: 0945-3245. DOI: 10.1007/s00211-005-0618-1. URL: http://dx.doi.org/10.1007/s00211-005-0618-1 (cit. on pp. 5, 10, 32, 126).

[19] Steffen Börm, Lars Grasedyck, and Wolfgang Hackbush. "Hierarchical Matrices". June 2003. URL: https://www.mis.mpg.de/preprints/ln/lecturenote-2103.pdf (cit. on pp. 130, 131, 137).

[20] Xiao-Chuan Cai and Olof B. Widlund. "Domain decomposition algorithms for indefinite elliptic problems". In: *SIAM Journal on Scientific and Statistical Computing* 13.1 (1992), pp. 243–258 (cit. on p. 120).

[21] Stéphanie Chaillat, Luca Desiderio, and Patrick Ciarlet. "Theory and implementation of H-matrix based iterative and direct solvers for Helmholtz and elastodynamic oscillatory kernels". In: *Journal of Computational Physics* 351 (Dec. 2017), pp. 165–186. DOI: 10.1016/j.jcp.2017.09.013 (cit. on p. 150).

[22] Simon N. Chandler-Wilde, David P. Hewett, and Andrea Moiola. "Interpolation of Hilbert and Sobolev spaces: quantitative estimates and counterexamples". In: *Mathematika* 61.2 (Nov. 2014), pp. 414–443. DOI: 10.1112/s0025579314000278 (cit. on p. 173).

[23] Yu Chang and Roger F Harrington. "A surface formulation for characteristic modes of material bodies". In: *Antennas and Propagation, IEEE Transactions on* 25.6 (1977), pp. 789–795 (cit. on pp. 4, 9, 56, 61).

[24] Zhiming Chen and Xueshuang Xiang. "A Source Transfer Domain Decomposition Method for Helmholtz Equations in Unbounded Domain". In: *SIAM Journal on Numerical Analysis* 51.4 (Jan. 2013), pp. 2331–2356. DOI: 10.1137/130917144 (cit. on p. 123).

[25] Cédric Chevalier and François Pellegrini. "PT-Scotch: A tool for efficient parallel graph ordering". In: *Parallel Computing* 34.6-8 (July 2008), pp. 318–331. DOI: 10.1016/j.parco.2007.12.001 (cit. on pp. 40, 88).

[26] Snorre H. Christiansen and Jean-Claude Nédélec. "A Preconditioner for the Electric Field Integral Equation Based on Calderon Formulas". In: *SIAM Journal on Numerical Analysis* 40.3 (2002), pp. 1100–1135. DOI: 10.1137/S0036142901388731. eprint: https://doi.org/10.1137/S0036142901388731. URL: https://doi.org/10.1137/S0036142901388731 (cit. on p. 27).

[27] Snorre Harald Christiansen and Jean-Claude Nédélec. "Des préconditionneurs pour la résolution numérique des équations intégrales de frontière de l'acoustique". In: *Comptes Rendus de l'Académie des Sciences-Series I-Mathematics* 330.7 (2000), pp. 617–622 (cit. on p. 62).

[28] Snorre Harald Christiansen and Jean-Claude Nédélec. "Des préconditionneurs pour la résolution numérique des équations intégrales de frontière de l'électromagnétisme". In: *Comptes Rendus de l'Académie des Sciences-Series I-Mathematics* 331.9 (2000), pp. 733–738 (cit. on p. 62).

[29] Xavier Claeys. "A single trace integral formulation of the second kind for acoustic scattering". In: *ETH, Seminar of Applied Mathematics Research*. 2011, pp. 2011–14 (cit. on p. 59).

[30] Xavier Claeys. "Boundary integral equations of time harmonic wave scattering at complex structures". Habilitation à diriger des recherches en mathématiques et en informatique. Université Pierre et Marie Curie - Paris 6 & CNRS, Feb. 2016 (cit. on p. 58).

[31] Xavier Claeys. "Essential spectrum of local multi-trace boundary integral operators". In: *IMA Journal of Applied Mathematics* 81.6 (2016), pp. 961–983. ISSN: 0272-4960 (cit. on pp. 56, 63–67).

[32] Xavier Claeys. "Quasi-local multitrace boundary integral formulations". In: *Numerical Methods for Partial Differential Equations. An International Journal* 31.6 (2015), pp. 2043–2062. ISSN: 0749-159X. DOI: 10.1002/num.21981. URL: http://dx.doi.org/10.1002/num.21981 (cit. on p. 63).

[33] Xavier Claeys, Victorita Dolean, and Martin J. Gander. "An introduction to multi-trace formulations and associated domain decomposition solvers". In: *Applied Numerical Mathematics* 135 (Jan. 2019), pp. 69–86. DOI: 10.1016/j.apnum.2018.07.006 (cit. on pp. 4, 9, 10, 56, 68).

[34] Xavier Claeys, Victorita Dolean, and Martin J. Gander. "An introduction to Multitrace Formulations and Associated Domain Decomposition Solvers". working paper or preprint. Oct. 2015. URL: https://hal.archives-ouvertes.fr/hal-01207193 (cit. on p. 69).

[35] Xavier Claeys and Ralf Hiptmair. *Boundary integral formulation of the first kind for acoustic scattering by composite structures*. Tech. rep. 2011-45. ETH Zürich, July 2011 (cit. on p. 63).

[36] Xavier Claeys and Ralf Hiptmair. "Integral equations for acoustic scattering by partially impenetrable composite objects". In: *Integral Equations and Operator Theory* 81.2 (2015), pp. 151–189. ISSN: 0378-620X. DOI: 10.1007/s00020-014-2197-y. URL: http://dx.doi.org/10.1007/s00020-014-2197-y (cit. on p. 65).

[37] Xavier Claeys and Ralf Hiptmair. "Multi-trace boundary integral formulation for acoustic scattering by composite structures". In: *Communications on Pure and Applied Mathematics* 66.8 (2013), pp. 1163–1201. ISSN: 0010-3640. DOI: 10.1002/cpa.21462. URL: http://dx.doi.org/10.1002/cpa.21462 (cit. on pp. 57, 60).

[38] Xavier Claeys, Ralf Hiptmair, and Carlos Jerez-Hanckes. "Multitrace boundary integral equations". In: *Direct and inverse problems in wave propagation and applications*. Vol. 14. Radon Ser. Comput. Appl. Math. De Gruyter, Berlin, 2013, pp. 51–100 (cit. on pp. 4, 9, 56, 61, 62).

[39] Xavier Claeys and Pierre Marchand. "Boundary integral multi-trace formulations and Optimised Schwarz Methods". working paper or preprint. Nov. 2018. URL: https://hal.inria.fr/hal-01921113 (cit. on p. 56).

[40] Lea Conen et al. "A coarse space for heterogeneous Helmholtz problems based on the Dirichlet-to-Neumann operator". working paper or preprint. June 2013. URL: https://hal.archives-ouvertes.fr/hal-00831347 (cit. on p. 123).

[41] Martin Costabel. "Boundary integral operators on Lipschitz domains: elementary results". In: *SIAM Journal on Mathematical Analysis* 19.3 (1988), pp. 613–626. ISSN: 0036-1410. DOI: 10.1137/0519043. URL: http://dx.doi.org/10.1137/0519043 (cit. on pp. 24, 30).

[42] Eric Darve. "The Fast Multipole Method I: Error Analysis and Asymptotic Complexity". In: *SIAM Journal on Numerical Analysis* 38.1 (2000), pp. 98–128. DOI: 10.1137/S0036142999330379. eprint: http://dx.doi.org/10.1137/S0036142999330379. URL: http://dx.doi.org/10.1137/S0036142999330379 (cit. on pp. 5, 10, 32, 126).

[43] Eric Darve. "The fast multipole method: numerical implementation". In: *Journal of Computational Physics* 160.1 (2000), pp. 195–240 (cit. on pp. 5, 10, 32, 126).

[44] Robert H. Dennard et al. "Design of ion-implanted MOSFET's with very small physical dimensions". In: *IEEE Journal of Solid-State Circuits* 9.5 (Oct. 1974), pp. 256–268. DOI: 10.1109/jssc.1974.1050511 (cit. on pp. 2, 7).

[45] Victorita Dolean, Pierre Jolivet, and Frédéric Nataf. *An introduction to domain decomposition methods*. Algorithms, theory, and parallel implementation. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2015, pp. x+238. ISBN: 978-1-611974-05-8. DOI: 10.1137/1.9781611974065.ch1. URL: http://dx.doi.org/10.1137/1.9781611974065.ch1 (cit. on pp. 36, 38, 44, 47, 49–54, 94, 98, 115).

[46] Carl Eckart and Gale Young. "The approximation of one matrix by another of lower rank". In: *Psychometrika* 1.3 (Sept. 1936), pp. 211–218. DOI: 10.1007/bf02288367 (cit. on p. 129).

[47] Evridiki Efstathiou and Martin J. Gander. "Why Restricted Additive Schwarz Converges Faster than Additive Schwarz". In: *BIT Numerical Mathematics* 43.5 (2003), pp. 945–959. DOI: 10.1023/b:bitn.0000014563.33622.1d (cit. on p. 45).

[48] Björn Engquist and Lexing Ying. "Sweeping preconditioner for the Helmholtz equation: Hierarchical matrix representation". In: *Communications on Pure and Applied Mathematics* 64.5 (Feb. 2011), pp. 697–735. DOI: 10.1002/cpa.20358 (cit. on p. 123).

[49] Björn Engquist and Lexing Ying. "Sweeping Preconditioner for the Helmholtz Equation: Moving Perfectly Matched Layers". In: *Multiscale Modeling & Simulation* 9.2 (Apr. 2011), pp. 686–710. DOI: 10.1137/100804644 (cit. on p. 123).

[50] Yogi Ahmad Erlangga. "Advances in Iterative Methods and Preconditioners for the Helmholtz Equation". In: *Archives of Computational Methods in Engineering* 15.1 (Dec. 2007), pp. 37–66. DOI: 10.1007/s11831-007-9013-7 (cit. on p. 120).

[51] Yogi Ahmad Erlangga, Cornelis W. Oosterlee, and Cornelis Vuik. "A Novel Multigrid Based Preconditioner For Heterogeneous Helmholtz Problems". In: *SIAM Journal on Scientific Computing* 27.4 (Jan. 2006), pp. 1471–1492. DOI: 10.1137/040615195 (cit. on p. 120).

[52] Yogi Ahmad Erlangga, Cornelis Vuik, and Cornelis W. Oosterlee. "On a class of preconditioners for solving the Helmholtz equation". In: *Applied Numerical Mathematics* 50.3-4 (Sept. 2004), pp. 409–425. DOI: 10.1016/j.apnum.2004.01.009 (cit. on p. 120).

[53] Alexandre Ern and Jean-Luc Guermond. *Theory and Practice of Finite Elements*. Vol. 159. Springer New York, 2004. DOI: 10.1007/978-1-4757-4355-5 (cit. on pp. 39, 40, 87).

[54] Oliver G. Ernst and Martin J. Gander. "Why it is Difficult to Solve Helmholtz Problems with Classical Iterative Methods". In: *Lecture Notes in Computational Science and Engineering*. Springer Berlin Heidelberg, Aug. 2011, pp. 325–363. DOI: 10.1007/978-3-642-22061-6_10 (cit. on p. 120).

[55] Charbel Farhat, Antonini Macedo, and Michel Lesoinne. "A two-level domain decomposition method for the iterative solution of high frequency exterior Helmholtz problems". In: *Numerische Mathematik* 85.2 (Apr. 2000), pp. 283–308. DOI: 10.1007/pl00005389 (cit. on p. 123).

[56] Michael Feischl et al. "Adaptive boundary element methods". In: *Archives of Computational Methods in Engineering* 22.3 (2015), pp. 309–389 (cit. on pp. 39, 86, 93, 113).

[57] Martin J. Gander. "Schwarz methods over the course of time". In: *Electronic Transactions on Numerical Analysis* 31 (2008), pp. 228–255. ISSN: 1068-9613 (cit. on pp. 35, 36).

[58] Martin J. Gander, Frédéric Magoulès, and Frédéric Nataf. "Optimized Schwarz Methods without Overlap for the Helmholtz Equation". In: *SIAM Journal on Scientific Computing* 24.1 (Jan. 2002), pp. 38–60. DOI: 10.1137/s1064827501387012 (cit. on p. 123).

[59] Martin J. Gander and Hui Zhang. "A Class of Iterative Solvers for the Helmholtz Equation: Factorizations, Sweeping Preconditioners, Source Transfer, Single Layer Potentials, Polarized Traces, and Optimized Schwarz Methods". In: *SIAM Review* 61.1 (Jan. 2019), pp. 3–76. DOI: 10.1137/16m109781x (cit. on p. 123).

[60] Christophe Geuzaine and Jean-François Remacle. "Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities". In: *International Journal for Numerical Methods in Engineering* 79.11 (Sept. 2009), pp. 1309–1331. DOI: 10.1002/nme.2579 (cit. on pp. 74, 105, 140).

[61] Ibtihel Ben Gharbia et al. "Fast Solution of Boundary Integral Equations for Elasticity Around a Crack Network: A Comparative Study". In: *ESAIM: Proceedings and Surveys* 63 (2018). Ed. by L. Grigori, C. Japhet, and P. Moireau, pp. 135–151. DOI: 10.1051/proc/201863135 (cit. on p. 127).

[62] Martin B. van Gijzen, Yogi Ahmad Erlangga, and Cornelis Vuik. "Spectral Analysis of the Discrete Helmholtz Operator Preconditioned with a Shifted Laplacian". In: *SIAM Journal on Scientific Computing* 29.5 (Jan. 2007), pp. 1942–1958. DOI: 10.1137/060661491 (cit. on p. 120).

[63] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, 1996. ISBN: 0-8018-5413-X. URL: https://www.amazon.com/Computations-Hopkins-Studies-Mathematical-Sciences/dp/080185413X?SubscriptionId=AKIAIOBINVZYXZQZ2U3A&tag=chimbori05-20&linkCode=xm2&camp=2025&creative=165953&creativeASIN=080185413X (cit. on pp. 128, 129).

[64] Sergei A. Goreinov, Eugene E. Tyrtyshnikov, and Nickolai L. Zamarashkin. "A theory of pseudoskeleton approximations". In: *Linear Algebra and its Applications* 261.1-3 (Aug. 1997), pp. 1–21. DOI: 10.1016/s0024-3795(96)00301-1 (cit. on p. 129).

[65] Ivan G. Graham. "Finite elements on degenerate meshes: inverse-type inequalities and applications". In: *IMA Journal of Numerical Analysis* 25.2 (Jan. 2005), pp. 379–407. DOI: 10.1093/imanum/drh017 (cit. on p. 114).

[66] Ivan G. Graham, Euan A. Spence, and Eero Vainikko. "Domain decomposition preconditioning for high-frequency Helmholtz problems with absorption". In: *Mathematics of Computation* 86.307 (July 8, 2015), pp. 2089–2127. DOI: 10.1090/mcom/3190 (cit. on pp. 116, 123).

[67] Ivan G. Graham, Euan A. Spence, and Eero Vainikko. "Recent Results on Domain Decomposition Preconditioning for the High-Frequency Helmholtz Equation Using Absorption". In: *Modern Solvers for Helmholtz Problems*. Springer International Publishing, 2017, pp. 3–26. DOI: 10.1007/978-3-319-28832-1_1 (cit. on p. 123).

[68] Ivan G. Graham, Euan A. Spence, and Jun Zou. "Domain Decomposition with local impedance conditions for the Helmholtz equation". In: (June 10, 2018). arXiv: http://arxiv.org/abs/1806.03731v3 [math.NA] (cit. on pp. 116, 123).

[69] Lars Grasedyck. "Adaptive Recompression of $\mathcal{H}$-Matrices for BEM". In: *Computing* 74.3 (Dec. 2004), pp. 205–223. DOI: 10.1007/s00607-004-0103-1 (cit. on p. 150).

[70] Leslie Greengard et al. "Accelerating fast multipole methods for the Helmholtz equation at low frequencies". In: *IEEE Computational Science and Engineering* 5.3 (1998), pp. 32–38. DOI: 10.1109/99.714591 (cit. on pp. 5, 10, 32, 126).

[71] Leslie Greengard et al. "Fast direct solvers for integral equations in complex three-dimensional domains". In: *Acta Numerica* 18 (May 2009), p. 243. DOI: 10.1017/s0962492906410011 (cit. on pp. 33, 126).

[72] Michael Griebel and Peter Oswald. "On the abstract theory of additive and multiplicative Schwarz algorithms". In: *Numerische Mathematik* 70.2 (1995), pp. 163–180 (cit. on p. 45).

[73] Pierre Grisvard. *Elliptic Problems in Nonsmooth Domains*. Society for Industrial and Applied Mathematics, Jan. 2011. DOI: 10.1137/1.9781611972030 (cit. on p. 16).

[74] Wolfgang Hackbusch. "A Sparse Matrix Arithmetic Based on $\mathcal{H}$ -Matrices. Part I: Introduction to $\mathcal{H}$ -Matrices". In: *Computing* 62.2 (1999), pp. 89–108. ISSN: 1436-5057. DOI: 10.1007/s006070050015. URL: http://dx.doi.org/10.1007/s006070050015 (cit. on pp. 5, 10, 32, 126).

[75] Wolfgang Hackbusch. *Hierarchical Matrices: Algorithms and Analysis*. Vol. 49. Springer Series in Computational Mathematics. Springer-Verlag, Berlin, 2015. DOI: 10.1007/978-3-662-47324-5 (cit. on pp. 5, 10, 32, 33, 104, 126–128, 150).

[76] Wolfgang Hackbusch and Boris N. Khoromskij. "A Sparse $\mathcal{H}$-Matrix Arithmetic. Part II: Application to Multi-Dimensional Problems". In: *Computing* 64.1 (2000), pp. 21–47. ISSN: 1436-5057. DOI: 10.1007/PL00021408. URL: http://dx.doi.org/10.1007/PL00021408 (cit. on pp. 5, 10, 32, 126).

[77] Wolfgang Hackbusch, Boris N. Khoromskij, and Stefan A. Sauter. "On $\mathcal{H}^2$-Matrices". In: *Lectures on applied mathematics : proceedings of the symposium organized by the Sonderforschungsbereich 438 on the occasion of Karl-Heinz Hoffmann's 60th birthday, Munich, June 30 - July 1, 1999*. Ed. by Hans-Joachim Bungartz, Ronald H. W. Hoppe, and Christoph Zenger. Berlin [u. a.]: Springer, 2000, pp. 9–29. ISBN: 3-540-66734-2. URL: http://www.mis.mpg.de/de/publications/preprints/1999/prepr1999-50.html (cit. on pp. 5, 10, 32, 126).

[78] Wolfgang Hackbusch and Zenon Paul Nowak. "On the fast matrix multiplication in the boundary element method by panel clustering". In: *Numerische Mathematik* 54.4 (1989), pp. 463–491. ISSN: 0945-3245. DOI: 10.1007/BF01396324. URL: http://dx.doi.org/10.1007/BF01396324 (cit. on pp. 5, 10, 32, 126).

[79] Wolfgang Hackbusch and Stefan A. Sauter. "On the efficient use of the Galerkin-method to solve Fredholm integral equations". In: *Applications of Mathematics* 38.4-5 (1993), pp. 301–322. ISSN: 0862-7940. URL: http://dml.cz/dmlcz/104558 (cit. on pp. 5, 10, 32, 126).

[80] Manfred Hahne and Ernst P. Stephan. "Schwarz iterations for the efficient solution of screen problems with boundary elements". In: *Computing* 56 (Mar. 1, 1996), pp. 61–85. ISSN: 1436-5057. DOI: 10.1007/BF02238292. URL: http://dx.doi.org/10.1007/BF02238292 (cit. on pp. 3, 4, 8, 10, 55, 85, 86).

[81] Houde Han. "The boundary integro-differential equations of three-dimensional Neumann problem in linear elasticity". In: *Numerische Mathematik* 68.2 (July 1994), pp. 269–281. ISSN: 0945-3245. DOI: 10.1007/s002110050061. URL: https://doi.org/10.1007/s002110050061 (cit. on p. 113).

[82] Frédéric Hecht. "New development in FreeFem++". In: *Journal of numerical mathematics* 20.3-4 (2012), pp. 251–266 (cit. on p. 127).

[83] Magnus Rudolph Hestenes and Eduard Stiefel. "Methods of conjugate gradients for solving linear systems". In: *Journal of Research of the National Bureau of Standards* 49.6 (1952) (cit. on pp. 33, 78).

[84] Norbert Heuer. "Additive Schwarz method for the p-version of the boundary element method for the single layer potential operator on a plane screen". In: *Numerische Mathematik* 88.3 (2001), pp. 485–511. ISSN: 0945-3245. DOI: 10.1007/s211-001-8012-7. URL: http://dx.doi.org/10.1007/s211-001-8012-7 (cit. on p. 86).

[85] Norbert Heuer. "Additive Schwarz methods for indefinite hypersingular integral equations in $\mathbb{R}^3$—the p-version". In: *Applicable Analysis* 72.3-4 (1999), pp. 411–437 (cit. on p. 120).

[86] Norbert Heuer. "Additive Schwarz Methods for Weakly Singular Integral Equations In $\mathbb{R}^3$ — the p-Version". In: *Boundary Elements: Implementation and Analysis of Advanced Algorithms: Proceedings of the Twelfth GAMM-Seminar Kiel, January 19–21, 1996*. Ed. by Wolfgang Hackbusch and Gabriel Wittum. Wiesbaden: Vieweg+Teubner Verlag, 1996, pp. 126–135. ISBN: 978-3-322-89941-5. DOI: 10.1007/978-3-322-89941-5_10. URL: http://dx.doi.org/10.1007/978-3-322-89941-5_10 (cit. on pp. 33, 86).

[87] Norbert Heuer. "An iterative substructuring method for the p -version of the boundary element method for hypersingular integral operators in three dimensions". In: *Numerische Mathematik* 79.3 (1998). Attention, cf erratum Heuer2001c, pp. 371–396. ISSN: 0945-3245. DOI: 10.1007/s002110050344. URL: http://dx.doi.org/10.1007/s002110050344 (cit. on p. 86).

[88] Norbert Heuer. "Efficient Algorithms for the p-Version of the Boundary Element Method". In: *Journal of Integral Equations and Applications* 8.3 (Sept. 1996), pp. 337–360. DOI: 10.1216/jiea/1181075956. URL: http://dx.doi.org/10.1216/jiea/1181075956 (cit. on pp. 4, 10, 86).

[89] Norbert Heuer. "Preconditioners for the p-version of the boundary element Galerkin method in $\mathbb{R}^3$". PhD thesis. Habilitation Thesis, Institut f ur Angewandte Mathematik, Universit at Hannover, Germany, 1998 (cit. on p. 86).

[90] Norbert Heuer and Ernst P. Stephan. "An additive Schwarz method for the h-p version of the boundary element method for hypersingular integral equations in $\mathbb{R}^3$". In: *IMA Journal of Numerical Analysis* 21.1 (2001), pp. 265–283. DOI: 10.1093/imanum/21.1.265. eprint: http://imajna.oxfordjournals.org/content/21/1/265.full.pdf+html. URL: http://imajna.oxfordjournals.org/content/21/1/265.abstract (cit. on p. 86).

[91] Norbert Heuer and Ernst P. Stephan. "An Overlapping Domain Decomposition Preconditioner for High Order BEM with Anisotropic Elements". In: *Advances in Computational Mathematics* 19.1 (2003), pp. 211–230. ISSN: 1572-9044. DOI: 10.1023/A:1022899205324. URL: http://dx.doi.org/10.1023/A:1022899205324 (cit. on p. 86).

[92] Norbert Heuer and Ernst P. Stephan. "Iterative Substructuring for Hypersingular Integral Equations in $\mathbb{R}^3$". In: *SIAM Journal on Scientific Computing* 20.2 (1998), pp. 739–749. DOI: 10.1137/S1064827596311797. eprint: http://dx.doi.org/10.1137/S1064827596311797. URL: http://dx.doi.org/10.1137/S1064827596311797 (cit. on p. 86).

[93] Norbert Heuer, Ernst P. Stephan, and Thanh Tran. "Multilevel additive Schwarz method for the h-p version of the Galerkin boundary element method". In: *Mathematics of Computation* 67.222 (1998), pp. 501–518. ISSN: 0025-5718. DOI: 10.1090/S0025-5718-98-00926-0. URL: http://dx.doi.org/10.1090/S0025-5718-98-00926-0 (cit. on p. 86).

[94] Ralf Hiptmair. "Operator preconditioning". In: *Computers & Mathematics with Applications* 52.5 (2006), pp. 699–706. ISSN: 0898-1221. DOI: 10.1016/j.camwa.2006.10.008. URL: http://dx.doi.org/10.1016/j.camwa.2006.10.008 (cit. on pp. 27, 33, 62).

[95] Ralf Hiptmair and Carlos Jerez-Hanckes. "Multiple traces boundary integral formulation for Helmholtz transmission problems". In: *ETH, Seminar of Applied Mathematics Research Report*. 2010-30. 2010 (cit. on p. 62).

[96] Ralf Hiptmair and Carlos Jerez-Hanckes. "Multiple traces boundary integral formulation for Helmholtz transmission problems". In: *Advances in Computational Mathematics* 37.1 (2012), pp. 39–91. ISSN: 1019-7168. DOI: 10.1007/s10444-011-9194-3. URL: http://dx.doi.org/10.1007/s10444-011-9194-3 (cit. on pp. 56, 63, 71).

[97] Ralf Hiptmair, Carlos Jerez-Hanckes, and Carolina Urzua-Torres. "Mesh-independent operator preconditioning for boundary elements on open curves". In: *SIAM Journal on Numerical Analysis* 52.5 (2014), pp. 2295–2314 (cit. on p. 116).

[98] Ralf Hiptmair, Carlos Jerez-Hanckes, and Carolina Urzúa-Torres. *Optimal Operator preconditioning for hypersingular operator over 3D screens*. Tech. rep. 2016-09. Switzerland: Seminar for Applied Mathematics, ETH Zürich, 2016. URL: https://www.sam.math.ethz.ch/sam_reports/reports_final/reports2016/2016-09.pdf (cit. on pp. 106, 116).

[99] Ralf Hiptmair, Carlos Jerez-Hanckes, and Carolina Urzúa-Torres. *Optimal Operator preconditioning for weakly singular operator over 3D screens*. Tech. rep. 2017-13. Switzerland: Seminar for Applied Mathematics, ETH Zürich, 2017. URL: https://www.sam.math.ethz.ch/sam_reports/reports_final/reports2017/2017-13.pdf (cit. on p. 116).

[100] Ralf Hiptmair et al. "Domain decomposition for boundary integral equations via local multi-trace formulations". In: *Domain Decomposition Methods in Science and Engineering XXI*. Springer, 2014, pp. 43–57 (cit. on p. 63).

[101] Torsten Hoefler et al. "The scalable process topology interface of MPI 2.2". In: *Concurrency and Computation: Practice and Experience* 23.4 (Aug. 2010), pp. 293–310. DOI: 10.1002/cpe.1643 (cit. on p. 76).

[102] George C. Hsiao and Wolfgang L. Wendland. *Boundary integral equations*. Vol. 164. Applied Mathematical Sciences. Springer-Verlag, Berlin, 2008, pp. xx+618. ISBN: 978-3-540-15284-2. DOI: 10.1007/978-3-540-68545-6. URL: http://dx.doi.org/10.1007/978-3-540-68545-6 (cit. on pp. 21, 30).

[103] Martin Huber and Joachim Schöberl. "Hybrid Domain Decomposition Solvers for the Helmholtz Equation". In: *Lecture Notes in Computational Science and Engineering*. Springer International Publishing, 2014, pp. 351–358. DOI: 10.1007/978-3-319-05789-7_32 (cit. on p. 123).

[104] Pierre Jolivet et al. "Scalable domain decomposition preconditioners for heterogeneous elliptic problems". In: *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis on - SC '13*. ACM Press, 2013. DOI: 10.1145/2503210.2503212 (cit. on pp. 104, 126).

[105] George Karypis and Vipin Kumar. "A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs". In: *SIAM Journal on Scientific Computing* 20.1 (Jan. 1998), pp. 359–392. DOI: 10.1137/s1064827595287997 (cit. on pp. 40, 88).

[106] Jung-Han Kimn and Marcus Sarkis. "Restricted overlapping balancing domain decomposition methods and restricted coarse problems for the Helmholtz problem". In: *Computer Methods in Applied Mechanics and Engineering* 196.8 (Jan. 2007), pp. 1507–1514. DOI: 10.1016/j.cma.2006.03.016 (cit. on p. 123).

[107] Jung-Han Kimn and Marcus Sarkis. "Shifted Laplacian RAS Solvers for the Helmholtz Equation". In: *Lecture Notes in Computational Science and Engineering*. Springer Berlin Heidelberg, 2013, pp. 151–158. DOI: 10.1007/978-3-642-35275-1_16 (cit. on p. 123).

[108] Jonathan Koomey et al. “Implications of Historical Trends in the Electrical Efficiency of Computing”. In: *IEEE Annals of the History of Computing* 33.3 (Mar. 2011), pp. 46–54. DOI: 10.1109/mahc.2010.28 (cit. on pp. 2, 7).

[109] Alistair L. Laird and Michael B. Giles. *Preconditioned iterative solution of the 2D Helmholtz equation*. Tech. rep. Oxford University Computing Laboratory, 2002 (cit. on p. 120).

[110] Ulrich Langer and Olaf Steinbach. “Boundary element tearing and interconnecting methods”. In: *Computing* 71.3 (2003), pp. 205–228. ISSN: 0010-485X. DOI: 10.1007/s00607-003-0018-2. URL: http://dx.doi.org/10.1007/s00607-003-0018-2 (cit. on pp. 4, 9, 56, 63).

[111] Jacques-Louis Lions and Enrico Magenes. *Non-homogeneous boundary value problems and applications : Vol. 1*. Vol. 181. Die Grundlehren der mathematischen Wissenschaften. Springer Berlin Heidelberg, 1972. DOI: 10.1007/978-3-642-65161-8 (cit. on p. 16).

[112] Pierre-Louis Lions. “On the Schwarz alternating method. I”. In: *Domain Decomposition Methods for Partial Differential Equations*. Ed. by Roland Glowinski et al. SIAM. 1988, pp. 1–42 (cit. on pp. 37, 45).

[113] Pierre-Louis Lions. “On the Schwarz alternating method. III: a variant for nonoverlapping subdomains”. In: *Domain Decomposition Methods for Partial Differential Equations*. Ed. by Tony Chan et al. Vol. 6. SIAM. 1989, pp. 202–223 (cit. on p. 37).

[114] Jian Liu and Jian-Ming Jin. “Scattering analysis of a large body with deep cavities”. In: *IEEE Transactions on Antennas and Propagation* 51.6 (2003), pp. 1157–1167 (cit. on p. 119).

[115] Benoît Lizé. “Résolution directe rapide pour les éléments finis de frontière en électromagnétisme et acoustique: $\mathcal{H}$-Matrices. Parallélisme et applications industrielles”. PhD thesis. Université Paris-Nord-Paris XIII, 2014 (cit. on pp. 127, 133).

[116] Mardochée Magolu monga Made. “Incomplete factorization-based preconditionings for solving the Helmholtz equation”. In: *International Journal for Numerical Methods in Engineering* 50.5 (2001), pp. 1077–1101. DOI: 10.1002/1097-0207(20010220)50:5<1077::aid-nme65>3.0.co;2-p (cit. on p. 120).

[117] Mardochée Magolu monga Made, Robert Beauwens, and Guy Warzée. “Preconditioning of discrete Helmholtz operators perturbed by a diagonal complex matrix”. In: *Communications in Numerical Methods in Engineering* 16.11 (2000), pp. 801–817. DOI: 10.1002/1099-0887(200011)16:11<801::aid-cnm377>3.0.co;2-m (cit. on p. 120).

[118] Jan Mandel and Bedřich Sousedík. “Coarse spaces over the ages”. In: *Domain decomposition methods in science and engineering XIX*. Springer, 2011, pp. 213–220 (cit. on pp. 36, 50).

[119] Pierre Marchand et al. “Two-level preconditioning for h-version boundary element approximation of hypersingular operator with GenEO”. working paper or preprint. July 2019. URL: https://hal.archives-ouvertes.fr/hal-02188771 (cit. on p. 86).

[120] Per-Gunnar Martinsson. “Randomized methods for matrix computations”. In: (July 6, 2016). arXiv: http://arxiv.org/abs/1607.01649v2 [math.NA] (cit. on p. 150).

[121] Per-Gunnar Martinsson and Vladimir Rokhlin. “A fast direct solver for boundary integral equations in two dimensions”. In: *Journal of Computational Physics* 205.1 (May 2005), pp. 1–23. DOI: 10.1016/j.jcp.2004.10.033 (cit. on pp. 33, 126).

[122] Aleksandr M. Matsokin and Sergey V. Nepomnyaschikh. “The Schwarz alternation method in a subspace”. In: *Izvestiya Vysshikh Uchebnykh Zavedenii. Matematika* 10 (1985), pp. 61–66. ISSN: 0021-3446 (cit. on p. 45).

[123] William McLean. *Strongly Elliptic Systems and Boundary Integral Equations*. Cambridge University Press, Cambridge, June 11, 2010, pp. xiv+357. 372 pp. ISBN: 0-521-66375-X. URL: http://www.ebook.de/de/product/3650114/william_charles_hector_mclean_strongly_elliptic_systems_and_boundary_integral_equations.html (cit. on pp. 14–17, 19, 173).

[124] William McLean and Olaf Steinbach. “Boundary element preconditioners for a hypersingular integral equation on an interval”. In: *Advances in Computational Mathematics* 11.4 (Dec. 1999), pp. 271–286. ISSN: 1572-9044. DOI: 10.1023/A:1018944530343. URL: https://doi.org/10.1023/A:1018944530343 (cit. on pp. 103, 170).

[125] William McLean and Thanh Tran. “A preconditioning strategy for boundary element Galerkin methods”. In: *Numerical Methods for Partial Differential Equations* 13.3 (May 1997), pp. 283–301. DOI: 10.1002/(sici)1098-2426(199705)13:3<283::aid-num4>3.0.co;2-j (cit. on p. 30).

[126] Leon Mirsky. “Symmetric gauge functions and unitarly invariant norms”. In: *The Quarterly Journal of Mathematics* 11.1 (1960), pp. 50–59. DOI: 10.1093/qmath/11.1.50 (cit. on p. 129).

[127] Frédéric Nataf. “Mathematical Analysis of Robustness of Two-Level Domain Decomposition Methods with respect to Approximate Coarse Solves”. working paper or preprint. Nov. 2017. URL: https://hal.archives-ouvertes.fr/hal-01573197 (cit. on p. 54).

[128] Frédéric Nataf, Francois Rogier, and Eric de Sturler. “Optimal interface conditions for domain decomposition methods”. In: *CMAP (Ecole Polytechnique)* 301 (1994), pp. 1–18 (cit. on pp. 38, 56, 71, 73).

[129] Sergey V. Nepomnyaschikh. “Mesh theorems on traces, normalizations of function traces and their inversion”. In: *Russian Journal of Numerical Analysis and Mathematical Modelling* 6.3 (1991), pp. 223–242 (cit. on p. 45).

[130] *NIST Digital Library of Mathematical Functions*. http://dlmf.nist.gov/, Release 1.0.21 of 2018-12-15. F. W. J. Olver, A. B. Olde Daalhuis, D. W. Lozier, B. I. Schneider, R. F. Boisvert, C. W. Clark, B. R. Miller and B. V. Saunders, eds. URL: http://dlmf.nist.gov/ (cit. on p. 23).

[131] Andrew J. Poggio and Edmund K. Miller. *Integral equation solutions of three-dimensional scattering problems*. MB Assoc., 1970 (cit. on pp. 4, 9, 56, 61).

[132] Sergej Rjasanow and Olaf Steinbach. *The Fast Solution of Boundary Integral Equations*. Mathematical and Analytical Techniques with Applications to Engineering. Springer Science & Business Media, May 11, 2007, pp. xii+279. ISBN: 978-0-387-34041-8. URL: http://www.ebook.de/de/product/6497790/sergej_rjasanow_olaf_steinbach_the_fast_solution_of_boundary_integral_equations.html (cit. on pp. 14, 18, 22, 104, 127, 130, 137, 139, 140).

[133] Sergej Rjasanow and Lucy Weggler. “Matrix valued adaptive cross approximation”. In: *Mathematical Methods in the Applied Sciences* 40.7 (Sept. 2016), pp. 2522–2531. DOI: 10.1002/mma.4174 (cit. on p. 150).

[134] Vladimir Rokhlin. “Rapid solution of integral equations of classical potential theory”. In: *Journal of Computational Physics* 60.2 (1985), pp. 187–207 (cit. on pp. 5, 10, 32, 126).

[135] Walter Rudin. *Functional Analysis*. McGraw-Hill Science/Engineering/Math, Jan. 1991. ISBN: 0070542368 (cit. on p. 67).

[136] Vyacheslav S. Rychkov. "On restrictions and extensions of the Besov and Triebel-Lizorkin spaces with respect to lipschitz domains". In: *Journal of the London Mathematical Society* 60.1 (1999), pp. 237–257 (cit. on p. 173).

[137] Youcef Saad and Martin H. Schultz. "GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems". In: *SIAM Journal on Scientific and Statistical Computing* 7.3 (July 1986), pp. 856–869. DOI: 10.1137/0907058 (cit. on pp. 33, 78).

[138] Yousef Saad. *Iterative Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics, Jan. 2003. DOI: 10.1137/1.9780898718003 (cit. on p. 33).

[139] Stefan A. Sauter and Christoph Schwab. *Boundary element methods*. Vol. 39. Springer Series in Computational Mathematics. Translated and expanded from the 2004 German original. Springer-Verlag, Berlin, 2011, pp. xviii+561. ISBN: 978-3-540-68092-5. DOI: 10.1007/978-3-540-68093-2. URL: http://dx.doi.org/10.1007/978-3-540-68093-2 (cit. on pp. 14, 16–18, 20–27, 29–33, 90, 91, 126).

[140] Robert Scheichl, Panayot S. Vassilevski, and Ludmil T. Zikatanov. "Weak Approximation Properties of Elliptic Projections with Functional Constraints". In: *Multiscale Modeling & Simulation* 9.4 (2011), pp. 1677–1699. DOI: 10.1137/110821639. eprint: https://doi.org/10.1137/110821639. URL: https://doi.org/10.1137/110821639 (cit. on pp. 43, 47).

[141] Oliver K. Smith. "Eigenvalues of a symmetric 3 × 3 matrix". In: *Communications of the ACM* 4.4 (Apr. 1961), p. 168. DOI: 10.1145/355578.366316 (cit. on p. 137).

[142] Nicole Spillane. "Robust domain decomposition methods for symmetric positive definite problems". Theses. Université Pierre et Marie Curie - Paris VI, Jan. 2014. URL: https://tel.archives-ouvertes.fr/tel-00958252 (cit. on p. 36).

[143] Nicole Spillane et al. "Abstract robust coarse spaces for systems of PDEs via generalized eigenproblems in the overlaps". In: *Numerische Mathematik* 126.4 (2014), pp. 741–770. ISSN: 0945-3245. DOI: 10.1007/s00211-013-0576-y. URL: http://dx.doi.org/10.1007/s00211-013-0576-y (cit. on pp. 43, 50).

[144] Elias M. Stein. *Singular integrals and differentiability properties of functions*. Vol. 30. Princeton university press, 2016 (cit. on p. 173).

[145] Olaf Steinbach. *Numerical approximation methods for elliptic boundary value problems*. Finite and boundary elements, Translated from the 2003 German original. Springer, New York, 2008, pp. xii+386. ISBN: 978-0-387-31312-2. DOI: 10.1007/978-0-387-68805-3. URL: http://dx.doi.org/10.1007/978-0-387-68805-3 (cit. on p. 127).

[146] Olaf Steinbach and Wolfgang L. Wendland. "The construction of some efficient preconditioners in the boundary element method". In: *Advances in Computational Mathematics* 9.1-2 (1998). Numerical treatment of boundary integral equations, pp. 191–216. ISSN: 1019-7168. DOI: 10.1023/A:1018937506719. URL: http://dx.doi.org/10.1023/A:1018937506719 (cit. on pp. 27, 33, 62, 103, 108, 170).

[147] Ernst P. Stephan. "Additive Schwarz methods for integral equations of the first kind". In: *MATHEMATICS OF FINITE ELEMENTS AND APPLICATIONS* 9 (1996), pp. 123–144 (cit. on p. 86).

[148] Ernst P. Stephan. "Boundary integral equations for screen problems in $\mathbb{R}^3$". In: *Integral Equations and Operator Theory* 10.2 (1987), pp. 236–257. ISSN: 1420-8989. DOI: 10.1007/BF01199079. URL: http://dx.doi.org/10.1007/BF01199079 (cit. on p. 32).

[149] Ersnt P. Stephan and Thanh Tran. "Domain decomposition algorithms for indefinite hypersingular integral equations: the h and p versions". In: *SIAM Journal on Scientific Computing* 19.4 (1998), pp. 1139–1153 (cit. on p. 120).

[150] Ersnt P. Stephan and Thanh Tran. "Domain decomposition algorithms for indefinite weakly singular integral equations: the h and p versions". In: *IMA journal of numerical analysis* 20.1 (2000), pp. 1–24 (cit. on p. 120).

[151] Christiaan C. Stolk. "A rapidly converging domain decomposition method for the Helmholtz equation". In: *Journal of Computational Physics* 241 (May 2013), pp. 240–252. DOI: 10.1016/j.jcp.2013.01.039 (cit. on p. 123).

[152] Jok M. Tang et al. "Comparison of Two-Level Preconditioners Derived from Deflation, Domain Decomposition and Multigrid Methods". In: *Journal of Scientific Computing* 39.3 (Jan. 2009), pp. 340–370. DOI: 10.1007/s10915-009-9272-6 (cit. on pp. 49, 115).

[153] Andrea Toselli and Olof B. Widlund. *Domain decomposition methods: algorithms and theory*. Vol. 34. Springer, 2005 (cit. on pp. 5, 10, 36, 48, 49, 120).

[154] Thanh Tran. "Overlapping additive Schwarz preconditioners for boundary element methods". In: *Journal of Integral Equations and Applications* 12.2 (2000), pp. 177–206 (cit. on p. 86).

[155] Thanh Tran and Ersnt P. Stephan. "Additive Schwarz algorithms for the p version of the Galerkin boundary element method". In: *Numerische Mathematik* 85.3 (2000), pp. 433–468 (cit. on p. 86).

[156] Thanh Tran and Ersnt P. Stephan. "Additive Schwarz methods for the h-version boundary element method". In: *Applicable Analysis* 60.1-2 (1996), pp. 63–84. DOI: 10.1080/00036819608840418. URL: https://doi.org/10.1080/00036819608840418 (cit. on pp. 4, 10, 33, 86).

[157] Thanh Tran and Ersnt P. Stephan. "An overlapping additive Schwarz preconditioner for boundary element approximations to the Laplace screen and Lamé crack problems". In: *Journal of Numerical Mathematics jnma* 12.4 (2004), pp. 311–330 (cit. on pp. 86, 113).

[158] Todd Veldhuizen. "Techniques for scientific C++". In: *Computer science technical report* 542 (2000), p. 60 (cit. on p. 142).

[159] Tobias Von Petersdorff. "Boundary integral equations for mixed Dirichlet, Neumann and transmission problems". In: *Mathematical methods in the applied sciences* 11.2 (1989), pp. 185–213. DOI: 10.1002/mma.1670110203 (cit. on pp. 57, 58, 61).

[160] Douglas B. West. *Introduction to Graph Theory (2nd Edition)*. Pearson, 2000. ISBN: 0-13-014400-2. URL: https://www.amazon.com/Introduction-Graph-Theory-Douglas-West/dp/0130144002?SubscriptionId=AKIAIOBINVZYXZQZ2U3A&tag=chimbori05-20&linkCode=xm2&camp=2025&creative=165953&creativeASIN=0130144002 (cit. on p. 44).

[161] Olof B. Widlund. "The Development of Coarse Spaces for Domain Decomposition Algorithms". In: *Domain Decomposition Methods in Science and Engineering XVIII*. Ed. by Michel Bercovier et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 241–248. ISBN: 978-3-642-02677-5. DOI: 10.1007/978-3-642-02677-5_26. URL: http://dx.doi.org/10.1007/978-3-642-02677-5_26 (cit. on pp. 36, 50).

[162] Olof Widlund and Maksymilian Dryja. *An additive variant of the Schwarz alternating method for the case of many subregions*. Tech. Rep 339. Department of Computer Science, Courant Institute, 1987 (cit. on p. 45).

[163] Te-Kao Wu and Leonard L. Tsai. “Scattering from arbitrarily-shaped lossy dielectric bodies of revolution”. In: *Radio Science* 12.5 (1977), pp. 709–718 (cit. on pp. 4, 9, 56, 61).

[164] Leonardo Zepeda-Núñez and Laurent Demanet. “The method of polarized traces for the 2D Helmholtz equation”. In: *Journal of Computational Physics* 308 (Mar. 2016), pp. 347–388. DOI: 10.1016/j.jcp.2015.11.040 (cit. on p. 123).

APPENDIX A

Computation of the Sobolev-Slobodeckij mass matrix

Outline of the current chapter

We detail here two methods to compute the Sobolev-Slobodeckij matrix $\mathbf{H}_p$ introduced in Equation (5.14). In any case, we need to compute a global integral as we said in Remark 5.26. This implies that the assembly of the finite element matrix will be quite different to usual BEM and FEM matrices. To simplify, we will focus on building the global matrix $\mathbf{H}$ defined by

$$(\mathbf{H})_{i,j} := \int_\Gamma \varphi_i \varphi_j \, \mathrm{d}\sigma(\mathbf{x}) \, \mathrm{d}\sigma(\mathbf{y}) + \int_{\Gamma\times\Gamma} \frac{(\varphi_i(\mathbf{x}) - \varphi_i(\mathbf{y}))(\varphi_j(\mathbf{x}) - \varphi_j(\mathbf{y}))}{|\mathbf{x}-\mathbf{y}|^{d+1}} \, \mathrm{d}\sigma(\mathbf{x}) \, \mathrm{d}\sigma(\mathbf{y}),$$

with $1 \leq i, j \leq N$ and using $\mathbb{P}_1$ Lagrange functions. Let us denote $\mathbf{E}_i \in \mathbb{R}^N$ such that

$$(\mathbf{E}_i)_j = \begin{cases} 1 & \text{if i=j} \\ 0 & \text{otherwise} \end{cases} \quad \text{and} \quad (\mathbf{M}_{i,j}) = \int_\Gamma \varphi_i(\mathbf{x}) \varphi_j(\mathbf{x}) \, \mathrm{d}\sigma(\mathbf{x}).$$

A.1 Looping over the elements

To compute efficiently $\mathbf{H}$, we can loop over the elements and compute local stiffness matrices, as usually done in FEM and BEM.

$$\begin{aligned}\mathbf{H} =&\mathbf{M} + \sum_{i=1}^{N}\sum_{j=1}^{N} \mathbf{E}_i\mathbf{E}_j^T \int_{\Gamma\times\Gamma} \frac{(\varphi_i(\mathbf{x})-\varphi_i(\mathbf{y}))(\varphi_j(\mathbf{x})-\varphi_j(\mathbf{y}))}{|\mathbf{x}-\mathbf{y}|^{d+1}}\,\mathrm{d}\sigma(\mathbf{x})\,\mathrm{d}\sigma(\mathbf{y})\\ =&\mathbf{M} + \sum_{i=1}^{N}\sum_{j=1}^{N} \mathbf{E}_i\mathbf{E}_j^T \sum_{T\in\mathcal{T}_h}\sum_{T'\in\mathcal{T}_h} \int_{T\times T'} \frac{(\varphi_i(\mathbf{x})-\varphi_i(\mathbf{y}))(\varphi_j(\mathbf{x})-\varphi_j(\mathbf{y}))}{|\mathbf{x}-\mathbf{y}|^{d+1}}\,\mathrm{d}\sigma(\mathbf{x})\,\mathrm{d}\sigma(\mathbf{y})\\ =&\mathbf{M} + \sum_{T\in\mathcal{T}_h}\sum_{T'\in\mathcal{T}_h} \underbrace{\int_{T\times T'} \sum_{i=1}^{N}\sum_{j=1}^{N} \mathbf{E}_i\mathbf{E}_j^T \frac{(\varphi_i(\mathbf{x})-\varphi_i(\mathbf{y}))(\varphi_j(\mathbf{x})-\varphi_j(\mathbf{y}))}{|\mathbf{x}-\mathbf{y}|^{d+1}}\,\mathrm{d}\sigma(\mathbf{x})\,\mathrm{d}\sigma(\mathbf{y})}_{K_{\mathrm{loc}}(T,T')}\end{aligned}$$

where $K_{\mathrm{loc}}(T,T')$ is the elementary contribution to $\mathbf{H}$ associated with the interaction between T and T', it is also called *local stiffness matrix*.

One can see that

$$(K_{\mathrm{loc}}(T,T'))_{i,j} \neq 0 \iff \mathrm{supp}(\varphi_i)\cap(T\cup T') \text{ and } \mathrm{supp}(\varphi_j)\cap(T\cup T'),$$

so that, taking $\mathbb{P}_1$ functions and a triangular mesh, $K_{\mathrm{loc}}(T,T')$ can be reduced to a $(2d-k)\times(2d-k)$ matrix with k the number of common nodes. Indeed, suppose that $\overline{T}\cap\overline{T'}=\emptyset$ and φ_i is associated with a node of T, then $(K_{\mathrm{loc}}(T,T'))_{i,j}$ is non-zero as soon as φ_j is associated with a node of T', similarly to BEM. But also if φ_j is associated with a node of T. In particular, if $i=j$, you see that $(K_{\mathrm{loc}}(T,T'))_{i,i}$ is a part of the global integral

$$(\mathbf{H})_{i,i} = \mathbf{M}_{i,i} + \mathbf{E}_i\mathbf{E}_i^T \int_{\Gamma\times\Gamma} \frac{(\varphi_i(\mathbf{x})-\varphi_i(\mathbf{y}))^2}{|\mathbf{x}-\mathbf{y}|^{d+1}}\,\mathrm{d}\sigma(\mathbf{x})\,\mathrm{d}\sigma(\mathbf{y}).$$

In other words, the computation of the global integrals is distributed in all the computations of the local stiffness matrices, so that the complexity of building $\mathbf{H}$ is $O(N_{\mathrm{el}}^2)$ at worst, with N_{el} the number of elements in the mesh $\mathcal{T}_h$. In our applications, we are computing $\mathbf{H}_p$ so that the complexity is related to the number of elements of the subdomain, that is why this approach is still acceptable in terms of computation time.

This method can be actually quite inconvenient when using compression. For example, in the case of hierarchical matrices, we need to be able to compute extra diagonal blocks that are compressed. But if in practice we compute this local stiffness matrices, then we will generate part of the diagonal at the same time.

A.2 Looping over the degrees of freedom

Another approach consists in looping over the degrees of freedom so that, for every $1\leq i,j\leq N$, we compute

$$(\mathbf{H})_{i,j} = \mathbf{M}_{i,j} + \mathbf{E}_i\mathbf{E}_j^T \int_{\Gamma\times\Gamma} \frac{(\varphi_i(\mathbf{x})-\varphi_i(\mathbf{y}))(\varphi_j(\mathbf{x})-\varphi_j(\mathbf{y}))}{|\mathbf{x}-\mathbf{y}|^{d+1}}\,\mathrm{d}\sigma(\mathbf{x})\,\mathrm{d}\sigma(\mathbf{y}),$$

and notice that it involves a global integral as soon as $\operatorname{supp}(\varphi_i) \cap \operatorname{supp}(\varphi_j) \neq 0$. Thus, the computation of such a coefficient costs $O(N_{\mathrm{el}})$ so that the overall computation will be roughly at worse $O(N(m_{\max} N_{\mathrm{el}} + N - m_{\min}))$, where $m_{\max}$ and $m_{\min}$ are respectively the maximum and the minimum number of degrees of freedom whose support intersect.

In this case, there is no issue with compression compared to looping over elements. But the complexity can be greater when $m_{\max}$ is large, which might occur especially in 3D.

A.3 Conclusion

Assembling $\mathbf{H}$ is far from trivial and quite different from usual BEM and FEM. We showed two different methods that can be useful in different cases. The first one is more adapted to dense matrices, while the second one is useful with compression. Ideally, one would like to use both, the first one for far interactions and the second one for close interactions. In practice, it also depends on how easy it can be to access the associated vertices of one element and vice versa.

Note that computing $\mathbf{H}$ is particularly useful when approximating the fractional Laplacian and apart from the point we made here, it is quite similar to BEM, see [16, Section 3.3] and [1, Section 4.1 and Appendix 1].

APPENDIX B

Discussion on the analysis of GenEO Single layer coarse space

Outline of the current chapter

In this appendix, we want to show why we were not able to fully analysis the GenEO single layer coarse space introduced in Section 5.3.4 using a local weakly singular operator defined as

$$\langle \mathrm{V}_p\, u_p, v_p \rangle_{H^{1/2}(\Gamma_p)\times \widetilde{H}^{-1/2}(\Gamma_p)} := \langle \mathrm{V}\, \mathrm{E}_{\Gamma_p}(u_p), \mathrm{E}_{\Gamma_p}(v_p) \rangle_{H^{1/2}(\Gamma)\times \widetilde{H}^{-1/2}(\Gamma)}$$

for every u_p, $v_p \in \widetilde{H}^{-1/2}(\Gamma_p)$. This local operator was used to define an equivalent $H^{1/2}$-norm following the GenEO Slobodeckij coarse space. The issue will also prevent us to fully analyse a coarse space for the weakly singular operator.

B.1 Analysis

Following Section 5.3.4, it remains to prove an inequality like Equation (5.6) that we recall here

$$C_{\mathrm{loc}} \sum_{p=1}^{n} (\mathbf{B}_p \mathbf{R}_p \mathbf{u}_h, \mathbf{R}_p \mathbf{u}_h) \leqslant a_{\mathrm{W}}(P_h \mathbf{u}_h, P_h \mathbf{u}_h),$$

where $\mathbf{B}_p$ is related to V_p. We proved in Lemma 5.23 the following inequality

$$\frac{C_{\mathrm{W}}^-}{k_1} \sum_{p=1}^{n} \|u|_{\Gamma_p}\|^2_{H^{1/2}(\Gamma_p)} \leqslant a_{\mathrm{W}}(u,u).$$

Suppose we have the following equivalence

$$C^-_{\text{ext},p}\|u\|^2_{\widetilde{H}^{-1/2}(\Gamma_p)} \leq \|\mathrm{E}_{\Gamma_p}(u)\|^2_{\widetilde{H}^{-1/2}(\partial\Omega)} \leq C^+_{\text{ext},p}\|u\|^2_{\widetilde{H}^{-1/2}(\Gamma_p)}.$$

Then, we have the following lemma

Lemma B.1. *Let $\mathbf{u}_h \in \mathbb{R}^N$, then we have*

$$\frac{C^-_{\mathrm{W}}C^-_{\mathrm{V}}}{k_1 C^-_{\text{ext},p}}\sum_{p=1}^{n}\langle \mathrm{V}_p^{-1}(u_h|_{\Gamma_p}), u_h|_{\Gamma_p}\rangle \leq a(u_h,u_h).$$

Proof. Let us denote $\psi_p = \mathrm{V}_p^{-1}\, u_h|_{\Gamma_p} \in \widetilde{H}^{-1/2}(\Gamma_p)$. Then, we have

$$\|\psi_p\|^2_{\widetilde{H}^{-1/2}(\Gamma_p)} \leq C_p\|\mathrm{E}_{\Gamma_p}(\psi_p)\|^2_{\widetilde{H}^{-1/2}(\Gamma)}.$$

And since the weakly singular operator is continuous and positive definite, we have

$$\begin{aligned}
C^-_{\mathrm{V}}\|\psi_p\|^2_{\widetilde{H}^{-1/2}(\Gamma_p)} &\leq C^-_{\text{ext},p}C^-_{\mathrm{V}}\|\mathrm{E}_{\Gamma_p}(\psi_p)\|^2_{\widetilde{H}^{-1/2}(\Gamma)}\\
&\leq C^-_{\text{ext},p}a_{\mathrm{V}}(\mathrm{E}_{\Gamma_p}(\psi_p),\mathrm{E}_{\Gamma_p}(\psi_p))\\
&\leq C^-_{\text{ext},p}\langle \mathrm{V}_p\,\psi_p,\psi_p\rangle_{H^{1/2}(\Gamma_p)\times\widetilde{H}^{-1/2}(\Gamma_p)}\\
&\leq C^-_{\text{ext},p}\|\mathrm{V}_p\,\psi_p\|_{H^{1/2}(\Gamma_p)}\|\psi_p\|_{\widetilde{H}^{-1/2}(\Gamma_p)}.
\end{aligned}$$

We deduce

$$\begin{aligned}
\langle \mathrm{V}_p^{-1}\, u_h|_{\Gamma_p}, u_h|_{\Gamma_p}\rangle_{\widetilde{H}^{-1/2}(\Gamma_p)\times H^{1/2}(\Gamma_p)} &= \langle \psi_p, \mathrm{V}_p\,\psi_p\rangle_{\widetilde{H}^{-1/2}(\Gamma_p)\times H^{1/2}(\Gamma_p)}\\
&\leq \|\psi_p\|_{\widetilde{H}^{-1/2}(\Gamma_p)}\|V_p\psi_p\|_{H^{1/2}(\Gamma_p)}\\
&\leq \frac{C^-_{\text{ext},p}}{C^-_{\mathrm{V}}}\|\mathrm{V}_p\,\psi_p\|^2_{H^{1/2}(\Gamma_p)} = \frac{C^-_{\text{ext},p}}{C^-_{\mathrm{V}}}\|u_h|_{\Gamma_p}\|^2_{H^{1/2}(\Gamma_p)}.
\end{aligned}$$

Using the last relation for each term of the sum in Lemma 5.23, we obtain the expected result. □

Notice that we need to be able to compute the left-hand side of the inequality obtained in Lemma B.1, because we need to compute a matrix $\mathbf{B}_p$. But we do not know analytically V_p^{-1} so that we cannot directly compute the left-hand term of the inequality from Lemma B.1. The next lemma shows that we can approximate the Galerkin approximation of the inverse of V_p by the inverse of the approximation of V_p. A similar difficulty arises with operator preconditioning (see [124, 146]). To overcome this, we prove a lemma similar to [124, Theorem 3.1.] and [146, Theorem 3.1.]:

Lemma B.2. *Let $\mathbf{u}_h \in \mathbb{R}^N$ and $u_h = P_h\mathbf{u}_h$, we have*

$$(\widetilde{\mathbf{C}}_p\mathbf{R}_p\mathbf{u}_h, \mathbf{R}_p\mathbf{u}_h) \leq (\mathbf{C}_p\mathbf{R}_p\mathbf{u}_h, \mathbf{R}_p\mathbf{u}_h) = \langle \mathrm{V}_p^{-1}(u_h|_{\Gamma_p}), u_h|_{\Gamma_p}\rangle_{\widetilde{H}^{-1/2}(\Gamma_p)\times H^{1/2}(\Gamma_p)}$$

where

$$\begin{aligned}
\mathbf{C}_p(i,j) &:= \langle \mathrm{V}_p^{-1}(\varphi_{\sigma_p(j)}|_{\Gamma_p}), \varphi_{\sigma_p(i)}|_{\Gamma_p}\rangle_{\widetilde{H}^{-1/2}(\Gamma_p)\times H^{1/2}(\Gamma_p)},\\
\mathbf{V}_p(i,j) &:= \langle \mathrm{V}_p(\varphi_{\sigma_p(j)}|_{\Gamma_p}), \varphi_{\sigma_p(i)}|_{\Gamma_p}\rangle_{H^{1/2}(\Gamma_p)\times \widetilde{H}^{-1/2}(\Gamma_p)},\\
\mathbf{M}_p(i,j) &:= \langle \varphi_{\sigma_p(j)}|_{\Gamma_p}, \varphi_{\sigma_p(i)}|_{\Gamma_p}\rangle_{\widetilde{H}^{-1/2}(\Gamma_p)\times H^{1/2}(\Gamma_p)},\\
\widetilde{\mathbf{C}}_p &:= \mathbf{M}_p\mathbf{V}_p^{-1}\mathbf{M}_p,
\end{aligned}$$

with $\varphi_{\sigma_p(i)}, \varphi_{\sigma_p(j)} \in \mathcal{V}_{h,p}$ *for all* $1 \le i,j \le N_p$

Proof. Let us denote $u_h|_{\Gamma_p} = \hat{u}_h^p = \sum_{j=1}^{N_p} u_{h,j}^p \varphi_{\sigma_p(i)}|_{\Gamma_p}$ with σ_p, the permutation from the local to the global numbering, for $1 \le p \le n$. Notice that $\hat{u}_h^p \in H^{1/2}(\Gamma_p)$ and let $t_p = \mathrm{V}_p^{-1}\hat{u}_h^p \in \widetilde{H}^{-1/2}(\Gamma_p)$ so that

$$(\mathbf{C}_p\mathbf{R}_p\mathbf{u}_h, \mathbf{R}_p\mathbf{u}_h) = \langle \mathrm{V}_p^{-1}(\hat{u}_h^p), \hat{u}_h^p\rangle_{\widetilde{H}^{-1/2}(\Gamma_p)\times H^{1/2}(\Gamma_p)} = \langle t_p, \mathrm{V}_p(t_p)\rangle_{\widetilde{H}^{-1/2}(\Gamma_p)\times H^{1/2}(\Gamma_p)}.$$

Then, we define $t_h^p = \sum_{i=1}^{N_p} t_{h,i}^p \varphi_{\sigma(i)}|_{\Gamma_p} \in H^{1/2}(\Gamma_p) \subset \widetilde{H}^{-1/2}(\Gamma_p)$ such that, for every $v_h^p = \sum_{i=1}^{N_p} v_{h,i}^p \varphi_{\sigma_p(i)}|_{\Gamma_p}$

$$\langle \mathrm{V}_p(t_h^p), v_h^p\rangle_{H^{1/2}(\Gamma_p)\times\widetilde{H}^{-1/2}(\Gamma_p)} = \langle \hat{u}_h^p, v_h^p\rangle_{H^{1/2}(\Gamma_p)\times\widetilde{H}^{-1/2}(\Gamma_p)}.$$

Remark that $t_p \neq t_h^p$ and we want to actually characterize the difference. Denoting $\mathbf{t}_h^p = (t_{h,i}^p)$, the previous relation can be written as follows

$$\mathbf{V}_p\mathbf{t}_h^p = \mathbf{M}_p\mathbf{R}_p\mathbf{u}_h.$$

Since $\mathbf{M}_p$ and $\mathbf{V}_p$ are symmetric, we have

$$\begin{aligned}
(\widetilde{\mathbf{C}}_p\mathbf{R}_p\mathbf{u}_h, \mathbf{R}_p\mathbf{u}_h) &= (\mathbf{M}_p\mathbf{V}_p^{-1}\mathbf{M}_p\mathbf{R}_p\mathbf{u}_h, \mathbf{R}_p\mathbf{u}_h)\\
&= (\mathbf{V}_p^{-1}\mathbf{M}_p\mathbf{R}_p\mathbf{u}_h, \mathbf{M}_p\mathbf{R}_p\mathbf{u}_h)\\
&= (\mathbf{V}_p^{-1}\mathbf{M}_p\mathbf{R}_p\mathbf{u}_h, \mathbf{V}_p\mathbf{t}_h^p)\\
&= (\mathbf{M}_p\mathbf{R}_p\mathbf{u}_h, \mathbf{t}_h^p)\\
&= \langle \hat{u}_h^p, t_h^p\rangle_{H^{1/2}(\Gamma_p)\times\widetilde{H}^{-1/2}(\Gamma_p)}.
\end{aligned}$$

Then, one can remark that

$$\langle \hat{u}_h^p, t_h^p\rangle_{H^{1/2}(\Gamma_p)\times\widetilde{H}^{-1/2}(\Gamma_p)} = \langle \mathrm{V}_p(t_p), t_h^p\rangle_{H^{1/2}(\Gamma_p)\times\widetilde{H}^{-1/2}(\Gamma_p)} = \langle \mathrm{V}_p(t_h^p), t_h^p\rangle_{H^{1/2}(\Gamma_p)\times\widetilde{H}^{-1/2}(\Gamma_p)},$$

so that, using the positivity and symmetry of V_p, we find that

$$\begin{aligned}
0 &\le \langle \mathrm{V}_p(t_p - t_h^p), (t_p - t_h^p)\rangle_{H^{1/2}(\Gamma_p)\times\widetilde{H}^{-1/2}(\Gamma_p)}\\
&\le \langle \mathrm{V}_p(t_p), t_p\rangle_{H^{1/2}(\Gamma_p)\times\widetilde{H}^{-1/2}(\Gamma_p)} - \langle \mathrm{V}_p(t_h^p), t_h^p\rangle_{H^{1/2}(\Gamma_p)\times\widetilde{H}^{-1/2}(\Gamma_p)}\\
&\le (\mathbf{C}_p\mathbf{R}_p\mathbf{u}_h, \mathbf{R}_p\mathbf{u}_h) - (\widetilde{\mathbf{C}}_p\mathbf{R}_p\mathbf{u}_h, \mathbf{R}_p\mathbf{u}_h).
\end{aligned}$$

□

Using the last two lemmas, we obtain a similar relation to Equation (5.6) with $C_{\mathrm{loc}} = \frac{C_{\mathrm{V}}^- C_{\mathrm{W}}^-}{k_1 C_{\mathrm{ext},p}^-}$

and $\mathbf{B}_p = \widetilde{\mathbf{C}}_p$ for every $1 \leq p \leq n$. Then, we can obtain the following theorem

Theorem B.3. *We have the following condition number estimate*

$$\kappa(\mathbf{P}_{\mathrm{ASM,sing}}\mathbf{A}_h) \leq 2\max\left(1, N_c\frac{5}{2}\frac{C_{\mathrm{W}}^{+}}{C_{\mathrm{W}}^{-}}\right)\left(2+\left(1+5N_c\frac{C^{+}}{C^{-}}\right)\frac{\tau k_1 C_{\mathrm{ext},p}^{-}}{C_{\mathrm{V}}^{-}C_{\mathrm{W}}^{-}}\right),$$

where $\mathbf{P}_{\mathrm{ASM,sing}}$ *is defined with the GenEO coarse space and the localization from Lemma B.1 where the local terms are approximated using Lemma B.2.*

Proof. Using Theorem 3.9 and its notations, we have to satisfy its two hypothesis.

(i) The first hypothesis is obtained using Lemma 5.12 with $c_R = 2\max(1, N_c\frac{5C_{\mathrm{W}}^{+}}{2C_{\mathrm{W}}^{-}})$.

(ii) The second hypothesis is obtained using Lemma 5.17 with $\mathbf{B}_p = \widetilde{\mathbf{C}}_p$ for every $1 \leq p \leq n$ and Lemmas B.1 and B.2 so that $c_T^{-1} = 2 + \left(1 + 5N_c\frac{C_{\mathrm{W}}^{+}k_1}{C_{\mathrm{W}}^{-}}\right)\frac{\tau k_1 C_{\mathrm{ext},p}^{-}}{C_{\mathrm{V}}^{-}C_{\mathrm{W}}^{-}}$.

□

Unfortunately, the previous theorem is not enough to get a rigorous bound on the condition number of $\mathbf{P}_{\mathrm{ASM,sing}}$. Indeed, all the constants are global constant that should not depend on the mesh size h or the number of subdomains n, except $C_{\mathrm{ext},p}^{-}$ that may depend on the size of Γ_p, and indirectly on the number of subdomains in a strong scaling test for example.

B.2 Issue with the operator of extension by zero

In this section, we formulate some remarks on the constant $C_{\mathrm{ext},p}^{-}$. We will drop the subscript p and just consider the case of an extension by zero for a function $u \in \widetilde{H}^{-1/2}(\Gamma)$, where $\Gamma \subsetneq \partial\Omega$ and $\Omega \subset \mathbb{R}^d$. So that the equivalence relation we would like is

$$C_{\mathrm{ext}}^{-}\|u\|_{\widetilde{H}^{-1/2}(\Gamma)} \leq \|\mathrm{E}_\Gamma(u)\|_{\widetilde{H}^{-1/2}(\partial\Omega)} \leq C_{\mathrm{ext}}^{+}\|u\|_{\widetilde{H}^{-1/2}(\Gamma)}, \tag{B.1}$$

where C^{-}ext and C^{+}ext are positive constants. And we need to understand to some extent how these constants depend on Γ. The next lemma relates the proof of the equivalence relation given in Equation (B.1) to the construction of a continuous extension operator $\mathcal{E}_\Gamma : H^{1/2}(\Gamma) \to H^{1/2}(\partial\Omega)$, i.e., $\mathcal{E}_\Gamma(v)|_\Gamma = v$ for every $v \in H^{1/2}(\Gamma)$.

Lemma B.4. *Let* $\Gamma \subsetneq \partial\Omega$ *with* $\Omega \subset \mathbb{R}^d$ *and suppose that there exists a continuous extension operator* $\mathcal{E}_\Gamma : H^{1/2}(\Gamma) \to H^{1/2}(\partial\Omega)$. *Then we have*

$$\frac{1}{C_{\mathcal{E}_\Gamma}}\|u\|_{\widetilde{H}^{-1/2}(\Gamma)} \leq \|\mathrm{E}_\Gamma(u)\|_{\widetilde{H}^{-1/2}(\partial\Omega)} \leq \|u\|_{\widetilde{H}^{-1/2}(\Gamma)},$$

where $C_{\mathcal{E}_\Gamma}$ *the operator norm of* $\mathcal{E}_\Gamma$.

Proof. Let $u \in \widetilde{H}^{-1/2}(\Gamma)$. By definition of $H^{1/2}(\Gamma)$, there exists $v = V|_\Gamma$ such that $V \in H^{1/2}(\partial\Omega)$ so that we can rewrite the definition of the $\widetilde{H}^{-1/2}$ norm as

$$\|u\|_{\widetilde{H}^{-1/2}(\Gamma)} = \sup_{V \in H^{1/2}(\partial\Omega)} \frac{\langle u, V|_\Gamma\rangle_{\widetilde{H}^{-1/2}(\Gamma)\times H^{1/2}(\Gamma)}}{\|V|_\Gamma\|_{H^{1/2}(\Gamma)}} = \sup_{V \in H^{1/2}(\partial\Omega)} \frac{\langle \mathrm{E}_\Gamma(u), V\rangle_{\widetilde{H}^{-1/2}(\partial\Omega)\times H^{1/2}(\partial\Omega)}}{\|V|_\Gamma\|_{H^{1/2}(\Gamma)}}.$$

Then, we have $\|V|_\Gamma\|_{H^{1/2}(\Gamma)} \leq \|V\|_{H^{1/2}(\partial\Omega)}$ using the expression of the norm, so that

$$\|u\|_{\widetilde{H}^{-1/2}(\Gamma)} \geq \sup_{V\in H^{1/2}(\partial\Omega)} \frac{\langle \mathrm{E}_\Gamma(u), V\rangle_{\widetilde{H}^{-1/2}(\partial\Omega)\times H^{1/2}(\partial\Omega)}}{\|V\|_{H^{1/2}(\partial\Omega)}} = \|\mathrm{E}_\Gamma(u)\|_{\widetilde{H}^{-1/2}(\partial\Omega)}.$$

To show the other inequality, remark that we have

$$\begin{aligned}\|\mathrm{E}_\Gamma(u)\|_{\widetilde{H}^{-1/2}(\partial\Omega)} &= \sup_{V\in H^{1/2}(\partial\Omega)} \frac{\langle \mathrm{E}_\Gamma(u), V\rangle_{\widetilde{H}^{-1/2}(\partial\Omega)\times H^{1/2}(\partial\Omega)}}{\|V\|_{H^{1/2}(\partial\Omega)}} \\ &\geq \sup_{v\in H^{1/2}(\Gamma)} \frac{\langle \mathrm{E}_\Gamma(u), \mathcal{E}_\Gamma(v)\rangle_{\widetilde{H}^{-1/2}(\partial\Omega)\times H^{1/2}(\partial\Omega)}}{\|\mathcal{E}_\Gamma(v)\|_{H^{1/2}(\partial\Omega)}}.\end{aligned}$$

Then, since $\mathcal{E}_\Gamma$ is continuous, we have $\|\mathcal{E}_\Gamma(v)\|_{H^{1/2}(\partial\Omega)} \leq C_{\mathcal{E}_\Gamma}\|v\|_{H^{1/2}(\Gamma)}$ for every $v \in H^{1/2}(\Gamma)$ and we can use the following relation

$$\langle \mathrm{E}_\Gamma(u), \mathcal{E}_\Gamma(v)\rangle_{\widetilde{H}^{-1/2}(\partial\Omega)\times H^{1/2}(\partial\Omega)} = \langle u, \mathcal{E}_\Gamma(v)|_\Gamma\rangle_{\widetilde{H}^{-1/2}(\Gamma)\times H^{1/2}(\Gamma)} = \langle u, v\rangle_{\widetilde{H}^{-1/2}(\Gamma)\times H^{1/2}(\Gamma)}$$

to obtain

$$\begin{aligned}\|\mathrm{E}_\Gamma(u)\|_{\widetilde{H}^{-1/2}(\partial\Omega)} &\geq \sup_{v\in H^{1/2}(\Gamma)} \frac{\langle \mathrm{E}_\Gamma(u), \mathcal{E}_\Gamma(v)\rangle_{\widetilde{H}^{-1/2}(\partial\Omega)\times H^{1/2}(\partial\Omega)}}{\|\mathcal{E}_\Gamma(v)\|_{H^{1/2}(\partial\Omega)}} \\ &\geq \sup_{v\in H^{1/2}(\Gamma)} \frac{\langle u, v\rangle_{\widetilde{H}^{-1/2}(\Gamma)\times H^{1/2}(\Gamma)}}{\|\mathcal{E}_\Gamma(v)\|_{H^{1/2}(\partial\Omega)}} \\ &\geq \frac{1}{C_{\mathcal{E}_\Gamma}} \sup_{v\in H^{1/2}(\Gamma)} \frac{\langle u, v\rangle_{\widetilde{H}^{-1/2}(\Gamma)\times H^{1/2}(\Gamma)}}{\|v\|_{H^{1/2}(\Gamma)}} = \frac{1}{C_{\mathcal{E}_\Gamma}}\|u\|_{\widetilde{H}^{-1/2}(\Gamma)}.\end{aligned}$$

□

This lemma is also valid if we define the $H^{1/2}$-norm by interpolation (see [123, 22, 14] for its definition) since the only part that depends explicitly on the norm is the inequality $\|V|_\Gamma\|_{H^{1/2}(\Gamma)} \leq \|V\|_{H^{1/2}(\partial\Omega)}$. And that can be proved easily by interpolation. And similarly, if we define the $H^{1/2}(\Gamma)$-norm as

$$\inf_{U\in H^{1/2}(\partial\Omega)} \left\{\|U\|_{H^{1/2}(\partial\Omega)} \,|\, U|_\Gamma = v\right\},$$

for $v \in H^{1/2}(\Gamma)$, then we have the same type of relation. In any case, we have $C^+_{\mathrm{ext}} = 1$ and $C^-_{\mathrm{ext}} = 1/C_{\mathcal{E}_\Gamma}$ where $\mathcal{E}_\Gamma$ is a continuous operator. Such continuous extension operator have been introduced (see [136, 144] for example), but we did not find any explicit dependence on Γ, and we were not able to prove that $C_{\mathcal{E}_\Gamma}$ is independent of the size of Γ.

B.3 Concluding remarks

We proved that the equivalence relation given in Equation (B.1) is related to the existence of a continuous operator, and that the constant in the equivalence relation depends on the continuity constant of this operator. Unfortunately, we have not found any proof that $C_{\mathcal{E}_\Gamma}$ is independent of the size of Γ, and we were not able to prove it ourself. But notice that an equality between

$\|u\|_{\widetilde{H}^{-1/2}(\Gamma)}$ and $\|\mathrm{E}_\Gamma(u)\|_{\widetilde{H}^{-1/2}(\partial\Omega)}$ is sometime used without rigorous proof. The same remark could be done between the $\widetilde{H}^{1/2}$-norms when they are defined by interpolation.

This issue is also what prevents us to fully analysis a DDM preconditioner for the weakly singular operator.

APPENDIX C

Example of code using Htool

We show a small practical example of code in Listing C.1. The lines between 5 and 41 correspond to the definition of the interface IMatrix. We define a simple operator that takes two arrays of 3D points `p1` and `p2`, and returns `1/ (p1[j]-p2[k])` for the interaction between `p1[j]` and `p2[k]`. It has also some auxiliary functions used afterward for comparing dense and hierarchical matrices. Then, the lines between 60 and 82 correspond to the generation of two randomized set of points corresponding to the geometry Figure 6.1. The actual part of the code where we build a hierarchical matrix and do a $\mathcal{H}$-matrix-vector product is located between the lines 84 and 88.

This example is part of the library in the example folder[1].

Remark C.1. *Htool defines some usual operations on the class vector from the C++ STL.*

Listing C.1 – Small example of code using Htool

```
1   #include <htool/htool.hpp>
2
3   using namespace std;
4   using namespace htool;
5
6   class MyMatrix: public IMatrix<double>{
7       const vector<R3>& p1;
8       const vector<R3>& p2;
9
10  public:
11      // Constructor
12      MyMatrix(const vector<R3>& p10,const vector<R3>& p20 ):
13          IMatrix(p10.size(),p20.size()),p1(p10),p2(p20) {}
14
15      // Virtual function to overload
16      double get_coef(const int& k, const int& j)const {
```

[1] https://github.com/PierreMarchand20/htool/tree/master/examples

```
17              return 1./(norm2(p1[j]-p2[k]));
18          }
19
20          // Matrix vector product
21          std::vector<double> operator*(std::vector<double> a){
22              std::vector<double> result(p1.size(),0);
23              for (int j=0;j<p1.size();j++){
24                  for (int k=0;k<p2.size();k++){
25                      result[j]+=this->get_coef(j,k)*a[k];
26                  }
27              }
28              return result;
29          }
30
31          // Frobenius norm
32          double norm(){
33              double norm = 0;
34              for (int j=0;j<p1.size();j++){
35                  for (int k=0;k<p2.size();k++){
36                      norm+=this->get_coef(j,k);
37                  }
38              }
39              return norm;
40          }
41      };
42
43
44      int main(int argc, char *argv[]) {
45
46          // Initialize the MPI environment
47          MPI_Init(&argc,&argv);
48
49          // Htool parameters
50          SetEpsilon(0.001);
51          SetEta(100);
52
53          // Data
54          srand (1);
55          int nr = 10000;
56          int nc = 5000;
57
58          // p1: points in a unit disk of the plane z=z1
59          double z1 = 1;
60          vector<R3> p1(nr);
61          for(int j=0; j<nr; j++){
62              double rho = ((double) rand() / (double)(RAND_MAX));
63              double theta = ((double) rand() / (double)(RAND_MAX));
64              p1[j][0] = sqrt(rho)*cos(2*M_PI*theta);
65              p1[j][1] = sqrt(rho)*sin(2*M_PI*theta);
```

```
66              p1[j][2] = z1;
67          }
68
69          // p2: points in a unit disk of the plane z=z2
70          double z2 = 2;
71          vector<R3> p2(nc);
72          for(int j=0; j<nc; j++){
73              double rho = ((double) rand() / (RAND_MAX));
74              double theta = ((double) rand() / (RAND_MAX));
75              p2[j][0] = sqrt(rho)*cos(2*M_PI*theta);
76              p2[j][1] = sqrt(rho)*sin(2*M_PI*theta);
77              p2[j][2] = z2;
78          }
79
80          // Hmatrix
81          MyMatrix A(p1,p2);
82          std::vector<double> x(nc,1),result(nr,0);
83          HMatrix<fullACA,double> HA(A,p1,p2);
84          result = HA*x;
85
86          // Output
87          HA.print_infos();
88          std::cout<< Frobenius_absolute_error(HA,A)/A.norm()<<"\n";
89          std::cout<< norm2(A*x-result)/norm2(A*x)<<std::endl;
90
91          // Finalize the MPI environment.
92          MPI_Finalize();
93      }
```

Schwarz methods and boundary integral equations

Abstract

The objective of this thesis is to use domain decomposition methods to develop new efficient methods for high performance computing and boundary integral equations. One can think of two approaches for domain decomposition. One can make a decomposition of the original domain where the solution is sought, a *volume decomposition*, and then formulate a boundary integral equation in each subdomain with some ways of coupling them. Or we could first set up a boundary integral equation and then apply a domain decomposition of the boundary, a *surface decomposition*. In the first approach, we show that the local variant of the multi-trace formulation, which is naturally well-adapted to parallelization, has optimal properties of convergence in the case of constant coefficients in the whole domain for a geometry without junction points. This property is similar to the convergence of the optimal Schwarz method, and we actually prove an equivalence between these two methods. Numerical tests are provided to illustrate the convergence property and show the potentialities and the limits of this approach when coefficients are piecewise constants instead of constants in the whole domain. In the second approach, we present how we use the framework of the fictitious space lemma and the approach of the GenEO (Generalized Eigenproblems in the Overlap) coarse space to define several two-level preconditioners for the hypersingular operator associated with any symmetric positive definite equation. Numerical experiments are provided to show scalability in terms of iterations using the conjugate gradient method and GMRes. To be able to use Schwarz preconditioners and the boundary element method, we also need to adapt a compression method to a distributed-memory parallel framework. This led us to implement Htool, a C++ library for hierarchical matrices parallelized using MPI and OpenMP.

Keywords: numerical analysis, domain decomposition methods, Schwarz methods, boundary integral equations, boundary element method, two-level preconditioners, hierarchical matrices

---

Méthodes de Schwarz et équations intégrales de frontière

Résumé

L'objectif de cette thèse est d'utiliser des méthodes de décomposition de domaine pour mettre au point de nouvelles méthodes pour le calcul haute performance et les équations intégrales de frontière. Dans le cas des équations intégrales de frontière, on peut penser à deux approches de décomposition de domaine. Nous pouvons faire une décomposition du domaine où la solution est recherchée, une *décomposition volumique*, puis formuler une équation intégrale de frontière dans chaque sous-domaine en les couplant. Ou nous pouvons d'abord établir une équation intégrale de frontière et ensuite appliquer une décomposition de domaine à la frontière, une *décomposition surfacique*. Dans la première approche, nous montrons que la variante locale de la formulation multi-trace, naturellement bien adaptée à la parallélisation, possède des propriétés de convergence optimales dans le cas de coefficients constants dans tout le domaine pour une géométrie sans points de jonction. Cette propriété est similaire à la convergence de la méthode optimale de Schwarz, et nous prouvons en réalité une équivalence entre ces deux méthodes. Des tests numériques sont fournis pour illustrer la propriété de convergence et montrer les potentialités et les limites de cette approche lorsque les coefficients sont constants par morceaux au lieu de constants dans l'ensemble du domaine. Dans la deuxième approche, nous présentons comment nous utilisons le cadre du lemme de l'espace fictif et l'approche de l'espace grossier GenEO (Generalized Eigenproblems in the Overlap) pour définir plusieurs préconditionneurs à deux niveaux pour l'opérateur hypersingulier associé à toute équation symétrique et définie positive. Des expériences numériques sont fournies pour montrer leur extensibilité en termes d'itérations avec la méthode du gradient conjugué et GMRes. Pour pouvoir utiliser les préconditionneurs de Schwarz et la méthode des éléments finis de frontière, nous devons également adapter une méthode de compression à un environnement parallèle à mémoire distribuée. Cela nous a conduit à implémenter une bibliothèque C++ pour les matrices hiérarchiques parallélisée en utilisant MPI et OpenMP.

Mots clés : analyse numérique, méthodes de décomposition de domaine, méthodes de Schwarz, équations intégrales de frontière, méthode des éléments finis de frontière, préconditionneurs à deux niveaux, matrices hiérarchiques

---

Laboratoire Jacques-Louis Lions
4 place Jussieu – 75005 Paris – France