Development of an algebraic multigrid solver for the indefinite Helmholtz equation

Clément Richefort

THÈSE PRÉSENTÉE

POUR OBTENIR LE GRADE DE

DOCTEUR
DE L'UNIVERSITÉ DE BORDEAUX

ECOLE DOCTORALE MATHEMATIQUES ET INFORMATIQUE

Informatique

Par **Clément RICHEFORT**

Développement d'un solveur multigrille algébrique pour l'équation de Helmholtz indéfinie

Sous la direction de : **Pierre RAMET**

Soutenue le 27 Novembre 2024

Membres du jury :

M. Pierre RAMET	Professeur	Université de Bordeaux	Directeur
M. Edmond CHOW	Professeur	Georgia Institute of Technology	Rapporteur
Mme. Stéphanie CHAILLAT	Directrice de Recherche	CNRS	Rapportrice
M. Ulrich RÜDE	Professeur	FAU Erlangen-Nürnberg	Examinateur
Mme. Vandana DWARKA	Chargée de Recherche	TU Delft	Examinatrice
M. Matthieu LECOUVEZ	Docteur	CEA	Invité
M. Robert D. FALGOUT	Docteur	Lawrence Livermore National Laboratory	Invité

Acknowledgements

Ces trois dernières années furent autant l'occasion d'approfondir mes connaissances en algèbre linéaire numérique que de grandir sur bien d'autres aspects! Ce manuscrit resterait incomplet si n'y étaient mentionnés quelques-uns parmi ceux qui peupleront à tout jamais mes souvenirs de thèse.

Merci à mon directeur de thèse, Pierre Ramet, dont l'accompagnement et l'ensemble des conseils prodigués furent irréprochables. Il y a plus de quatre ans maintenant, je t'envoyai cet interminable email plein de questions comme je sais si bien les écrire. Tu répondis point par point, et m'accordas de ton temps pour discuter recherche alors que l'on ne se connaissait pas encore. Je ne suis pas certain que tous en eussent fait autant. Pour sûr, mon aventure au CEA commença grâce au soin que tu as pris à me répondre, et je t'en serai toujours reconnaissant. Je te remercie aussi pour ta bienveillance et la facilité avec laquelle il est possible de te parler. Merci à Matthieu Lecouvez de m'avoir ouvert les portes du CEA. J'ai toujours l'email en sauvegarde, où, enfin, tu m'annonças que la thèse m'était attribuée. J'ai dû le lire une vingtaine de fois déjà. La finesse de tes observations m'impressionnera toujours et mes travaux ont beaucoup profité de ton esprit résolument scientifique. Comme je te l'ai dit, ce fut un réel plaisir de travailler avec toi et j'espère en avoir encore l'occasion. Voilà que ce souhait demeure à tout jamais gravé dans mon manuscrit! Je remercie mon troisième encadrant de thèse Rob Falgout, dont la contribution à ces travaux de recherche est inestimable. Je te sais bien entouré pour traduire le français, alors sache que te visiter en Californie fut une de ces grâces desquelles je resterai éternellement reconnaissant. Ton ardeur à la tâche et la pertinence de tes analyses m'ont beaucoup apporté. Plus que tout le reste, c'est ton sourire devant le tableau blanc que je m'appliquerai à imiter dans toute de ma carrière. La recherche t'amuse, et c'est précisément ce qui fait ton excellence. Rob, merci pour tout.

Je remercie mes deux rapporteurs Stéphanie Chaillat et Edmond Chow pour leurs relectures du manuscrit et leurs présences à la soutenance. Stéphanie, merci pour l'exhaustivité de tes commentaires et pour le gentil message qui leur était associé, ça m'aura mis dans d'excellentes dispositions pour la soutenance. Quant à toi Edmond, par deux fois ta curiosité scientifique nous poussa à discuter ensemble. La première fut à Copper, la seconde dans le bureau de Pierre. C'est toujours épatant de voir d'excellents chercheurs démontrer un réel intérêt pour le travail des étudiants. J'aimerais également remercier mes deux examinateurs Vandana Dwarka et Ulrich Rüde pour la pertinence de leurs questions. Je suis sûr que l'on aura l'occasion d'échanger à nouveau, et ce sera avec très grand plaisir. La passion se

transmet, et si je m'épanouis autant à gratter des formules d'algèbre sur le papier, c'est aussi parce que Serge Petiton fut mon professeur. Un grand merci, j'espère pouvoir transmettre mon goût pour la recherche avec autant de talent que vous.

Merci au CEA de m'avoir ouvert ses portes. En particulier, merci à Geneviève Maze-Merceur qui, avec Matthieu, m'avez donné la chance de travailler sur ce sujet de thèse. Merci à Chantal, Olivier, Agnès, Murielle, Pierre, Frédéric, Justine, Amandine, Emmanuele, Ilyès, Bruno, Sébastien, et Matthias, parce que vos salutations quotidiennes ont fait du labo un très chaleureux environnement de travail. Je remercie également Alexis Touzalin, illustration Larousse du terme "Bon gars" et vétéran du 205, pour sa remarquable gentillesse. Je peine encore à comprendre ton sujet de thèse, mais je sais par avance que tu la clôtureras avec brio. Je suis également heureux d'avoir pu partager mon bureau avec Roxanne Delville-Atchekzai. Puisque tu as déjà reçu le titre de la meilleure présentation de thèse en 180 secondes, je sais que tu vas tout déchirer pour ta soutenance! Je garderai aussi un excellent souvenir de tous les doctorants, et remercie l'INRIA pour son accueil.

Lorsque l'air sera plus humide, et que d'épais nuages couvriront le ciel, la nostalgie me renverra à mes deux étés passés au Lawrence Livermore National Laboratory. Merci de m'avoir accueilli, et merci à mes chefs de service de m'avoir laissé partir. Pensée pour mon co-bureau du LLNL et dorénavant ami Taoli Shen, avec qui j'aurais la joie de repartir aux sommets de la Sierra Nevada ou aux confins des Canyonlands. Merci à Sarah Osborn pour les s'mores, les BBQs, la Subaru et tout l'équipement de camping. Plus globalement, merci pour ton accueil, ton sourire et ta générosité! Enfin, merci à Claire Henze de m'avoir invité plusieurs fois à dîner sous son toit. Claire, tu as une superbe famille, et je te remercie pour ton attachement singulier à la France.

Mon séjour bordelais fut l'occasion de bien des rencontres. Raj', mon foie se souviendra toujours de toi, non pas tant pour nos apéros au Simone, mais surtout à cause d'un bel uppercut droit intercalé un mardi soir du mois de Mars. J'écourterai l'exposé ici, car j'ai déjà usé de mon droit de réponse le surlendemain en t'envoyant un joli crochet dans la narine. T'en as pas encore fini de mon amitié, car je compte bien continuer à te savater la tronche. Fabien, nous nous sommes rencontrés sur le tard, mais ce fut un réel plaisir de discuter avec toi. Avec Estelle, vous êtes de cet équipage que j'aimerais garder pour longtemps. Quant à mon ami et parrain Benoît, je suis heureux de t'avoir compté parmi mes collègues. Débouler en grandes pompes dans ton bureau pour discuter apologétique fut toujours très instructif. Certes, nos conversations prennent parfois un tournant plus gras, mais d'entre toutes je dois avouer que ce sont celles que je préfère. J'aurai grand plaisir à te revoir bientôt avec Manon. Je salue Yassine, fidèle ami d'école d'ingénieur dont les années à venir me rapprocheront encore.

Merci à ma famille d'avoir été là pour la soutenance comme dans tous les moments importants. On ne se fait jamais tout seul, et vous en êtes pour beaucoup. Papa, Maman, Lilou, et Harry, je vous aime énormément.

Enfin, au surplomb de toutes les grâces reçues au cours de ces trois années de thèse, celle du baptême fut la plus belle. Seigneur Jésus, tu es venu me chercher et m'a aimé le premier. Je relis toute ma vie à la lumière de notre rencontre : tu étais là depuis le début. Seigneur, merci. Je remercie également la paroisse de Saint Michel, et notamment le Père Henri Duc-Maugé pour son accompagnement spirituel. Les prêtres s'effacent toujours un peu devant les compliments, mais tout de même, je vous remercie pour le travail monumental que vous abattez. Merci à Patricia, Madeleine, et Sœur Michèle pour le suivi des catéchumènes. Merci au groupe des jeunes pros et aux prêtres d'être venus pour la soutenance. J'aimerais aussi te remercier chère Astrid, car notre rencontre fut déterminante. Souviens-toi toujours des lasagnes.

Résumé : La simulation numérique de phénomènes physiques complexes requiert généralement la résolution de systèmes d'équations linéaires. La méthode de résolution doit tirer profit des infrastructures modernes de calcul, et passer à l'échelle d'un parallélisme accru. En particulier, les méthodes multigrilles répondent à cette exigence de scalabilité, et permettent de résoudre une grande variété de problèmes dont le noyau a un aspect géométriquement lisse, et où les matrices de discrétisation sont symétriques définies positives.

Cette thèse vise à étendre les méthodes multigrilles à l'équation de Helmholtz, dont le noyau est oscillant et la matrice de discrétisation indéfinie. En particulier, le lisseur doit capturer les grandes valeurs propres indépendamment du signe et les opérateurs d'interpolation doivent ici propager une information oscillante et inconnue lors de la phase d'initialisation de la méthode. Enfin, la correction grossière perd ses propriétés de minimisation car la matrice indéfinie ne génère aucune norme. Par conséquent, une correction grossière alternative doit être développée pour garantir la contraction de l'erreur au fil des itérations. L'objectif est d'obtenir une méthode multi-niveaux convergente, et en un nombre d'itérations constant indépendant de la taille du problème. De nombreuses expériences numériques sont présentées tout au long de cette thèse.

Abstract : The numerical simulation of complex physical phenomena generally requires to solve systems of linear equations. The solver should benefit from modern computing machines and scale on highly parallel architectures. In particular, multigrid methods are scalable methods that enable the resolution of a wide range of problems where the discretization matrix is symmetric positive definite and the near-kernel space geometrically smooth.

In this thesis, our target is to extend multigrid methods to the oscillatory and indefinite Helmholtz equation. Each multigrid operator should be adapted to these challenging properties. In particular, the smoother should capture large eigenvalues independently of the sign. Moreover, the range of interpolation should now approximate the oscillatory near-kernel space that is unknown at the set-up phase of multigrid. Last, indefinite matrices do not generate a norm. As a consequence, the coarse correction has no minimization properties and can amplify the error. In this thesis, we present an alternative coarse correction that contracts the error properly. We target a multil-level method that converges in a constant number of iterations independently of the matrix size. Various numerical experiments are presented throughout this thesis.

Contents

Résumé étendu

Notre compréhension de la physique repose sur des modèles mathématiques mettant en relation diverses variables. Bien qu'une analyse théorique des équations du modèle puisse fournir beaucoup d'information, elles permettent aussi de simuler numériquement certains phénomènes physiques complexes, dangereux, ou trop chers à reproduire par l'expérience. Toutefois, ces équations appartiennent au monde continu. Par conséquent, elles doivent être discrétisées pour que la logique binaire du monde informatique puisse les appréhender. Cette étape si cruciale de la simulation numérique a demandé beaucoup d'efforts de recherche, et donna lieu à différentes techniques de discrétisation comme la méthode des éléments finis par exemple. Dans de nombreux cas, la simulation numérique d'un phénomène physique se ramène à la résolution d'un système d'équations linéaires désigné par la relation matricielle $A\boldsymbol{x} = \boldsymbol{b}$. Ici, le vecteur $\boldsymbol{x}$ de taille n contient les variables inconnues du problème. Conformément au modèle continu, la matrice A représente la transformation discrétisée pour passer de $\boldsymbol{x}$ à $\boldsymbol{b}$.

En théorie, inverser la matrice suffit à résoudre le système. Une telle opération serait hélas beaucoup trop coûteuse en pratique, mais de nombreuses alternatives existent. Un paramètre important à prendre en compte dans le choix de la méthode de résolution du système concerne la plateforme de calculs. L'augmentation progressive des puissances de calculs provenait d'abord des progrès réalisés sur la fréquence des processeurs, avant d'arriver à stagnation dans les années 2000. Dorénavant, l'approche consiste à ajouter de plus en plus de processeurs, en les couplant parfois avec des accélérateurs dédiés à certaines tâches bien spécifiées. Un code est dit "scalable" si la multiplication du nombre de processeurs par p divise son temps d'exécution par p. Ce critère de scalabilité est extrêmement important dans le choix de la méthode. Deux catégories de méthodes se distinguent à ce jour. La première catégorie est celle des méthodes directes, visant à factoriser la matrice A en deux matrices plus faciles à inverser. Ces méthodes sont robustes, mais difficilement scalables. La deuxième catégorie regroupe l'ensemble des méthodes itératives. Elles fonctionnent en raffinant, au fil des itérations, une approximation $\tilde{\boldsymbol{x}}$ de la solution $\boldsymbol{x}$ jusqu'à atteindre une différence suffisamment petite.

Dans ce contexte, le CEA étudie le comportement électromagnétique d'objets complexes en trois dimensions. Ici, c'est la discrétisation des équations de Maxwell qui permet la simulation d'un tel phénomène physique, et la résolution du système s'effectue à l'aide d'une méthode de décomposition de domaine. Cette méthode consiste à résoudre de plus petits systèmes associés à des sous-domaines du domaine

global. A chaque itération de la méthode, les approximations locales sont transmises à l'approximation de la solution globale, jusqu'à atteindre convergence. Bien que cette méthode permette la résolution des équations de Maxwell discrétisées, elle manque de scalabilité. Pour mettre la méthode à l'échelle des prochaines machines exaflopiques, une première idée consisterait à augmenter le nombre de sous-domaines. Cependant, la théorie indique qu'augmenter le nombre de sous-domaines décroît la vitesse de convergence de la méthode. La seconde option consisterait à augmenter la taille des sous-domaines, mais les problèmes locaux sont résolus à l'aide de méthodes directes qui manquent aussi de scalabilité. Pour ces raisons, le CEA recherche une méthode alternative afin d'anticiper l'arrivée des prochains supercalculateurs.

Une piste prometteuse est d'adapter les méthodes multigrilles pour la résolution des équations de Maxwell. Sous certaines conditions d'implémentation, ces méthodes peuvent être scalables. Le principe des méthodes multigrilles est d'accélérer le calcul de la solution en tirant profit d'une collection de problèmes grossiers de plus petites tailles. A chaque itération, le problème le plus petit est résolu à l'aide d'une méthode directe. L'approximation grossière est ensuite propagée du niveau le plus grossier jusqu'au plus fin par l'intermédiaire d'opérateurs d'interpolation. Au passage de chaque niveau, l'approximation est raffinée à l'aide d'un lisseur.

Initialement, un tel procédé visait la résolution de problèmes elliptiques dont les matrices de discrétisation sont symétriques définies positives, et dont l'espace associé aux plus petites valeurs propres a une forme géométriquement lisse. Pour ce qui suit, nous utiliserons le terme "near-kernel space" de la littérature scientifique pour désigner cet espace. Les différents opérateurs multigrilles tirent profit de telles caractéristiques. Premièrement, les méthodes itératives dites "stationnaires" comme les méthodes de Jacobi ou de Gauss-Seidel sont en général de très bons lisseurs dans le cas où les valeurs propres sont positives. Deuxièmement, l'aspect géométriquement lisse du near-kernel space facilite sa propagation par les opérateurs d'interpolation. Par exemple, une simple moyenne de points grossiers est parfois suffisante pour interpoler un point du niveau fin. Enfin, la matrice étant symétrique définie positive, elle génère une norme attribuant certaines propriétés de minimisation à l'opérateur de correction grossière.

Dans le cadre des équations de Maxwell, la matrice de discrétisation est indéfinie et son near-kernel space a un aspect oscillant. Cette thèse vise à adapter chacun des opérateurs multigrilles à ce contexte particulièrement exigeant en considérant une équation de Helmholtz dans un premier temps. En particulier, le lisseur doit traiter les grandes valeurs propres, qu'elles soient positives ou négatives. Les opérateurs d'interpolation, quant à eux, doivent propager efficacement le near-kernel space oscillant. Enfin, la matrice ne génère aucune norme puisqu'elle est indéfinie. Par conséquent, l'opérateur de correction grossière perd ses propriétés de minimisation et n'offre plus aucune garantie de convergence à la méthode multigrille.

La plupart des algorithmes développés au long de cette thèse fonctionnent sur la base d'heuristiques permettant de contrôler la complexité des différents opérateurs

en jeu. Cependant, nous cherchons d'abord à rendre la méthode convergente en un nombre d'itérations constant indépendamment de la taille du problème. De futurs travaux de recherche devront être entrepris pour la rendre plus efficace. Pour faciliter l'exposé de cette thèse, nous commençons par réintroduire quelques fondamentaux sur les méthodes multigrilles au Chapitre 2. Le Chapitre 3 dresse l'état de l'art des méthodes multigrilles pour Helmholtz, et détaille le problème de la correction grossière dans le cas indéfini en introduisant le concept de pollution. Ce concept connecte l'erreur d'interpolation avec la correction grossière. En particulier, nous concluons de cette théorie que la correction grossière sera toujours susceptible d'amplifier l'erreur dans le cas indéfini, quand bien même l'opérateur d'interpolation serait précis. Chacun des trois chapitres suivants s'attelle à un opérateur multigrille en particulier. Le Chapitre 4 présente un lisseur polynomial développé à partir des équations normales pour capturer les grandes valeurs propres indépendamment du signe. Un opérateur d'interpolation adapté à l'aspect oscillant du near-kernel space est introduit au Chapitre 5. Nous ouvrons ce chapitre en étudiant une approximation de l'opérateur idéal classique, puis le comparons à une approche différente basée sur une stratégie de minimisation de moindres carrés appliquée à une approximation du near-kernel space. Des illustrations du phénomène d'amplification de l'erreur opérée par la correction grossière sont présentées au début du chapitre 6, et justifient le développement d'une correction grossière alternative basée sur la minimisation d'une norme euclidienne. Enfin, le Chapitre 7 expose les résultats d'expériences numériques.

Sauf cas extrêmement indéfinis, les expériences numériques montrent que notre méthode à deux-niveaux converge là où la correction grossière traditionnelle amplifie l'erreur. Le spectre d'une matrice extrêmement indéfinie est plus susceptible de comporter des valeurs propres proches de zéro particulièrement sensibles à la pollution associée aux grandes valeurs propres. En l'état, nous voyons deux moyens pour accélérer la convergence de la méthode dans ce cas. Le premier vise à réduire la pollution en améliorant l'opérateur d'interpolation, tout en limitant le nombre d'éléments non nuls. Un second moyen consisterait à filtrer davantage les grands vecteurs propres dans l'espace de minimisation de la correction grossière alternative. Toutefois, de tels efforts devront être réitérés à mesure que les valeurs propres seront petites.

Enfin, les expériences numériques montrent qu'une des méthodes multi-niveaux résout l'équation de Helmholtz avec conditions de bords absorbantes en un nombre d'itérations constant indépendamment de la taille du problème, et jusqu'à six niveaux. Cependant, les opérateurs ne semblent pas encore assez creux pour aboutir à une implémentation scalable. Une piste de recherche consisterait à travailler sur des approximations plus creuses des matrices grossières afin de gagner en performance. Dans le cas symétrique défini positif, de telles approximations des matrices grossières pourraient rendre la méthode divergente si certaines conditions théoriques ne sont pas satisfaites. Dans notre cas, la correction grossière alternative fonctionne sur un principe de minimisation en norme euclidienne, et règle d'emblée ce problème.

Notations

To simplify the discussion in what follows, we use the term "small/large eigenvector" to mean an eigenvector with small/large eigenvalue in magnitude. We similarly say "positive/negative eigenvector" when referring to the eigenvalue sign. Additionally, capital italic Roman letters (A,E,P) denote matrices and bold lowercase letters denote vectors ($\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{r}, \boldsymbol{\alpha}$). Other lowercase letters denote scalar (σ, λ), while capital calligraphic letters denote sets and spaces ($\mathcal{C}$,$\mathcal{F}$,$\mathcal{K}$).

The following list dresses the main mathematical objects used in this manuscript.

Matrices

A	Matrix of the linear system $A\boldsymbol{x} = \boldsymbol{b}$ of size $n \times n$
V	Set of eigenvectors of A
V_c	Set of eigenvectors associated with the n_c smallest eigenvalues in magnitude
V_f	Set of eigenvectors associated with the n_f largest eigenvalues in magnitude
T	Set of test vectors of size $n \times \kappa$
P	Interpolation operator of size $n \times n_c$
P_*	Ideal interpolation operator of size $n \times n_c$
R^T	Coarse variable operator of size $n \times n_c$
S	Fine variable operator of size $n \times n_f$
$\hat{R}^T$	Least-squares coarse variable operator of size $n \times n_c$
$\hat{S}$	Least-squares fine variable operator of size $n \times n_f$
$\hat{P}$	Interpolation operator built on the least-squares variable operators of size $n \times n_c$
E	Error propagation matrix of the coarse correction
$p_d(A^2)$	Polynomial smoother of degree d built on normal equations
$q_{d+1}(A^2)$	Error propagation matrix of the polynomial smoother $p_d(A^2)$
K_f	Block of pollution of size $n_f \times n_c$
$\Pi\,(Q)$	l_2-orthogonal projection onto the range of Q
$\Pi_M\,(Q)$	M-orthogonal projection onto the range of Q
W_p	Alternative coarse correction's minimization space at the pth iteration of size $n \times p$
Z_p	Orthonormal operator associated with W_p of size $n \times p$
H_p	Hessenberg matrix associated with W_p of size $p \times p$

Vectors

$\boldsymbol{x}$	Solution to the linear system $A\boldsymbol{x} = \boldsymbol{b}$
$\boldsymbol{b}$	Right-hand side of the linear system $A\boldsymbol{x} = \boldsymbol{b}$
$\tilde{\boldsymbol{x}}$	Approximation of the solution to the linear system $A\boldsymbol{x} = \boldsymbol{b}$
$\boldsymbol{e}$	Error between the solution $\boldsymbol{x}$ and its approximation $\tilde{\boldsymbol{x}}$
$\boldsymbol{r}$	Residual or right-hand side of the linear system $A\boldsymbol{e} = \boldsymbol{r}$
$\boldsymbol{v}_i$	Eigenvector of A associated with the i^{th} largest eigenvalue λ_i in magnitude

Scalars and integers

α	Shift of the shifted Laplacian matrices (1.2) and (1.3), $\alpha = (kh)^2$
k	Wavenumber
h	Mesh size
κ	Number of test vectors in T
d	Degree of the polynomial smoother
a	Lower-bound of the interval $[a, b]$ for the selection of the Chebyshev nodes
b	Upper-bound of the interval $[a, b]$ for the selection of the Chebyshev nodes
λ_i	i^{th} largest eigenvalue in magnitude of A
n	Total number of points
n_c	Number of selected $\mathcal{C}$-points
n_f	Number of $\mathcal{F}$-points, i.e., $n_f = n - n_c$
τ	Parameter that controls the number of selected columns of $\hat{S}$
m	Parameter that controls the pattern of non-zero entries in the SPAI approach

---

Sets

$\mathcal{C}$	Group of interpolation $\mathcal{C}$-points
$\mathcal{F}$	Group of $\mathcal{F}$-points
$\mathcal{S}_i$	Group of strongly connected neighbors to the i^{th} point
$\mathcal{C}_i$	Group of $\mathcal{C}$-points strongly connected to the point i

---

List of Figures

List of Tables

Chapter 1

Introduction

Our understanding of physic relies on mathematical models that describe the evolution of a set of variables with respect to eventually numerous parameters. In particular, partial differential equations are the main among many mathematical tools on which are based these models, and intrinsically provide information by connecting the physical variables together. While a theoretical analysis of these special equations reveals a lot of information, they also enable the numerical simulations of physical phenomena that are often too complex, expensive and sometimes dangerous to experiment in practice.

Since partial differential equations belong to the abstract and continuous world of mathematics, they need to be discretized to fit into the finite and binary world of computers. Hence, a wide field of numerical analysis regards their discretization, and led to the development of different approaches such as finite element or finite difference methods for instance. In practice, simulating a discretized physical quantity often requires to solve a linear system of equations written $A\boldsymbol{x} = \boldsymbol{b}$ in matrix form, where the vector $\boldsymbol{x}$ of size n corresponds to the discretized unknown variables.

In theory, the system can be solved by inverting the matrix exactly. However, in practice, the inversion is computationally expensive and cheaper methods are preferred instead. Direct methods are the most robusts as they compute a convenient factorization of the matrix to enable a practical resolution of the system. Their downside is that they are difficult to scale on modern supercomputers and follow an $\mathcal{O}(n^3)$ complexity in general. Alternatively, iterative methods follow an $\mathcal{O}(n^2)$ complexity as they mostly rely on matrix vector products. However, their convergence is problem-dependent and can be very slow if the matrix is ill-conditioned for instance. For this reason, an important field of research in numerical linear algebra aims at designing cheap preconditioners that decrease the condition number of the matrix to accelerate the convergence.

The underlying motivation of this thesis is to simulate the electromagnetic behavior of three-dimensional complex objects on the most recent exascale machines. To do so, the French Alternative Energies and Atomic Energy Commission (CEA) already developed a software for solving the Maxwell's equations by combining a finite element method with an integral equation. The system arising from the finite ele-

ment discretization is currently solved by way of a domain decomposition method. This approach works iteratively by solving the sub-systems associated with the sub-domains that cover the entire discretization domain. Certain limitations of this method appear as the computing resources are growing. On the first hand, increasing the number of sub-domains deteriorates the convergence of the method. On the other hand, increasing the size of the sub-domains performs badly as direct solvers lack scalability on exascale machines. Through this thesis, the CEA wants to investigate multigrid as an alternative iterative method to domain decomposition.

The basic principle of multigrid is to use a collection of coarser problems that accelerates the convergence to the solution. A direct method is applied to the coarsest matrix, and the coarsest solution is interpolated up to the finest level. On each level, the approximation is refined with what we call a "smoother". Multigrid methods are known to be scalable and follow an $\mathcal{O}(n)$ complexity for sparse matrices and a good implementation. Multigrid methods are among the best solvers for a broad class of elliptic problems. However, they are not designed for indefinite matrices and extending them to oscillatory problems such as electromagnetism or acoustic is still an open question. Thereby, the final goal of this research program is to solve Maxwell's equations with multigrid, but the first step addressed in this thesis is to design a multigrid solver for the Helmholtz equation, as this problem is also indefinite and characterized by an oscillatory near-kernel space.

In fact, solving the indefinite Helmholtz equation is notably challenging. Finding a suitable and scalable iterative method that converges in a constant number of iterations independent of the matrix size remains an open question. In particular for multigrid, three main issues arise when solving the indefinite Helmholtz equation. First, Helmholtz has negative eigenvalues that are amplified when using traditional smoothers such as Jacobi or Gauss-Seidel for instance. Secondly, eigenvectors associated with the smallest eigenvalues in magnitude are not geometrically smooth as in elliptic problems for which multigrid methods are known to be particularly efficient. For Helmholtz, these smallest eigenvectors are oscillatory. Therefore, finding a suitable interpolation operator that interpolates the oscillatory near-kernel space from the coarsest level to the finest one is not simple. Lastly, the indefinite nature of the discretization matrix breaks the very convenient equivalence between the Galerkin coarse correction and a minimization problem in A-norm. This issue makes the theory inapplicable because the coarse correction loses its minimization properties. As a consequence, nothing guarantees the robustness of the method in the indefinite case and numerical experiments actually show that the method may diverge.

Our goal is to design a method that works algebraically and does not rely on prior knowledge of the underlying PDE. This choice is motivated by the ambition to solve a broader class of indefinite problems in the future such as Maxwell's equations. With that aim, our contributions can be split into four parts. After detailing the state of the art of multigrid for Helmholtz, we introduce the concept of "pollution" to better explain why the classical coarse correction appears hopeless in the indefinite case. This concept will be particularly important in this manuscript. Our

next contribution regards the smoother. We show that a Chebyshev polynomial smoother built on normal equations damps the large negative and positive eigenvalues in magnitude. Moreover, it also guarantees to preserve the smallest magnitude eigenvalues, which is especially important for constructing the interpolation operator introduced in the chapter that follows. In our case, the range of interpolation should be able to approximate the unknown oscillatory set of small eigenvectors. Therefore, extracting an approximation of the near-kernel space from the initial matrix is necessary. Our polynomial smoother makes it possible because it damps the large eigenvectors without touching the smallest ones. With this approximation of the near-kernel space, a sparse approximation of the ideal interpolation operator is constructed from an initial tentative interpolation operator built from a least-squares minimization strategy. The third part focuses on the problem of the coarse correction. In particular, we will see that the coarse correction is not guaranteed to contract the error in the indefinite case, even though the interpolation operator has good approximation properties. For this reason, we introduce an alternative coarse correction based on a Euclidean norm minimization. Lastly, numerical experiments are presented to challenge our method based on different parameters.

The main underlying motivation in our research is to find a scalable method for solving indefinite problems such as Maxwell or Helmholtz equations. As the main trouble for solving such a difficult class of problems with multigrid regards the convergence, this thesis targets an algebraic multigrid method that can solve Helmholtz in a constant number of iterations and independently of the problem size, but do not answer the question of scalability yet. While our alternative method obviously considers the question of computational complexity, more work should be undertaken in the future to make it more practical.

As major difficulties of solving the Helmholtz equation arise from its indefinite nature, we first focus on the two-dimensional shifted Laplacian (SL2D) matrix whose continuous problem is defined as follows

$$\text{(SL2D CP)} \quad \Leftrightarrow \quad \begin{cases} -\Delta \boldsymbol{u} - k^2 \boldsymbol{u} &= \boldsymbol{f} \quad \text{in } \Omega = [0,1] \times [0,1] \\ u &= 0 \quad \text{on } \partial\Omega \end{cases} . \tag{1.1}$$

Define $\alpha = (kh)^2$, and n_{grid} the grid size. The continuous shifted Laplacian problem (1.1) is discretized by way of a second order finite different scheme, such that the 5-point stencil of the resulting discretization matrix A is defined by

$$A \sim \frac{1}{h^2} \begin{bmatrix} & -1 & \\ -1 & 4-\alpha & -1 \\ & -1 & \end{bmatrix} \quad \text{with} \quad h := \frac{1}{n_{\text{grid}} + 1}. \tag{1.2}$$

In two dimensions, the size of the resulting discretization matrix is $n = n_{\text{grid}}^2$. Once n is fixed, the only varying parameter on which the discretization matrix A relies is the shift α. In fact, α controls the indefiniteness of the matrix. The problem gets more indefinite (i.e., the proportion of positive and negative eigenvalues reaches an equilibrium) as the shift α gets closer to 4 (the diagonal of A in (1.2) gets closer to zero). Moreover, higher shift leads to a more oscillatory problem. Extreme shifts lead to poor discretization but are interesting to make the problem

as hard as possible for the solver development. Throughout this manuscript, we always write the shift α as the discretization coefficient kh squared. For instance, $\alpha = 0.625^2$ corresponds to a problem discretized with 10 points per wavelength ω ($h = \omega/10 = 2\pi/10k \Leftrightarrow kh = 0.625$). The discretization matrix (1.2) has convenient structural properties that multigrid can benefit from. To challenge our method, we also work with the 9-point discretization matrix

$$A \sim \frac{1}{h^2} \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8-\alpha & -1 \\ -1 & -1 & -1 \end{bmatrix}. \tag{1.3}$$

Both discretization matrices (1.2) and (1.3) are Hermitian indefinites. This property is assumed in the following theoretical developments as it enables a convenient discussion based on eigenvalues and eigenvectors. Certain boundary conditions lead to non-hermitian indefinite matrices and therefore require to work with singular vectors for exactness. Since the difficulty of non-hermitian Helmholtz matrices is mostly due to their indefinite nature as well, we still rely on the eigenvectors/eigenvalues terminology for ease of discussion, and do not resort to singular vectors as we should for a rigorous development that consider all kind of matrices. We also note that matrices resulting from both stencils (1.2) and (1.3) are singular if α equals an eigenvalue of the Laplacian matrix. As working on the shifted Laplacian model problem (1.1) is only aimed at challenging our multigrid method with respect to the degree of indefiniteness of the problem, we always assume these matrices to be non-singular by setting the shift properly.

In their respective chapters, all of the three core components of our alternative multigrid method that are the smoother, the interpolation operator, and the coarse correction, are stressed in the two-level case and for different shifts α. The last Chapter 7 also addresses the multilevel case when replacing the Dirichlet boundary conditions by absorbing boundary conditions in (1.1).

The Chapter 2 introduces few fundamental ideas on iterative and multigrid methods that we use throughout this manuscript. The Chapter 3 presents a survey of past attempts of multigrid methods for Helmholtz. We also introduce our concept of "pollution" within Chapter 3 to better explain why the traditional coarse correction appears hopeless for indefinite problems. The next Chapter 4 addresses the design of an appropriate smoother for Helmholtz based on Chebyshev polynomials. Then, the design of good interpolation rules that approximate the oscillatory near-kernel space of the Helmholtz equation is presented in Chapter 5, and the alternative coarse correction aimed at contracting the error in the indefinite case is introduced in Chapter 6. Numerical experiments with extension to the multilevel case are discussed in Chapter 7. We end this manuscript with a conclusion and perspectives for future research in Chapter 8.

Chapter 2

Multigrid fundamentals

Before dipping into algebraic multigrid methods for Helmholtz, let us recall few fundamental concepts that will help the discussion throughout this manuscript. A system of linear equations can be solved directly by a convenient factorization of the initial matrix A, or iteratively by refining an approximation of the solution starting from an initial guess $\boldsymbol{x}^{(0)}$. In direct methods [52, 23], the solution is computed without iteration, and the precision only depends on the numerical round-off error. Despite their robustness, they lack scalability on modern high-performance computing architectures, and their total number of floating operations follows in general a $\mathcal{O}(n^3)$ complexity. Iterative methods [25, 66, 71] scale better with most of them having $\mathcal{O}(n^2)$ complexity. While the convergence of an iterative method is problem-dependent, they allow a trade-off between precision and computing complexity.

To better highlight the strength of multigrid and also because most classical iterative methods such as Jacobi, Gauss-Seidel or polynomial methods are still used in most sophisticated multigrid algorithms today, we reintroduce some of the most fundamental concepts of iterative methods.

2.1 Basics of iterative methods

Let $\boldsymbol{e}^{(k)}$ be the error between the solution $\boldsymbol{x}$ and its approximation $\boldsymbol{x}^{(k)}$ at the kth iteration. Also, let $\boldsymbol{r}^{(k)}$ be the residual, such that

$$\boldsymbol{e}^{(k)} = \boldsymbol{x} - \boldsymbol{x}^{(k)} \quad , \quad \boldsymbol{r}^{(k)} = A\boldsymbol{e}^{(k)} = \boldsymbol{b} - A\boldsymbol{x}^{(k)}. \tag{2.1}$$

Let M^{-1} be a practical approximation of the inverse of A. At every iteration, the new approximation $\boldsymbol{x}^{(k+1)}$ is computed by multiplying the residual $\boldsymbol{r}^{(k)}$ as follows

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + M^{-1}\boldsymbol{r}^{(k)}. \tag{2.2}$$

Thereafter, the residual is updated following (2.1). In addition, let E_M be the error propagation matrix, such that

$$E_M := I - M^{-1}A. \tag{2.3}$$

This operator will be particularly useful in the next chapters to study the different operators that compose multigrid. In certain cases, it is possible to derive from E_M

the damping factors associated with the eigenvectors of A. These damping factors are crucial in this manuscript. Substituting (2.2) in (2.1) leads to the recurrence relation

$$\begin{aligned}\boldsymbol{e}^{(k)} &= \boldsymbol{x} - \boldsymbol{x}^{(k)} = \boldsymbol{x} - \left(\boldsymbol{x}^{(k-1)} + M^{-1}\boldsymbol{r}^{(k-1)}\right) = \boldsymbol{x} - \boldsymbol{x}^{(k-1)} - M^{-1}A\boldsymbol{e}^{(k-1)} \\ &= \boldsymbol{e}^{(k-1)} - M^{-1}A\boldsymbol{e}^{(k-1)} = \left(I - M^{-1}A\right)\boldsymbol{e}^{(k-1)} = E_M\boldsymbol{e}^{(k-1)} \end{aligned} \tag{2.4}$$

It follows that the recurrence relation (2.4) corresponds to the geometric series

$$\boldsymbol{e}^{(k)} = (E_M)^k\boldsymbol{e}^{(0)}, \tag{2.5}$$

where the matrix E_M is the quotient. The norm of the remaining error at the kth iteration is bounded as follows

$$||\boldsymbol{e}^{(k)}||_2 = ||(E_M)^k\boldsymbol{e}^{(0)}||_2 \leq \rho(E_M)^k||\boldsymbol{e}^{(0)}||_2, \tag{2.6}$$

where $\rho(E_M)$ is the spectral radius of the error propagation matrix E_M. Because any geometric series converges if its quotient is strictly lower than one, then the iterative method converges if $\rho(E_M) < 1$. Moreover, (2.6) shows that a lower spectral radius leads to a faster convergence. Let $\epsilon = 10^{-q}$ be the relative tolerance, where q is a given number of digits. The iterative method satisfies the desired error tolerance when

$$\frac{||\boldsymbol{e}^{(k)}||_2}{||\boldsymbol{e}^{(0)}||_2} = \frac{||\left(E_M\right)^k\boldsymbol{e}^{(0)}||_2}{||\boldsymbol{e}^{(0)}||_2} \leq \epsilon = 10^{-q}. \tag{2.7}$$

From (2.5), this condition is satisfied if

$$\rho(E_M)^k \leq 10^{-q} \;\Leftrightarrow\; k \geq -\frac{q}{\log_{10}\rho(E_M)}. \tag{2.8}$$

The quantity $-\log_{10}\rho(E_M)$, called the *asymptotic convergence rate*, links the average number of iterations required with the number of exact digits expected.

2.1.1 Stationary methods

Stationary iterative methods work by way of a simple approximation M^{-1} of the inverse matrix A^{-1}, where M results from the splitting

$$A = M - N. \tag{2.9}$$

The matrix M should satisfy a trade-off between providing fast convergence and being practical to inverse. Let D be the diagonal of A, and L and U be the lower and the upper parts respectively, such that

$$A = D + L + U. \tag{2.10}$$

One of the most straightforward iterative methods is to approximate the inverse of A by the inverse of its diagonal D. This approach characterizes the so-called Jacobi method. Setting $M = D$ in (2.3) leads to the error propagation matrix

$$E_D = I - D^{-1}A. \tag{2.11}$$

Updating the approximation of the solution by applying D^{-1} is cheap, and convenient to implement in parallel. As shown by the following element-wise formulation, the approximation is updated by dividing each entry of the residual by its corresponding diagonal entry

$$\boldsymbol{x}_i^{(k+1)} = \boldsymbol{x}_i^{(k)} + \frac{\boldsymbol{r}_i^{(k)}}{a_{ii}}. \tag{2.12}$$

A variant of Jacobi called weighted-Jacobi adds a weight w to the diagonal inverse D^{-1}, such that the entry wise formulation and the error propagation matrix become

$$\boldsymbol{x}_i^{(k+1)} = \boldsymbol{x}_i^{(k)} + w\frac{\boldsymbol{r}_i^{(k)}}{a_{ii}} \quad , \quad E_{w^{-1}D} = I - wD^{-1}A. \tag{2.13}$$

Gauss-Seidel relaxation is another classical method that works by setting $M = D+L$. The shape of M being lower triangular, each element can be updated conveniently with a forward substitution. This feature leads to the following element-wise formulation

$$\boldsymbol{x}_i^{(k+1)} = \boldsymbol{x}_i^{(k)} + \frac{r_i^{(k)} - \sum_{j<i} a_{ij}x_j^{(k+1)}}{a_{ii}}. \tag{2.14}$$

In comparison with the Jacobi method where each entry are updated independently, Gauss-Seidel updates each new entry with the most recent approximation entries. Although this feature makes the method more sequential by nature, it may be possible to apply a pivoting strategy to improve the parallelism. Accordingly, the error propagator matrix of the Gauss-Seidel method is defined by

$$E_{D+L} = I - (D + L)^{-1}A. \tag{2.15}$$

Despite their popularity, these stationary methods are not further discussed in this Chapter as we barely mention them in the rest of the manuscript.

2.1.2 Polynomial methods

Chebyshev polynomials will play an important role in our multigrid method for Helmholtz. Let us now introduce this approach to anticipate the Chapter 4 dedicated to our polynomial smoother. The eigenvalues λ_i of the matrix A are the roots of its characteristic polynomial p_A, such that

$$p_A(\lambda) := \prod_{i=1}^{n}(\lambda - \lambda_i) = \sum_{i=0}^{n}\rho_i\lambda^i \quad \Rightarrow \quad \forall\lambda_i \in \sigma(A)\ ,\ p_A(\lambda_i) = 0. \tag{2.16}$$

The ρ_i in (2.16) simply correspond to the coefficients of the characteristic polynomial. The Cayley-Hamilton theorem states that replacing λ by A in the characteristic polynomial leads to the zero matrix. This property can be observed by the equality

$$p_A(A)\boldsymbol{v}_i = p_A(\lambda_i)\boldsymbol{v}_i = 0, \tag{2.17}$$

where $\boldsymbol{v}_i$ is the ith eigenvector of A associated with the eigenvalue λ_i. As a consequence, the Cayley-Hamilton theorem implies the following equivalence

$$p_A(A) = \sum_{i=0}^{n}\rho_i\lambda^i = 0 \quad \Leftrightarrow \quad A^{-1} = \sum_{i=1}^{n} -\frac{\rho_i}{\rho_0}A^{i-1}. \tag{2.18}$$

In other words, A^{-1} can be written as a polynomial of A. While generating such a polynomial of degree n is generally too expensive, polynomial iterative methods work by way of a truncation of the inverse polynomial formulation. Accordingly, a good polynomial p of degree $d \leq n$ satisfies

$$p_d(A) \approx A^{-1}. \tag{2.19}$$

As in many other stationary methods, this polynomial can be applied many times as an approximate inverse of the matrix until reaching a small relative residual norm, by setting $M^{-1} = p_d(A)$. In this case, the error propagation matrix of such a polynomial iterative method is defined by

$$q_{d+1}(A) := I - p_d(A)A, \tag{2.20}$$

where $p_d(A)$ is the inverse approximate used at each iteration. Note that the subscript of q_{d+1} is incremented due to the post-multiplication of p_d by A. A good inverse approximation should be a minimizer of the spectral radius of its associated error propagation matrix. From (2.17), finding such a polynomial is made through solving the minimization problem

$$p_d = \arg\min_{p \in \mathbb{P}_d} \rho\left(1 - p(A)A\right) = \arg\min_{p \in \mathbb{P}_d} \max_{\lambda \in \sigma(A)} \left|1 - p(\lambda)\lambda\right|. \tag{2.21}$$

with $\mathbb{P}_d$ the space containing all the real polynomials of degree d. In practice, the spectrum is unknown. Instead, d interpolation points x_i are chosen within a given interval $[a, b]$. If all the eigenvalues of the initial matrix A are contained in the interval (i.e., $\lambda_i \in [a, b]$, $i = 1, \ldots, n$), then the spectral radius is bounded as follows

$$\rho\left(q_{d+1}(A)\right) = \max_{\lambda_i \in \sigma(A)} \left|1 - p_d(\lambda)\lambda\right| \leq \max_{x \in [a,b]} \left|1 - p_d(x)x\right|. \tag{2.22}$$

To design a standalone solver based on a polynomial that approximates the inverse, a and b can be chosen as lower and upper estimates of the smallest and largest eigenvalues respectively. If A is positive, one algebraic manner of determining the bounds of the interval is to compute a few power iterations to estimate the largest eigenvalue for the upper-bound b, and set the lower-bound $a = 0$. Both smallest and largest eigenvalues can also be deduced from analytic information if accessible. Once the interval $[a, b]$ is established, the polynomial function $p_d(x)$ can be constructed to approximate the inverse function x^{-1} [53] by selecting $d + 1$ interpolation points x_i within the interval. In what follows, we call $q_{d+1}(x) := 1 - p_d(x)x$ the error propagation function, which is the homologous of the error propagation matrix $q_{d+1}(A)$ of (2.20) but with respect to the scalar x. Accordingly, the polynomial should satisfy the $d + 1$ following constraints

$$i = 0, \ldots, m \quad , \quad x_i \in [a, b] \quad , \quad p_d(x_i) = \frac{1}{x_i} \quad \Leftrightarrow \quad q_d(x_i) = 0. \tag{2.23}$$

These conditions are satisfied by using the Lagrangian formula

$$p_d(x) := \sum_{i=1}^{d+1} \frac{1}{x_i} \prod_{j=1\,,\,j \neq i}^{d+1} \frac{x - x_j}{x_i - x_j}. \tag{2.24}$$

Moreover, because the selected nodes x_i are the roots of q_d, we have

$$q_{d+1}(x) = \prod_{i=1}^{d+1} \frac{x - x_i}{-x_i}. \tag{2.25}$$

Note that the polynomial p_d is null at zero, such that

$$p_d(0) = 0 \quad \Leftrightarrow \quad q_{d+1}(0) = 1. \tag{2.26}$$

2.1.2.1 Chebyshev polynomials

The Cauchy's bound Theorem recalled in Section A.1 states that the error of the polynomial function in approximating x^{-1} is bounded by a function of interpolation points. In other words, we need to find the relevant set of $d+1$ interpolation points x_i that minimizes the error. Precisely, it appears that the roots of the first kind Chebyshev polynomial constitute the best set of interpolation points. Let T_{d+1} be the first kind Chebyshev polynomial of degree $d+1$ defined by

$$\forall t \in [-1, 1] \ , \ T_{d+1}(t) = \cos\left[(d+1)\arccos(t)\right]. \tag{2.27}$$

Also, define $\theta := \arccos(t)$. As developed in Appendix A.1, (2.27) can be expended to arrive at the three-term recurrence relation

$$T_0(t) = 1 \quad , \quad T_1(t) = t \quad , \quad T_{d+1}(t) = 2tT_d(t) - T_{d-1}(t). \tag{2.28}$$

The $d+1$ roots of T_{d+1} are given by

$$t_i = \cos\frac{(2i+1)\pi}{2(d+1)}. \tag{2.29}$$

One can demonstrate that the superior bound of the unitary polynomial $2^{-d}T_{d+1}$ is the smallest among all the unitary polynomials of degree $d+1$. In particular, one can show that

$$\frac{1}{2^d}\left\|T_{d+1}\right\|_\infty = \frac{1}{2^d} \leq \sup_{-1 \leq x \leq 1} \left\{ \left| \prod_{i=1}^{d+1} (x - x_i) \right| \right\} \tag{2.30}$$

Therefore, designing the polynomial p_d by selecting the interpolation points x_i as the roots of a Chebyshev polynomial minimizes the error of the Cauchy's bound. In that way, we need to remap the roots t_i of T_{d+1} into the domain $[a, b]$, which leads to

$$c_i := \frac{b+a}{2} + \frac{b-a}{2} \times t_i. \tag{2.31}$$

Selecting the nodes c_i of (2.31) as interpolation points provides the best polynomial function that approximates the inverse function x^{-1} within the interval $[a, b]$. It follows that they are the roots of q_{d+1} such that

$$p_d(c_i) = \frac{1}{c_i} \qquad \Leftrightarrow \qquad q_{d+1}(c_i) = 1 - p_d(c_i)c_i = 0. \tag{2.32}$$

Moreover, we have $q_{d+1}(0) = 1$. The error propagation function can finally be written as follows

$$q_{d+1}(x) = \prod_{i=1}^{d+1} \frac{(x - c_i)}{-c_i} = \prod_{i=1}^{d+1} \frac{(x - \frac{b+a}{2} - \frac{b-a}{2} t_i)}{\frac{b-a}{2} t_i - \frac{b+a}{2}}$$
$$= \left(\frac{b-a}{2}\right)^{d+1} \left(\frac{b-a}{2}\right)^{-d-1} \left[\prod_{i=1}^{d+1} \left(\frac{\frac{2x-b-a}{b-a} - t_i}{t_i - \frac{b+a}{b-a}}\right)\right]. \quad (2.33)$$

Lastly, and since the nodes t_i are the roots of T_{d+1}, the error propagation function q_{d+1} can finally be derived as the rescaled Chebyshev polynomial

$$q_{d+1}(x) = \frac{T_{d+1}\left(\frac{b+a-2x}{b-a}\right)}{T_{d+1}\left(\frac{b+a}{b-a}\right)}. \quad (2.34)$$

Because the roots are chosen to minimize the error between the polynomial function $p_d(x)$ and the inverse function x^{-1} within the interval $[a, b]$, the error propagation function (2.34) has the smallest superior bound within $[a, b]$. In other words, the Chebyshev polynomial provides the best damping factor for the eigenvalues of A that belongs to the interval $[a, b]$.

2.1.2.2 Convergence

Let us finish this introductory discussion on Chebyshev polynomial methods by a few words on the convergence. Assuming b bounds the largest eigenvalue such that $\lambda_{\max} \leq b$, the Chebyshev polynomial iterative method is convergent if the error propagation function $q_{d+1}(x)$ is bounded by one within $(0, b]$. As explained in [3], remark that $T_{d+1}(t)$ equals 1 for $t = 1$ and is strictly monotonically increasing for $t > 1$. Hence, the denominator of (2.34) is strictly greater than one because $\frac{b+a}{b-a} > 1$. Moreover, the numerator is upper bounded by one for $x \in [a, b]$ because $\frac{b+a-2x}{b-a} \in [-1, 1]$. As a consequence, $|q_{d+1}(x)| < 1$ on the interval $[a, b]$. Lastly, because q_{d+1} is strictly monotonically decreasing for $x \in [0, a]$ and equals one at zero, then

$$\forall x \in (0, b] \quad , \quad |q_{d+1}(x)| < 1 \quad \Rightarrow \quad \rho\left(q_{d+1}(A)\right) < 1. \quad (2.35)$$

2.1.3 Krylov methods

As stated in (2.18) of the previous section, the Cayley-Hamilton theorem implies that, for a given matrix A, there exists a unique polynomial of maximal degree n that is equal to the inverse matrix A^{-1} (i.e., $p_n(A) = A^{-1}$). Consequently, the solution to the linear system $A\boldsymbol{x} = \boldsymbol{b}$ satisfies

$$\boldsymbol{x} = A^{-1}\boldsymbol{b} = p(A)\boldsymbol{b} = \sum_{i=0}^{n-1} \rho_i A^i \boldsymbol{b}. \quad (2.36)$$

Let $\mathcal{K}_d$ be the Krylov subspace of size $d \leq n$ and Q_d be the $n \times d$ rectangular matrix whose columns compose an orthonormalized basis of $\mathcal{K}_d$, such that

$$\mathcal{K}_d := \text{span}\left\{\boldsymbol{b}, A\boldsymbol{b}, \ldots, A^{d-1}\boldsymbol{b}\right\} \quad \text{and} \quad \text{range}(Q_d) = \mathcal{K}_d. \quad (2.37)$$

Since a Krylov vector is the multiplication of a positive power of A with the right-hand side $\boldsymbol{b}$, (2.36) implies that the solution $\boldsymbol{x}$ belongs to the Krylov space $\mathcal{K}_n$ of size n. Neither constructing the polynomial $p_n(A)$ or seeking the solution within the Krylov space $\mathcal{K}_n$ of full size n helps in terms of computational complexity. Similarly to the previous polynomial approximation approach that is based on generating a polynomial of smaller degree d, the approximation is returned through generating a Krylov subspace of smaller size $d \leq n$. In practice, the Krylov space is augmented iteratively by orthonormalizing each new Krylov vector against all the previous ones. This step is generally called the Arnoldi procedure. The orthonormalization coefficients are stored in the $(d+1) \times d$ Hessenberg matrix denoted by $\bar{H}_d$, and the new Krylov vector is stored into the orthonormal set Q_d that also contains all the previous ones. The form of the Hessenberg matrix $\bar{H}_d$ is given in Appendix A.2 with the Arnoldi process summarized in Algorithm 7.

That said, the Arnoldi procedure gives the well-known relation

$$AQ_d = Q_{d+1}\bar{H}_d. \tag{2.38}$$

Given the approximation $\boldsymbol{x}^{(k)}$ and the residual $\boldsymbol{r}^{(k)} = b - A\boldsymbol{x}^{(k)}$ of the kth iteration, the method consists of minimizing the residual of the next iteration by solving

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \underset{\tilde{\boldsymbol{x}} \in \mathcal{K}_d}{\arg\min} \|\boldsymbol{r}^{(k)} - A\tilde{\boldsymbol{x}}\|_2. \tag{2.39}$$

To do so, GMRES first calls the Arnoldi procedure (see Algorithm 7) to construct the subspace $\mathcal{K}_d$. Secondly, the method minimizes the quantity (2.39) by taking benefit from both the orthonormality of Q_d and the convenient Hessenberg shape of $\bar{H}_d$ to compute

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + Q_d H_d^{-1} Q_d^T \boldsymbol{r}^{(k)}, \tag{2.40}$$

where H_d is obtained from $\bar{H}_d$ by deleting its last row. In practice, Given's rotations are used to turn H_d into an upper triangular matrix, and we apply a backward substitution to solve the system. This efficient approach that characterizes GMRES is further recalled in Appendix A.2.

2.2 Multigrid methods

Choosing one iterative method over another is problem-dependent. Stationary methods are easy to implement, but converge slowly as the problem size increases. The convergence analysis of Krylov methods is generally more tricky because of their right-hand side dependency, but this feature also makes them more adaptive. Nevertheless, all these methods have in common the same obstacle : the error composed of small eigenvectors is more difficult to eliminate. Likewise, the construction of Krylov subspaces relies on matrix powers, which tends to amplify the prevalence of large eigenvectors in the minimization space over the small ones. For this reason, finding a good preconditioner that accelerates the convergence of iterative methods is still a major topic of research [65].

The aim of any multigrid method [13] is to accelerate the convergence to the solution by projecting these difficult small eigenvectors onto a subspace spanned by the

columns of a sparse interpolation operator P. In fact, multigrid can be used both as a preconditioner or as a standalone solver [49]. For details regarding the parallelization of multigrid, we refer to the survey [17]. At every iteration of the method, a coarse correction computes the best approximation of the solution that belongs to the coarse space by minimizing the error in A-norm. Naturally, the A-norm requires the matrix to be SPD. We make this assumption throughout this section to introduce the fundamental ideas that govern multigrid. Assuming the coarse level is defined by way of the Galerkin triple matrix formula

$$A_c = P^T A P, \tag{2.41}$$

the coarse correction of a two-level method satisfies

$$P A_c^{-1} P^T \boldsymbol{r} = \underset{\tilde{\boldsymbol{x}} \in \text{span}\{P\}}{\arg\min} ||\boldsymbol{x} - \tilde{\boldsymbol{x}}||_A. \tag{2.42}$$

More details on the variational properties (2.42) of the coarse correction can be found in [72, Section A.2.4]. The left member of (2.42) can be interpreted as an approximate inverse of A but restricted within the range of P. Similarly to (2.3), the error propagation matrix associated with the two-level coarse correction is given by

$$E := I - P A_c^{-1} P^T A = I - \Pi_A(P). \tag{2.43}$$

As for any projection method, the error propagation matrix of the coarse correction E satisfies

$$\begin{aligned} E^2 &= \left(I - P A_c^{-1} P^T A\right)\left(I - P A_c^{-1} P^T A\right) \\ &= \left(I - P(P^T A P)^{-1} P^T A\right)\left(I - P(P^T A P)^{-1} P^T A\right) \\ &= I - 2P(P^T A P)^{-1} P^T A + P(P^T A P)^{-1} P^T A P (P^T A P)^{-1} P^T A \\ &= I - P(P^T A P)^{-1} P^T A = E. \end{aligned} \tag{2.44}$$

It follows from (2.44) that the eigenvalues of E either equal 0 or 1, and the set of eigenvectors associated with zero eigenvalues belongs to the range of interpolation. In other words, the coarse correction captures the space spanned by the columns of P, whereas the space spanned by $(I - PP^T)$ is invariant. For this reason, the coarse correction depends on how efficient is the interpolation operator in approximating the difficult and usually small eigenvectors. Deeper analysis on the effect of the coarse correction can be found in [13, 69, 81].

Assuming the coarse correction eliminates the difficult eigenvectors generally characterized by a slowly varying shape in the elliptic context, it is coupled with ν iterations of what we call a smoother. In general, the smoother corresponds to a stationary method such as Jacobi or Gauss-Seidel as introduced in Section 2.1.1. The term "smoother" simply designates that the ν iterations eliminate the high frequencies composed of large eigenvectors from the remaining error. Subsequently, the new residual is characterized by a smooth shape. Hence, the coarse correction should capture the new residual composed of low frequencies and associated with small eigenvalues. The smoother is applied before each restriction and after each interpolation to eliminate the transferred errors. Algorithm 1 recaps these steps that form a two-level cycle.

Algorithm 1 Two-level cycle

Inputs : $\boldsymbol{b}$ right-hand side, $\tilde{\boldsymbol{x}}$ approximation of $\boldsymbol{x}$ or initial guess, $\boldsymbol{r} = b - A\tilde{\boldsymbol{x}}$ residual
A initial matrix, M smoother, P interpolator
for $j = 1, \nu$ **do** ▷ ν iterations of pre-smoothing are applied
  $\tilde{\boldsymbol{x}} \leftarrow \tilde{\boldsymbol{x}} + M^{-1}\boldsymbol{r}$
  $\boldsymbol{r} \leftarrow \boldsymbol{b} - A\tilde{\boldsymbol{x}}$
end for
$\boldsymbol{r}_c \leftarrow P^T\boldsymbol{r}$ ▷ The smoothed residual is restricted to the coarse level
$\tilde{\boldsymbol{e}}_{\boldsymbol{c}} \leftarrow \text{Solve}(A_c, \boldsymbol{r}_c)$ ▷ The direct method is applied on the coarse system
$\tilde{\boldsymbol{x}} \leftarrow \tilde{\boldsymbol{x}} + P\tilde{\boldsymbol{e}}_c$ ▷ The coarse solution is interpolated to the fine level
$\boldsymbol{r} \leftarrow \boldsymbol{b} - A\tilde{\boldsymbol{x}}$
for $j = 1, \nu$ **do** ▷ ν iterations of post-smoothing are applied
  $\tilde{\boldsymbol{x}} \leftarrow \tilde{\boldsymbol{x}} + M^{-1}\boldsymbol{r}$
  $\boldsymbol{r} \leftarrow \boldsymbol{b} - A\tilde{\boldsymbol{x}}$
end for
Output : $\tilde{\boldsymbol{x}}$ approximation of $\boldsymbol{x}$ at the end of the cycle

In multilevel methods (more than two levels), the projection is repeated recursively until reaching a coarse matrix for which the factorization by a direct solver is fast, and made practical by enforcing the sparsity of P_l, with l the level index in the matrix hierarchy. In the literature, P_l is called an interpolation operator precisely because it interpolates the information from the level $l+1$ to l. This operator also determines the coarse projection subspace of each level l by enabling the construction of the coarse matrices. In most symmetric applications, these coarse matrices are constructed following the Galerkin formula $A_{l+1} = P_l^T A_l P_l$. Because the multilevel setup is equivalent to a two-level method recursively applied to solve the coarse system, we consider by default a two-level method to simplify the further discussion. Two-level methods are generally not of practical interest but help the introduction of major ideas on which multilevel methods rely.

2.2.1 Geometric Multigrid

Geometric multigrid methods [13, 78] assume the initial matrix to result from the discretization of a continuous problem over a grid Ω^h, with h the mesh size. A coarse matrix in the geometric setting can conversely be translated as a discretization over a coarser grid Ω^H where $H > h$. In this context, the interpolation operator P_H^h should enable the transfer from one grid to another such that

$$P_H^h \ : \ \mathbb{R}^{n_H} \ \rightarrow \ \mathbb{R}^{n_h} \qquad \text{and} \qquad (P_H^h)^T \ : \ \mathbb{R}^{n_h} \ \rightarrow \ \mathbb{R}^{n_H}. \tag{2.45}$$

In most common geometric multigrid methods, the coarse mesh size is generally twice that of its parent in the grid hierarchy. In that case, each coarse level has two times fewer variables in one dimension, four times fewer in two dimensions, and eight times fewer in three dimensions. Let $\mathcal{G}_L$ be the hierarchy of L grids defined as follows

$$\mathcal{G}_L := \left(\Omega^h, \Omega^{2h}, \ldots, \Omega^{2Lh}\right). \tag{2.46}$$

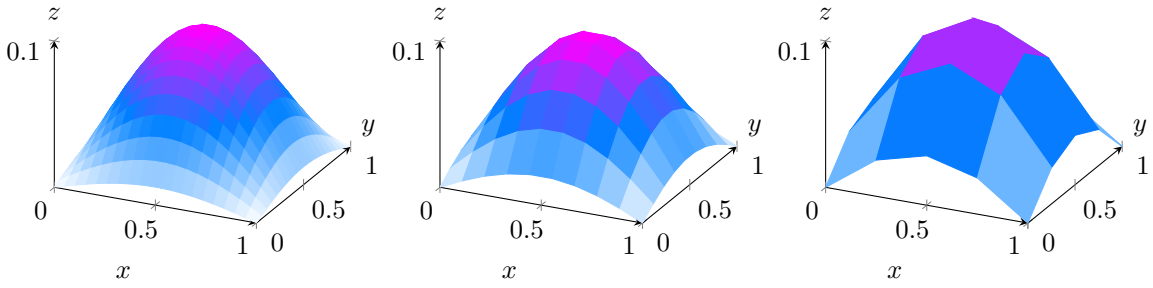
Accordingly, each matrix of the hierarchy results from the discretization of the same continuous problem over its associated grid in $\mathcal{G}_L$, which gives

$$\mathcal{A}_L := \left(A_h, A_{2h}, \ldots, A_{2Lh}\right). \tag{2.47}$$

In the geometric multigrid setting, a coarse discretization matrix of the collection (2.47) is not necessarily equivalent to the Galerkin triple matrix product (2.42).

Hence, the variational properties are not always satisfied in the geometrical case, which makes the convergence of the method more difficult to guarantee.

As mentioned previously, multigrid methods originate [40] for solving elliptic problems characterized by a dichotomy between low and high frequency eigenvectors, each respectively associated with small and large eigenvalues. Since the near-kernel space of elliptic problems is *geometrically smooth* [16] and the smoother mostly targets high frequencies, the coarse grid hierarchy should preserve the geometrically smooth information of the fine level. In particular, the Laplacian of linear functions equals zero, so P is generally designed to contain the constant function in its range. In particular, solving either the coarse problem illustrated by Figure 2.1b or Figure 2.1c provides a relevant approximation of the slowly varying eigenvector plotted on Figure 2.1a. The remaining matter is about finding an interpolation operator P that focuses on the geometrically smooth information captured from either of both coarse problems to the finest level. Enforcing P to contain the constant vector should satisfy this requirement in this example.



(a) $h = \frac{1}{20}$ (b) $h = \frac{1}{10}$ (c) $h = \frac{1}{5}$

Figure 2.1: Smallest eigenvector of a 2D Laplacian matrix with respect to the mesh size

Thereby, the goal of creating a set $\mathcal{P}_{L-1}$ of $L-1$ interpolation operators is to enable the propagation of the geometrical smoothness captured on the coarsest level up to the finest. Define

$$\mathcal{P}_{L-1} := \left(P_{2h}^{h}, P_{4h}^{2h}, \ldots, P_{2Lh}^{2(L-1)h}\right). \tag{2.48}$$

A standard rule of interpolation is to approximate each non-coarse variable by computing the mean of its neighboring coarse variables. Hence, linear vectors that are by definition the smoothest among all the vectors of Ω^h, are contained in the range of interpolation.

Example 1. *Let A_h be a one-dimensional Laplacian matrix with mesh size h. Conversely, let A_{2h} be its coarse counterpart discretized with a mesh size $2h$, and P_{2h}^{h} be*

the interpolation operator that links A_h with A_{2h} defined as follows

$$P_{2h}^h = \frac{1}{2}\begin{bmatrix} 2 & & & & & & \\ 1 & 1 & & & & & \\ & 2 & & & & & \\ & 1 & & & & & \\ & & \ddots & & & & \\ & & & 1 & & & \\ & & & 2 & & & \\ & & & 1 & 1 & & \\ & & & & 2 & & \end{bmatrix} \quad , \quad A_{2h} = \frac{1}{4h^2}\begin{bmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & & \ddots & & & \\ & & -1 & 2 & -1 & \\ & & & -1 & 2 & \end{bmatrix} \tag{2.49}$$

The Galerkin coarse matrix is linked with the coarse discretization matrix as follows

$$\frac{1}{2}\left(P_{2h}^h\right)^T A_h P_{2h}^h = A_{2h}. \tag{2.50}$$

Let the smoother of the two-level method be weighted Jacobi with $w := \frac{2}{3}$. Accordingly, define the error propagation matrix of the overall two-level method by

$$E_{TG} := (I - \frac{h^2}{3}A_h)(I - \frac{1}{2}P_{2h}^h A_{2h}^{-1}\left(P_{2h}^h\right)^T A_h)(I - \frac{h^2}{3}A_h). \tag{2.51}$$

Figure 2.2 represents the damping factors of A_h eigenvectors with respect to their associated eigenvalue, and for different methods.

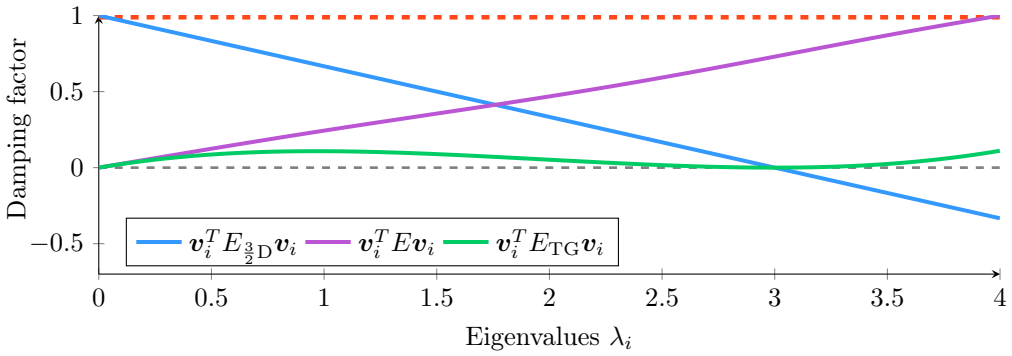


Figure 2.2: Contraction rates of the coarse correction with the Damping factors of the smoother and the two-grid method

As expected, Figure 2.2 shows that the coarse correction works in complementarity with the smoother. Indeed, while the smoother damps the large and oscillatory eigenvectors, the coarse correction captures the slowly varying small eigenvectors.

Enforcing the equivalence between the coarse correction and a minimization problem (see (2.43)) is not always simple. In Example 1, (2.50) shows that the coarse discretization matrix A_{2h} is related to the Galerkin coarse-grid. Scaling the coarse correction with the additional term 1/2 allows us to rediscretize the problem instead of computing the triple matrix product while preserving the minimization properties of the coarse correction. This equivalence does not hold for all kind of coarse

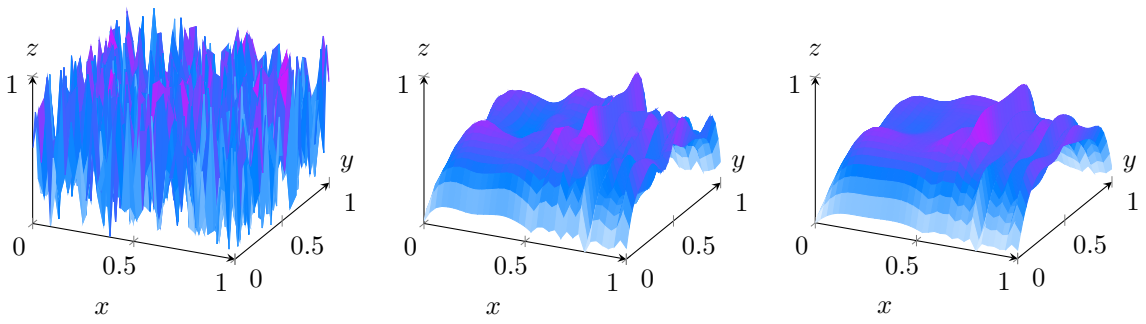
discretization matrices and interpolation rules. In fact, the Galerkin coarse-grid correction defined by the triple matrix product $P^T AP$ [47] was introduced to enable the equivalence between the coarse correction and solving a minimization problem in A-norm.

2.2.2 Algebraic Multigrid

For some problems, establishing a grid hierarchy that captures the geometrically smooth near-kernel space on the coarsest-level as in (2.46) is not simple. The Figure 2.3 represents the remaining error after several smoothing iterations for an anisotropic problem defined as follows

$$\text{(Anisotropic Diffusion)} \Leftrightarrow \begin{cases} -\left(a\dfrac{\partial^2 \boldsymbol{u}}{\partial x^2} + b\dfrac{\partial^2 \boldsymbol{u}}{\partial y^2}\right)\boldsymbol{u} = \boldsymbol{f} & \text{in } [0,1]^2 \\ a = b & \text{for } x \leq 1/2 \\ a \gg b & \text{for } x > 1/2 \end{cases} \tag{2.52}$$

This anisotropic example arises from [35], which illustrates the interest of algebraic multigrid. It shows that the geometrical smoothness can follow a special direction for which traditional geometrical coarsening approaches perform badly. Hence, finding a more general framework that better track the geometrical smoothness has been a major topic of research on multigrid methods. The comparison between the geometrical coarse correction and its algebraic counterpart is illustrated in Figure 2.4.



(a) 0 iterations (b) 10 iterations (c) 20 iterations

Figure 2.3: Remaining error with respect to the number of smoothing steps for a 2D Anisotropic Equation

Moreover, while multigrid methods are mostly used to solve PDE equations, extending the method to any kind of linear systems has also motivated the development of a more flexible approach based on the matrix entries only.

2.2.2.1 Expressing the geometrical smoothness in terms of matrix entries

Algebraic Multigrid methods (AMG) [35] originally aimed at solving elliptic problems. Hence, the goal is to design P to approximate the geometrical smoothness by looking at the matrix entries only. In fact, the only geometrical information requirement in the method is the geometrical smoothness assumption of the near-kernel

space. Hence, AMG first seeks an efficient coarse / non-coarse variable splitting denoted by the sets $\mathcal{C}/\mathcal{F}$. In a second pass, interpolation rules are constructed based on the interactions between both set of variables.

Both the design of interpolation rules and the splitting strategy of variables are based on geometrical smoothness in terms of matrix entries. Accordingly, let $\boldsymbol{e}$ be a vector of the near-kernel space of A such that

$$\boldsymbol{e}^T A \boldsymbol{e} \ll 1. \tag{2.53}$$

For instance, it is well known that the null space of the Laplacian operator is the space of linear functions. Thereby, assuming the constant vector belongs to the near-kernel space of A, it is reasonable to assume that A has row sum zero, which gives

$$a_{ii} = -\sum_{j \neq i} a_{ij}. \tag{2.54}$$

From (2.54), one can expend (2.53) to arrive at

$$e^T A e = \sum_{i<j} (-a_{ij})(e_i - e_j)^2 \ll 1. \tag{2.55}$$

As a slow variation is characterized by a small difference $(e_i - e_j)^2$, it follows that a smooth error varies slowly in the direction of large negative entries of the matrix. Therefore, these large entries are very important in the design of good interpolation rules aimed at propagating the geometrical smoothness of the near-kernel space.

2.2.2.2 Coarse and non-coarse variables splitting

The choice of coarse variables in the algebraic setting is important to construct a matrix hierarchy that preserves the geometrical smoothness of the near-kernel space. Equation (2.55) shows that this geometrical smoothness is characterized by the large entries of A. Then, the set of coarse variables $\mathcal{C}$ should contain the nodes that are the most "strongly connected" to the others by way of the largest entries of the matrix. In that way, we define the set $\mathcal{S}_i$ of variables which are strongly connected to $\boldsymbol{e}_i$, such that

$$\mathcal{S}_i := \left\{ j \ , \ -a_{ij} \geq \theta \max_{k \neq i}(-a_{ik}) \right\}. \tag{2.56}$$

with $\theta \in [0, 1]$. Choosing a relevant θ can be tricky, but its general purpose is to make the eligibility of strongly connected variables more flexible.

While coarse variables are the most strongly connected to others, the non-coarse variables conversely correspond to variables that are strongly connected to the coarse ones. This splitting strategy can be described sequentially as follows : at each iteration the node with the largest card $(\mathcal{S}_i)$ is inserted into in the group of coarse variables $\mathcal{C}$, and its strongly connected nodes that belong to $\mathcal{S}_i$ are put in the group of non-coarse variables $\mathcal{F}$. Both steps are repeated until all variables are assigned either in $\mathcal{C}$ or $\mathcal{F}$, such that

$$n_c := \mathrm{Card}(\mathcal{C}) \quad , \quad n_f := \mathrm{Card}(\mathcal{F}) \quad , \quad n_c + n_f = n. \tag{2.57}$$

This $\mathcal{C}/\mathcal{F}$ splitting is parallelizable in many ways [49, 80, 24].

2.2.2.3 Interpolation rules of non-coarse variables

Once all variables are assigned to the appropriate sets, the interpolation rules can finally be designed. Especially, the interpolation operator should approximate the geometrically smooth near-kernel space accurately. While coarse variables can simply be injected to the fine level, the purpose of the development that follows is to introduce appropriate interpolation rules for the non-coarse variables. Reordering the interpolation operator in terms of coarse and fine variables gives an interpolation operator of the form

$$(P\boldsymbol{e}_c)_i = \begin{cases} \sum_{j\in\mathcal{C}_i} w_{ij}\boldsymbol{e}_j & \text{if } i \in \mathcal{F} \\ \boldsymbol{e}_i & \text{if } i \in \mathcal{C} \end{cases} \quad \Leftrightarrow \quad P = \begin{bmatrix} W_f \\ I_c \end{bmatrix}, \tag{2.58}$$

with $\boldsymbol{e}_c$ a coarse vector of size n_c. Naturally, the $n_c \times n_c$ identity block I_c corresponds to the injection of coarse variables, while the block W_f of size $n_f \times n_c$ contains the weights of interpolation w_{ij} for the interpolation of the non-coarse variables. These weights now need to be defined.

First, let us define three sets for each non-coarse variable $i \in \mathcal{F}$, such that :

- $\mathcal{C}_i = \mathcal{C} \cap \mathcal{S}_i$ is the set of strongly connected coarse variables
- $\mathcal{F}_i = \mathcal{F} \cap \mathcal{S}_i$ is the set of strongly connected non-coarse variables
- $\mathcal{N}_i = \{j \ , \ a_{ij} \neq 0\} \cap \bar{\mathcal{S}}_i$ is the set of weakly connected variables

It follows that the right member of (2.54) can be split into three parts

$$a_{ii}\boldsymbol{e}_i = -\sum_{j\in\mathcal{C}_i} a_{ij}\boldsymbol{e}_j - \sum_{j\in\mathcal{F}_i} a_{ij}\boldsymbol{e}_j - \sum_{j\in\mathcal{N}_i} a_{ij}\boldsymbol{e}_j \quad , \quad i \in \mathcal{F} \tag{2.59}$$

where each part contributes in determining an appropriate interpolation rule for $\boldsymbol{e}_i$. First, one can use the matrix entries of the set $\mathcal{C}_i$ as follows

$$\boldsymbol{e}_i \approx -\sum_{j\in\mathcal{C}_i} \frac{a_{ij}}{a_{ii}}\boldsymbol{e}_j. \tag{2.60}$$

While this simple interpolation formula may be enough for certain problems, it can be improved with the second sum over the set $\mathcal{F}_i$. However, non-coarse variables can not be used for interpolation. Instead, they can be approximated by the coarse variables that belongs to $\mathcal{C}_i$ and by using the interpolation formula provided by (2.60), such that

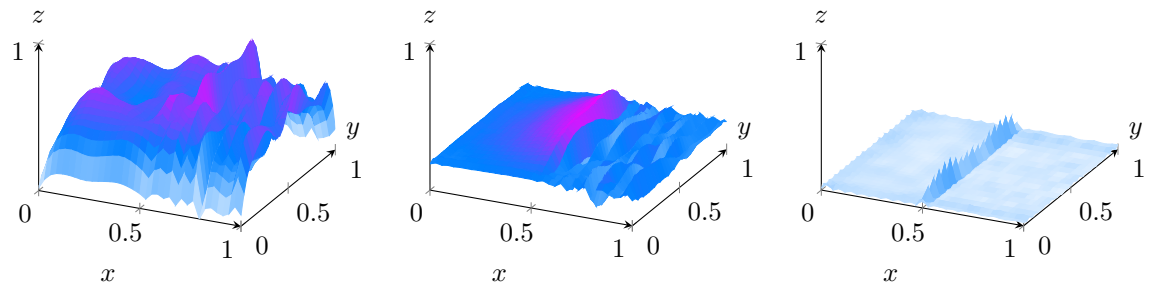
$$-\sum_{j\in\mathcal{F}_i} a_{ij}\boldsymbol{e}_j \approx -\sum_{j\in\mathcal{F}_i} a_{ij} \frac{\sum_{k\in\mathcal{C}_i} a_{jk}\boldsymbol{e}_k}{\sum_{k\in\mathcal{C}_i} a_{jk}}. \tag{2.61}$$

The small entries associated with variables of $\mathcal{N}_i$ are added to the diagonal entry of $\boldsymbol{e}_i$ (assuming $e_j \approx e_i$ for all weakly connected variables), which finally gives the weights of interpolation of (2.58)

$$w_{ij} := -\frac{a_{ij} + \sum_{k\in\mathcal{F}_i} \frac{a_{ik}\times a_{kj}}{\sum_{l\in\mathcal{C}_i} a_{kl}}}{a_{ii} + \sum_{k\in\mathcal{N}_i} a_{ik}}. \tag{2.62}$$

Since algebraic multigrid methods do not rely on geometric information (except the geometrical smoothness assumption of the near-kernel space), coarse matrices are constructed by way of the Galerkin formula (2.41).

The following Figure 2.4 compares the remaining error after the geometric coarse grid correction with the remaining error after the algebraic coarse correction on the anisotropic problem (2.52). Contrary to Figure 2.4c, the shape of Figure 2.4b varies slowly along the y-axis because the geometric multigrid coarsening does not follow the proper direction of the near-kernel space that characterizes anisotropic problems. Conversely, the remaining error after the algebraic coarse grid correction oscillates uniformly, except on the axis $x = \frac{1}{2}$ where the error reaches its maximal value. In fact, the remaining error in Figure 2.4c is small in amplitude and high in frequency precisely because the algebraic setting enables more flexible interpolation operators that better track the geometrically smooth near-kernel space.



(a) 10 iterations (b) + GMG Correction (c) + AMG Correction

Figure 2.4: Remaining error after 10 iterations vs. 10 iterations plus a Geometric Coarse Grid Correction vs. 10 iterations plus an Algebraic Coarse Grid Correction for a 2D Anisotropic problem

2.2.3 Ideal Framework

Although the classical AMG approach presented in Section 2.2.2 has proven to work remarkably well for a wide variety of problems, the original developpment of AMG heuristics was built upon the assumption that the off-diagonal entries are nonpositives [70], which can limit its applicability. New algebraic methods have been developed to address this issue such as AMGe [11, 48, 15] or Smoothed Aggregation [75, 73, 74]. An ideal framework [38, 39, 79] generalizes the algebraic multigrid concepts to address even broader classes of problems. More precisely, it gives more guidance in the design of multigrid methods such as the ideal interpolation given a fine variable space where the smoother should be the most effective. This feature is the major contribution of the ideal theory. Likewise, the ideal framework offers valuable ideas for the development of a multigrid method for the Helmholtz equation.

2.2.3.1 Motivations

The guidance of AMGe theories [11] was not only to design interpolation to approximate geometrical smoothness, but to satisfy the following heuristic :

"*Interpolation should be able to approximate an eigenvector with error bound proportional to the size of the associated eigenvalue*".

In that way, define R the operator that selects the n_c coarse variables from the initial domain. Hence, the interpolation of coarse variables is denoted by the operator $Q := PR$. For the sake of convenience, we assume $RP = I_c$ such that Q is a projection operator onto the range of P. The interpolation operator obeys with the above heuristic if it satisfies the so-called weak approximation property, such that

$$\forall \boldsymbol{e} \in \mathbb{C}^n \backslash \{0\} \quad , \quad \mu(Q, \boldsymbol{e}) := \frac{\|(I - Q)\,\boldsymbol{e}\|_2^2}{\|\boldsymbol{e}\|_A^2} \leq K, \tag{2.63}$$

for some constant K. The constant K provides a uniform bound to the two-grid convergence rate, by way of the following inequality that can be derived from (2.63)

$$\|E_{TG}\,(P)\|_A^2 \leq 1 - \frac{1}{K}. \tag{2.64}$$

In other words, satisfying the weak approximation property is a sufficient condition for two-grid method uniform convergence [77, 73]. While the numerator corresponds to the interpolation error from the coarse variables, (2.63) highlights how critical the interpolation of small eigenvectors is due to the denominator. In fact, the measure increases as $\boldsymbol{e}$ gets closer to the near-kernel space. Hence, the measure $\mu(Q, \boldsymbol{e})$ is minimal when P minimizes the interpolation error of eigenvectors in proportion with the inverse of their associated eigenvalues.

To design a coarse correction that also works in complementarity with a general form of smoother M, the range of interpolation should approximate the subspace that the smoother damps the least. In what follows, we assume that $\tilde{M} := M^T + M - A$ is SPD, which implies that the smoother designated by M is convergent (see Householder John Theorem in [63, Corollary 2.10]). Accordingly, the new approximation property provided by the ideal framework is satisfied if

$$\forall \boldsymbol{e} \in \mathbb{C}^n \backslash \{0\} \quad , \quad \mu_M(Q, \boldsymbol{e}) := \frac{\|(I - Q)\,\boldsymbol{e}\|_{\tilde{M}}^2}{\|\boldsymbol{e}\|_A^2} \leq K. \tag{2.65}$$

This is the main difference with the AMGe theory, which assumes a classical form of smoother that generally damps large eigenvectors characterized by high frequencies. In the ideal theory, the Euclidean norm of the numerator in (2.65) is replaced by the $\tilde{M}$-norm. Thereby, the ideal framework generalizes the algebraic multigrid setting because no assumption is required on M as long as $\tilde{M}$ is SPD. In fact, one can even imagine a coarse grid correction that works complementary with a near-kernel space smoother. For instance, block smoothers have been developed in the context of definite Maxwell's equations for damping its local near-kernel components [18, 19, 20].

2.2.3.2 Ideal interpolation

As in the classical algebraic setting, this generalized framework [38, 39] relies on an initial separation of coarse and non-coarse variables designated by the sets $\mathcal{C}$ and $\mathcal{F}$

respectively. In addition, the theory resorts to a coarse variable operator denoted by R^T and a fine variable operator denoted by S, such that

$$R^T \ : \ \mathbb{R}^{n_c} \to \mathbb{R}^n \quad , \quad \text{and } S \ : \ \mathbb{R}^{n_f} \to \mathbb{R}^n. \tag{2.66}$$

Naturally, because both sets $\mathcal{C}$ and $\mathcal{F}$ does not intersect

$$RS = 0 \qquad \Leftrightarrow \qquad \text{Range}(R^T) \cap \text{Range}(S) = \emptyset. \tag{2.67}$$

In fact, the space defined by the coarse variable operator R^T should be handled by the coarse correction, whereas the fine variable operator S defines the space where the smoother should work the best, in compliance with the complementarity principle. Moreover, the interpolation error is maximal for vectors in the range of S, whereas the error is zero for vectors in the range of P, such that

$$(I - Q)S = S \quad , \quad (I - Q)P = 0. \tag{2.68}$$

Beyond its benefits on the convergence analysis of algebraic multigrid methods, the generalized framework also provides the interpolation operator P_* that minimizes the generalized measure $\mu_M(Q, \boldsymbol{e})$ of (2.65). This interpolator called *ideal* is defined as follows

$$P_* := \arg\min_P \max_{\boldsymbol{e}\neq 0} \mu_M(Q, \boldsymbol{e}) = \left(I - S\left(S^T AS\right)^{-1} S^T A\right) R^T. \tag{2.69}$$

Accordingly, the minimum μ_M^* of $\mu_M(Q, \boldsymbol{e})$ is reached for $P = P_*$ such that

$$\mu_M^* = \frac{1}{\lambda_{\min}\left(\left(S^T MS\right)^{-1}\left(S^T AS\right)\right)}, \tag{2.70}$$

and the ideal two-grid convergence rate is bounded as follows

$$\|E_{TG}(P_*)\|_A^2 \leq 1 - \frac{1}{\mu_M^*} = 1 - \lambda_{\min}\left(\left(S^T MS\right)^{-1}\left(S^T AS\right)\right). \tag{2.71}$$

The left operator $(I - S(S^T AS)^{-1} S^T A)$ in (2.69) is an error propagation matrix of a projection onto the space spanned by S in A-norm. The fine variable space range(S) should be handled by the smoother but not by the coarse correction. Hence, the ideal interpolation operator works by removing from range(R^T) the S-related information that can be captured by the smoother.

Similarly, μ_M^* depends on how efficient M is in damping the S-space. Conversely, (2.70) reveals what we could call an *ideal* smoother. Assuming $S^T S = I_f$ for convenience, and letting M_f be defined by

$$M_f = S\left(S^T AS\right)^{-1} S^T, \tag{2.72}$$

then injecting (2.72) into (2.70) gives

$$\mu_{M_f}^* = 1 \ \Rightarrow \ \|E_{TG}\,\boldsymbol{e}\|_A^2 = 0. \tag{2.73}$$

In other words, one iteration of the ideal coarse correction plus one iteration M_f is equivalent to a direct method, such that

$$\left(I - P_*\left(P_*^T AP_*\right)^{-1} P_*^T A\right)\left(I - S\left(S^T AS\right)^{-1} S^T A\right) = 0. \tag{2.74}$$

2.2.3.3 Connection with the classical algebraic multigrid setting

Coarse and non-coarse variables simply result from a selection of nodes in the classical algebraic setting, so both associated R^T and S are injection operators of coarse and non-coarse variables respectively. These classical variable operators will be studied for Helmholtz in Chapter 5. Recalling the shape of P as established in (2.58), we have

$$P := \begin{bmatrix} W_f \\ I_c \end{bmatrix} \quad , \quad R := \begin{bmatrix} 0 & I_c \end{bmatrix} \quad , \quad S := \begin{bmatrix} I_f & 0 \end{bmatrix}^T , \tag{2.75}$$

where A is reordered in terms of coarse and fine variable connection blocks such that

$$A = \begin{bmatrix} S^T A R^T & S^T A S \\ RAS & RAR^T \end{bmatrix} = \begin{bmatrix} A_{ff} & A_{fc} \\ A_{cf} & A_{cc} \end{bmatrix} \tag{2.76}$$

Remark that the condition $RS = 0$ prescribed by the ideal theory (2.67) is satisfied in this setting. Injecting both variable operators defined in (2.75) in the formula of ideal interpolation (2.69) gives

$$P_* = \begin{bmatrix} -A_{ff}^{-1} A_{fc} \\ I_c \end{bmatrix} . \tag{2.77}$$

The resulting coarse matrix is a Schur complement, which corresponds to the coarse variable block of A subtracted from fine variable related information

$$P_*^T A P_* = A_{cc} - A_{cf} A_{ff}^{-1} A_{fc}. \tag{2.78}$$

Section A.3 presents an idealistic scenario given another pair of coarse and non-coarse variable operators. The example also appears in [38, Corollary 3.4].

2.2.3.4 Interest and practical implementations

Design of algebraic multigrid methods is generally driven by the theory. The ideal framework provides a theoretical definition of an ideal interpolation based on an initial pair of coarse and fine variables, and inspired an extensive amount of practical multigrid methods. Certain problems have convenient structural properties that can be exploited to construct the ideal interpolation and the Schur complement at lower cost. For certain discretization stencils such as the 5-point stencil introduced in (1.2), separating coarse and non-coarse variables based on a Red-Black coloring scheme gives a diagonal A_{ff} block. Hence, both ideal interpolation and M_f can be computed exactly, which gives an exact solver as highlighted in (2.74).

Although rarely computed exactly, the strategy of approximating the ideal interpolation and its compatible smoother as in (2.74) drives *reduction* methods [62, 84], and has proven to solve a wide variety of problems. In particular, *Multigrid-In-Time* [36] solvers tackle time-dependent problems and resort extensively to reduction-based ideas. Other reduction-based methods approximate the ideal interpolation operator by solving an optimization problem that seeks the best set of coarse variables [61, 85] to enable a more practical approximate inverse of the fine variable block A_{ff} in (2.77).

2.2.4 Optimal Framework

As described in the previous section, the term "ideal" is used because P_* minimizes the measure (2.65) given an initial coarse and non-coarse variable splitting. Such a splitting is used in most classical algebraic multigrid settings in practice, so the ideal framework is interesting as a guide for algorithm development. However, the bound on the convergence rate provided in (2.71) is not sharp, which means that for some problem, there might exist another interpolation operator based on different coarse variables that work better than one particular P_*.

2.2.4.1 Optimal approximation property

The optimal framework provides the "optimal" interpolation operator $P_\#$ for a given smoothing matrix M, such that the resulting convergence rate of the method is sharp [73, 10]. Let $\Pi_{\tilde{M}}$ be the $\tilde{M}$-orthogonal projection onto Range(P) defined as follows

$$\Pi_{\tilde{M}} := P\left(P^T\tilde{M}P\right)^{-1}P^T\tilde{M} \quad \text{such that} \quad \Pi_{\tilde{M}}\boldsymbol{e} = \underset{\tilde{\boldsymbol{e}}\in\text{Range}(P)}{\arg\min} \|\boldsymbol{e}-\tilde{\boldsymbol{e}}\|_{\tilde{M}}\,. \tag{2.79}$$

In a multigrid cycle, the best among all possible interpolation operators minimizes the interpolation error in proportion with the inverse of associated eigenvalues as in the weak-approximation property (2.63), but also the overlap with the smoother. This feature is the key for a good complementarity between the coarse correction and the smoother. Here, the measure provided by the optimal framework is

$$\forall \boldsymbol{e}\in\mathbb{C}^n\backslash\{0\} \qquad , \qquad u_{\tilde{M}}(P,\boldsymbol{e}) := \frac{\|(I-\Pi_{\tilde{M}})\boldsymbol{e}\|^2_{\tilde{M}}}{\|\boldsymbol{e}\|^2_A}. \tag{2.80}$$

Here again, the $\tilde{M}$-norm amplifies over the A-norm the eigenvectors that the smoother struggles to capture, and the denominator emphasizes the small eigenvectors.

2.2.4.2 Optimal interpolation and generalized eigenvalue problem

Let $P_\#$ be the optimal interpolation operator that minimizes $\mu_{\tilde{M}}(P,\boldsymbol{e})$ such that

$$P_\# := \underset{P}{\arg\min}\,\max_{\boldsymbol{e}\neq 0}\,\mu_{\tilde{M}}(P,\boldsymbol{e}). \tag{2.81}$$

The following lemma can be found in [10, Lemma 1] and connects the form of the optimal interpolation operator with the solution to a generalized eigenvalue problem.

Lemma 1. *Let $P : \mathbb{R}^{n_c} \to \mathbb{R}^n$ be full rank and let $\sigma_1 \leq \sigma_2 \leq \ldots \leq \sigma_n$ and $\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_n$ denote the eigenvalues and eigenvectors of the generalized eigenvalue problem*

$$A\boldsymbol{w}_i = \sigma_i\tilde{M}\boldsymbol{w}_i. \tag{2.82}$$

Then the optimal convergence rate of the two-grid method is given by

$$\|E_{TG}\,(P_\#)\|^2_A = 1-\frac{1}{\mu^\#_{\tilde{M}}} \qquad \textit{with} \qquad \mu^\#_{\tilde{M}} = \frac{1}{\sigma_{n_c+1}}, \tag{2.83}$$

where the optimal interpolation operator $P_\#$ satisfies

$$\textit{Range}(P_\#) = \textit{span}\,(\{\boldsymbol{w}_1,\ldots,\boldsymbol{w}_{n_c}\})\,. \tag{2.84}$$

The two-grid convergence rate provided by (2.83) is the best among all the possible interpolation operators P for a given smoother M. Worth noting how similar is (2.83) to the convergence rate provided by the ideal theory in (2.71). Since the space spanned by P is damped by the coarse correction, the overall convergence rate depends on the remaining space and how the smoother damps it. In other words, the eigenvalue σ_{n_c+1} represents the deviation of the smoother in approximating the energy that goes in the direction of least captured eigenvector $\boldsymbol{w}_{n_c+1}$.

That said, the condition (2.84) is difficult to satisfy because computing the n_c smallest eigenvectors of the generalized eigenvalue problem (2.82) is expensive. In addition, P should have a reasonable sparsity to remain practical.

Finally, [10, Lemma 2] demonstrates that both ideal and optimal frameworks are reconnected if the coarse and the fine variable operators $R_{\#}^T$ and $S_{\#}$ satisfy

$$\text{Range}\left(R_{\#}^T\right) = \{\boldsymbol{w}_1\ , \dots\ , \boldsymbol{w}_{n_c}\} \quad , \quad \text{Range}\left(S_{\#}\right) = \{\boldsymbol{w}_{n_c+1}\ , \dots\ , \boldsymbol{w}_{n}\}\,. \tag{2.85}$$

As a consequence, the ideal interpolation operator resulting from (2.85) is also optimal, and has the form

$$P_{\#} = P_* = (I - S_{\#}(S_{\#}^T A S_{\#})^{-1} S_{\#}^T A) R_{\#}^T, \tag{2.86}$$

such that the optimal convergence rate (2.83) is given by

$$\mu_M^{\#} = \frac{1}{\sigma_{n_c+1}} = \frac{1}{\lambda_{\min}\left(\left(S_{\#}^T M S_{\#}\right)^{-1}\left(S_{\#}^T A S_{\#}\right)\right)}. \tag{2.87}$$

Chapter 3

Multigrid for Helmholtz

The multigrid theory introduced in Chapter 2 exhibits the importance of a coarse correction that works in complementarity with the smoother (e.g., Figure 2.2). In elliptic problems, usual smoothers capture oscillatory errors associated with large positive eigenvalues. Hence, a good coarse correction results from an interpolation error that minimizes the remaining smooth errors associated with small positive eigenvalues. For these reasons, the geometrical smoothness of the near-kernel space and also the symmetric positive definiteness of the matrix are two key assumptions in most of multigrid methods.

Nevertheless, these convenient assumptions do not hold for all problems. This concern was first mentioned by Bakhvalov in the 1960s [4] and further developed by Brandt in the 1980s [9] for both nearly singular and slightly indefinite problems. In certain cases, the traditional smoothers do not satisfy the Householder-John Theorem [63, Corollary 2.10] and amplify certain eigenvectors as a consequence. In particular in the indefinite case, the traditional smoothers based on the stationary methods introduced in Section 2.1.1 tend to amplify the negative eigenvectors generally characterized by a slowly varying shape. The initial matrix may have negative eigenvalues but a positive constant diagonal as in (1.2). In that case, a Jacobi method would approximate the negative eigenvectors in the wrong direction. While the amplification of the error may only slow the convergence down, it can also lead to divergence if the coarse correction does not capture the amplified error properly. Moreover, certain types of coarse correction do not satisfy the variational properties that ensure their robustness. The latter issue happens in certain geometrical settings because the coarse matrices arise from a rediscretization of the initial continuous problem. Hence, they are not equivalent to the Galerkin triple matrix product (2.41). While these methods may provide fast convergence in practice, they can easily amplify the error if the problem is nearly singular or indefinite.

Typically, Helmholtz problems are characterized by indefinite matrices for which traditional smoothers and coarse corrections do not perform as in the SPD case. The Helmholtz equation can be seen as a shifted Laplacian equation where geometrically smooth eigenvectors (i.e., low Fourier modes, see Figure 3.1b) can be negative eigenvectors because of the shift. In the same way, the smallest eigenvectors of the shifted Laplacian (1.2) are higher in frequency (see Figure 3.2b). As a consequence, three main issues arise when solving the Helmholtz equation with multigrid. First,

usual smoothers amplify the negative eigenvectors as already mentioned. Secondly, small eigenvectors for which the interpolation error should be minimum are not geometrically smooth because of the shift. In fact, the near-kernel space of the Helmholtz equation is oscillatory, and interpolation operators should be designed to approximate it regardless of its inconvenient shape. Lastly, the discretization matrix is indefinite, which breaks the connection between Galerkin-based coarse corrections and energy-norm minimization principles.

In this chapter, we begin with a survey of the past attempts aimed at adapting multigrid methods to the indefinite and oscillatory Helmholtz equation. We conclude this chapter by demonstrating how the coarse correction can be corrupted by the error of interpolation through a concept of "pollution", although P has good approximation properties of the target near-kernel space.

3.1 State of the art on multigrid for Helmholtz

While other solvers exist for Helmholtz [55, 14], we only present multigrid solver based attempts. The Section 3.1.1 emphasizes the difficulties encountered by multigrid by way of a simple one-dimensional model problem. In this section, both the smoother and the coarse correction are analyzed with respect to the discretization coefficient kh to stick with the terminology of the literature, whereas we use α in the rest of the manuscript. The connection between the discretization coefficient kh and the shift α in (1.2) is provided by the relation $\alpha = (kh)^2$. Therefore kh also measures how indefinite the resulting discretization matrix is. For instance, $kh = 0.625$ corresponds to 10 per wavelength ω ($h = \omega/10 = 2\pi/10k \Leftrightarrow kh = 0.625$), which means shifting the eigenvalues of the one-dimensional Laplacian matrix by $\alpha = 0.625^2$ to the negative. Coarser discretization yields larger kh, and subsequently more negative eigenvalues.

3.1.1 One-dimensional spectral analysis of the multigrid operators

The spectral analysis of two-grid methods often relies on the one-dimensional Laplace modal analysis [44, chapter 2] also developed in [13, chapter 5]. This very classical development has been extended in [29, 33] to the following one-dimensional Helmholtz model problem

$$\text{(1D model problem)} \quad \Leftrightarrow \quad \left\{ \begin{array}{rcl} -\Delta \boldsymbol{u} - k^2 \boldsymbol{u} & = \boldsymbol{f} & \text{in } \Omega = [0,1] \\ u & = 0 & \text{on } \partial\Omega \end{array} \right. , \tag{3.1}$$

on which is based the discussion of this section. Thereupon, the finite difference discretization matrix resulting from (3.1) used in this discussion is

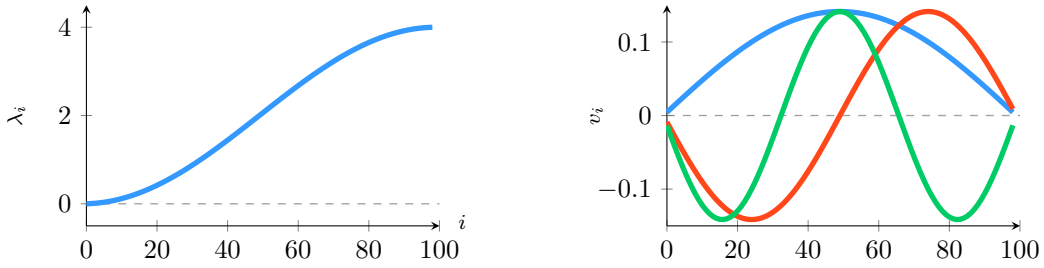
$$A := \frac{1}{h^2} \text{Tridiag}\left(-1, 2 - (kh)^2, -1\right) \quad , \quad h := \frac{1}{n+1} \tag{3.2}$$

where k is the wavenumber, h the mesh size and n the size of A. The model problem (3.1) is convenient to study because a Fourier analysis gives the exact eigenvalues and eigenvectors of A. The discretization matrix A defined in (3.2) is a shifted

Laplacian matrix whose eigenvalues and eigenvectors are defined as follows

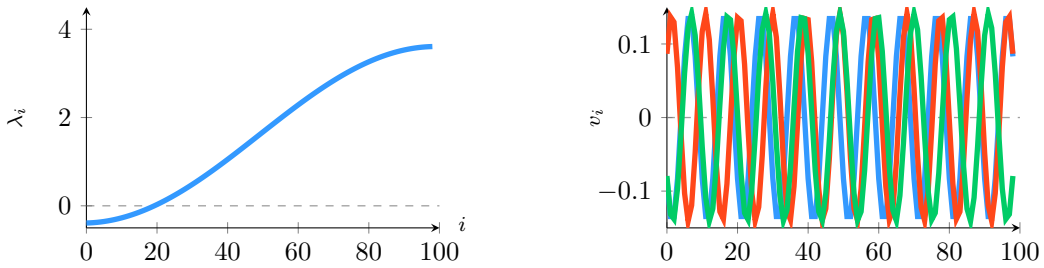
$$\lambda_j^h = \frac{2 - 2\cos(j\pi h)}{h^2} - k^2 \quad , \quad \boldsymbol{v}_j^h = [\sin(j\pi lh)]_{l=1}^n \quad , \quad \forall\, j = 1, \ldots, n. \tag{3.3}$$

The three eigenvectors associated with the three smallest magnitude eigenvalues in the non-shifted case are illustrated in Figure 3.1b, whereas Figure 3.2b portrays their counterparts when the discretization coefficient is $kh = 0.625$. As mentioned in the introduction of this section, the indefiniteness of the matrix depends on the discretization coefficient kh. We say that the matrix is highly indefinite when kh is close to $\sqrt{2}$ because its spectrum is characterized by a balance between negative and positive eigenvalues. We also note that the matrix becomes negative definite when $kh > 2$. Now that the model problem on which relies this chapter is introduced, let us detail how the multigrid operators are affected by the discretization coefficient kh when applied to Helmholtz.



(a) Eigenvalues (b) Eigenvectors

Figure 3.1: 1D Laplace eigenvalues and the three smallest eigenvectors



(a) Eigenvalues (b) Eigenvectors

Figure 3.2: 1D Helmholtz eigenvalues and the three smallest eigenvectors, $kh = 0.625$

3.1.1.1 The problem of the smoother

Let us consider a Jacobi smoother as introduced in the Section 2.1.1 dedicated to stationary iterative methods. Applied to the discretization matrix (3.2), the approximate inverse in the Jacobi method is $D := \mathrm{Diag}\,(A)$. The error propagation matrix is defined by

$$E_{\mathrm{Jac}} := I - D^{-1}A = -\frac{1}{2 - (kh)^2}\mathrm{Tridiag}\,(1, 0, 1)\,. \tag{3.4}$$

The error propagation matrix E_{Jac} has the same eigenvectors as A, but its eigenvalues are given by

$$\lambda_j\left(E_{\text{Jac}}\right) = 1 - \frac{\lambda_j^h}{2-(kh)^2} = \frac{2\cos(j\pi h)}{2-(kh)^2}. \tag{3.5}$$

The Jacobi method applied to the model problem (3.1) converges if its spectral radius is strictly less than one. As a consequence, we have

$$\left|\lambda_j\left(E_{\text{Jac}}\right)\right| < 1 \quad \Leftrightarrow \quad \left|\cos\left(j\pi h\right)\right| < \left|1 - \frac{(kh)^2}{2}\right| \quad , \quad j = 1, \ldots, n. \tag{3.6}$$

The condition of convergence provided by (3.6) is especially difficult to satisfy for the largest values of the cosine function reached when $j \approx 1$ and $j \approx n$ (i.e., $\cos(\pi h) \approx 1$ and $\cos(n\pi h) \approx -1$). The region $j \approx 1$ corresponds to the negative eigenvalues of A whose associated eigenvectors are geometrically smooth. Conversely, the region $j \approx n$ corresponds to the largest eigenvalues of A whose associated eigenvectors are very oscillatory. In other words, both smooth and very oscillatory modes are amplified in this case. By contrast, intermediate eigenvectors are damped because the cosine function is smaller, which helps satisfy the condition of convergence (3.6). We also remark that the right term $\left|1-(kh)^2/2\right|$ decreases as the discretization coefficient kh tends to $\sqrt{2}$. As a consequence, the number of modes that Jacobi is able to damp decreases as the mesh discretization gets coarser, and more eigenvectors are amplified.

The weighted-Jacobi smoother introduced in Section 2.1.1 helps the damping of a certain portion of the spectrum. In the context of solving the Helmholtz equation, adding a weight still provides a significant help in damping the very oscillatory modes that unweighted Jacobi tends to amplify. Let λ_{mid} be the midpoint between the largest negative and positive eigenvalues, such that

$$\lambda_{\text{mid}} := \frac{\lambda_1^h + \lambda_n^h}{2}. \tag{3.7}$$

Assuming λ_{mid} is positive, the optimal weight w_{opt} that allows us to efficiently damp the half most oscillatory eigenvectors whose eigenvalues belongs to the interval $[\lambda_{\text{mid}}, \lambda_n^h]$ is obtained by solving

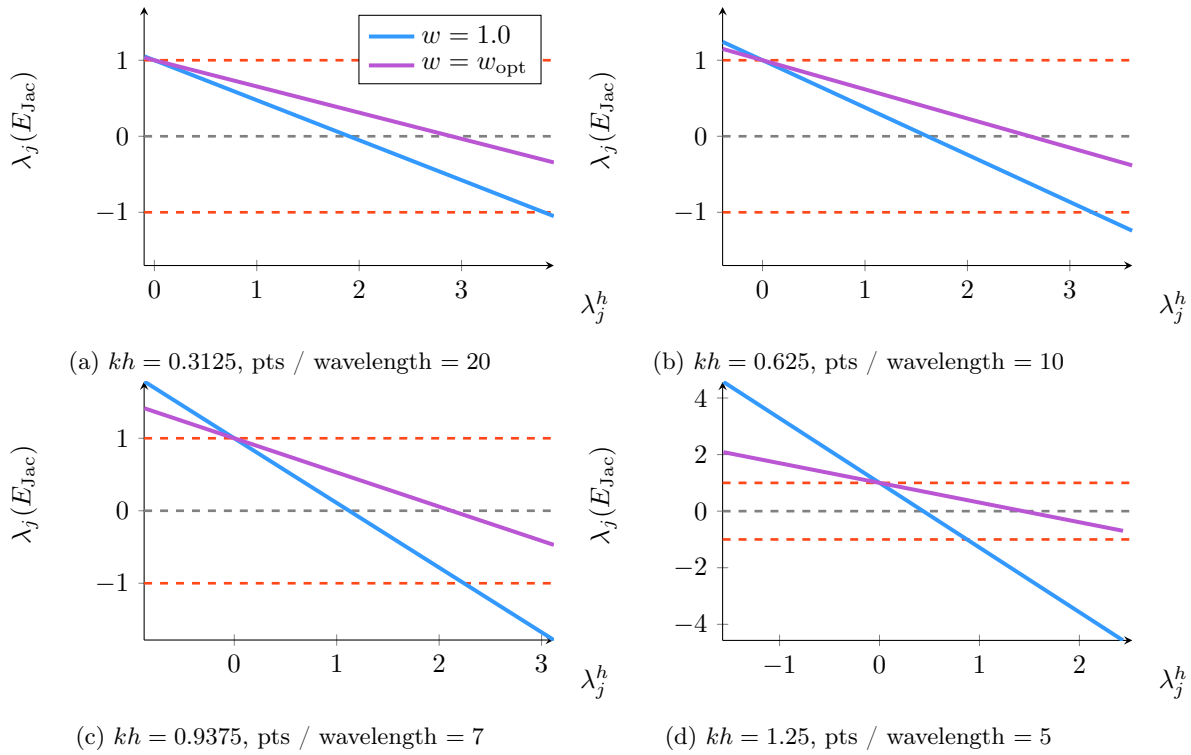
$$1 - w_{\text{opt}}\frac{\lambda_{\text{mid}}}{2-(kh)^2} = -\left(1 - w_{\text{opt}}\frac{\lambda_n^h}{2-(kh)^2}\right). \tag{3.8}$$

which leads to the optimal weight for the one-dimensional model problem (3.1)

$$w_{\text{opt}} = \frac{2-(kh)^2}{3-(kh)^2}. \tag{3.9}$$

Below figures plot the damping factor of each eigenvector with one Jacobi smoothing iterations with respect to the discretization coefficient kh and for two values of the weight w.

As expected, Figure 3.3 highlights how Jacobi with no weight amplifies both small and large eigenvalues with respect to kh. Even though adding the optimal weight



(a) $kh = 0.3125$, pts / wavelength $= 20$

(b) $kh = 0.625$, pts / wavelength $= 10$

(c) $kh = 0.9375$, pts / wavelength $= 7$

(d) $kh = 1.25$, pts / wavelength $= 5$

Figure 3.3: Damping factors of the Jacobi method with respect to kh and w.

(3.9) improves the capture of oscillatory modes, the smooth ones associated with negative eigenvalues are still amplified. The two-grid method would diverge in this case if the coarse correction does not contract these eigenvectors sufficiently.

While this analysis has been performed for Jacobi, the reasoning is the same for other traditional stationary methods such as Gauss-Seidel or Richardson.

3.1.1.2 The problem of the coarse correction

This section illustrates the difficulty that the coarse correction encounters when solving the indefinite Helmholtz equation. In particular, applying the classical geometrical setting to the model problem (3.1) reveals the interaction between the discretization coefficient and the robustness of the coarse correction. The next development originates from [29].

Let the number of fine variables n be odd, so that the $N := \frac{n-1}{2}$ coarse variables correspond to interior points of the fine grid. Let A_H be the coarse counterpart of $A_h := A$ with coarser mesh size $H := 2h$. Naturally, the associated coarse-grid eigenvectors are $\boldsymbol{v}_j^H = [\sin(j\pi lH)]_{l=1}^N$. One can show that the first N fine-grid eigenvectors are related to the last N fine-grid eigenvectors by the relation

$$\left[\boldsymbol{v}_j^h\right]_i = (-1)^{i+1} \left[\boldsymbol{v}_{n+1-j}^h\right]_i \qquad , \qquad j = 1, \ldots, N. \tag{3.10}$$

In addition, let the coarse-to-fine transformation be given by the uniform interpolation operator

$$\left[P_H^h \boldsymbol{e}^H\right]_i := \begin{cases} \boldsymbol{e}^H_{i/2} & \text{if } i \text{ even}, \\ \boldsymbol{e}^H_{(i-1)/2} + \boldsymbol{e}^H_{(i+1)/2} & \text{if } i \text{ odd}, \end{cases} \qquad 1 \le i \le n. \tag{3.11}$$

Note that P_H^h is the very classical form of interpolation introduced in (2.49) and is often called "linear" or "uniform". From both equations (3.10) and (3.11), it follows that

$$P_H^h \boldsymbol{v}_j^H = c_j^2 \boldsymbol{v}_j^h - s_j^2 \boldsymbol{v}_{n+1-j}^h \qquad , \qquad j = 1, \ldots, N, \tag{3.12}$$

with $c_j := \cos(\frac{j\pi h}{2})$ and $s_j := \sin(\frac{j\pi h}{2})$. Since coarse variables are interior points of the fine grid, the fine-to-coarse average transfer operator

$$P_h^H := \frac{1}{2}\left(P_H^h\right)^T \tag{3.13}$$

can be formulated element-wise as follows

$$\left[P_h^H \boldsymbol{e}^h\right]_i := \frac{1}{4}\left(\left[\boldsymbol{e}^H\right]_{2i-1} + 2\left[\boldsymbol{e}^H\right]_{2i} + \left[\boldsymbol{e}^H\right]_{2i+1}\right) \quad , \quad j = 1, \ldots, N. \tag{3.14}$$

Therefore, the following mapping properties of the restriction operator when applied to the jth fine-grid eigenvector $\boldsymbol{v}_j$ are given by

$$P_h^H \boldsymbol{v}_j^h = \begin{cases} c_j^2 \boldsymbol{v}_j^H, & 1 \le j \le N \\ 0, & j = N+1 \\ -c_j^2 \boldsymbol{v}_{n+1-j}^H, & N+2 \le j \le 2N+1 \end{cases}. \tag{3.15}$$

Then, let E be the error propagation matrix of the coarse correction such that

$$E := I - P_H^h A_H^{-1} P_h^H A_h \tag{3.16}$$

While (3.12) shows that the interpolation of coarse-grid eigenvectors results in a combination of fine-grid eigenvectors, (3.15) shows that the restriction of fine-grid eigenvectors results in a combination of coarse-grid eigenvectors. Both properties allow us to derive the remaining error after applying the coarse correction to $\boldsymbol{v}_j$ by

$$E\boldsymbol{v}_j^h = \begin{cases} \left(1 - c_j^4 \dfrac{\lambda_j^h}{\lambda_j^H}\right) \boldsymbol{v}_j^h + s_j^2 c_j^2 \dfrac{\lambda_j^h}{\lambda_j^H} \boldsymbol{v}_{n+1-j}^h, & 1 \le j \le N \\ \boldsymbol{v}_j^h, & j = N+1 \\ \left(1 - c_j^4 \dfrac{\lambda_j^h}{\lambda_{n+1-j}^h}\right) \boldsymbol{v}_j^h + s_j^2 c_j^2 \dfrac{\lambda_j^h}{\lambda_{n+1-j}^h} \boldsymbol{v}_{n+1-j}^h, & N+2 \le j \le 2N+1 \end{cases}$$

As a consequence, the two-dimensional spaces spanned by the pairs of complementary fine-grid eigenvectors are invariant under E such that

$$E\left[v_j^h, v_{n+1-j}\right] = \left[v_j^h, v_{n+1-j}\right] E_j, \qquad j = 1, \ldots, N, \tag{3.17}$$

where the 2×2 matrix E_j is defined as follows

$$E_j := \begin{bmatrix} 1 - c_j^4 \dfrac{\lambda_j^h}{\lambda_j^H} & s_j^2 c_j^2 \dfrac{\lambda_{n+1-j}^h}{\lambda_j^H} \\ s_j^2 c_j^2 \dfrac{\lambda_j^h}{\lambda_j^H} & 1 - s_j^4 \dfrac{\lambda_{n+1-j}^h}{\lambda_j^H} \end{bmatrix} \quad , \qquad j = 1, \dots, N. \tag{3.18}$$

Hence, the contraction rate of the fine-grid eigenvector $\boldsymbol{v}_j^h$ is

$$v_j^T E v_j = [E_j]_{1,1} = 1 - c_j^4 \frac{\lambda_j^h}{\lambda_j^H}. \tag{3.19}$$

Moreover, since fine and coarse eigenvalues can be respectively formulated by

$$\lambda_j^h = \frac{4}{h^2} s_j^2 - k^2 \quad \text{and} \quad \lambda_j^H = \frac{4}{h^2} s_j^2 c_j^2 - k^2, \tag{3.20}$$

the ratio of the fine eigenvalue over the coarse eigenvalue in (3.19) equals

$$\frac{\lambda_j^h}{\lambda_j^H} = \frac{s_j^2 - \left(\frac{kh}{2}\right)^2}{s_j^2 c_j^2 - \left(\frac{kh}{2}\right)^2} = 1 + \frac{s_j^4}{s_j^2 c_j^2 - \left(\frac{kh}{2}\right)^2}. \tag{3.21}$$

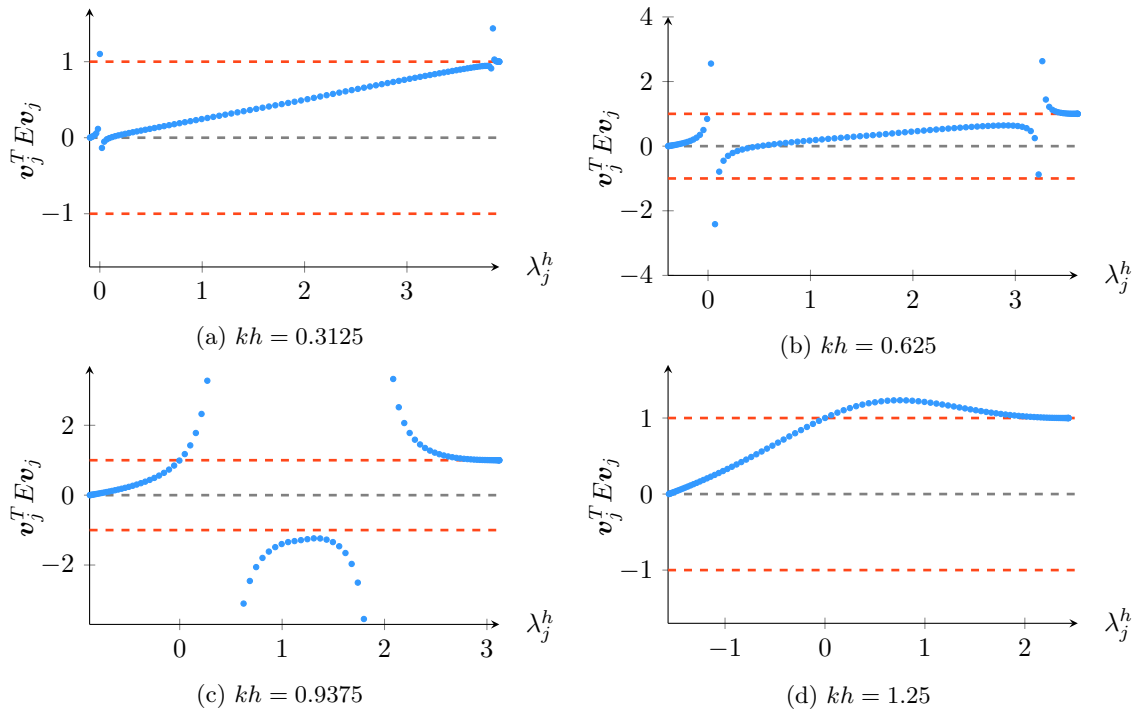
Equation (3.21) highlights the effect of the discretization coefficient on the effectiveness of the coarse correction. For instance, very fine and very coarse meshes yield positive and negative definite discretization matrices respectively, such that the coarse correction never amplifies the modes in either scenario. However, this statement is not true for intermediate mesh sizes. In particular, the ratio (3.21) tends to explode when $2 s_j c_j \approx kh$ (i.e., $\lambda_j^H \approx 0$). The coarse correction operates as a contraction on $\boldsymbol{v}_j$ if

$$\left| \boldsymbol{v}_j^T E \boldsymbol{v}_j \right| < 1 \quad \Leftrightarrow \quad \begin{cases} (kh)^2 < 4 s_j^2 c_j^2 \left(1 - \dfrac{s_j^2 c_j^2}{2 - c_j^4} \right), & \text{if } 2 s_j c_j > kh \\ (kh)^2 > 4 s_j^2, & \text{if } 2 s_j c_j < kh, \end{cases} . \tag{3.22}$$

The quantity $s_j^2 c_j^2$ is symmetric, such that

$$\begin{aligned} s_j^2 c_j^2 &= \sin\left(\frac{j\pi h}{2}\right)^2 \cos\left(\frac{j\pi h}{2}\right)^2 = \sin\left(\frac{\pi}{2} - \frac{j\pi h}{2}\right)^2 \cos\left(\frac{\pi}{2} - \frac{j\pi h}{2}\right)^2 \\ &= \sin\left(\frac{(n+1-j)\,\pi h}{2}\right)^2 \cos\left(\frac{(n+1-j)\,\pi h}{2}\right)^2 = s_{n+1-j}^2 c_{n+1-j}^2. \end{aligned} \tag{3.23}$$

In the case where $2 s_j c_j > kh$, the symmetry of $s_j^2 c_j^2$ implies that the coarse correction amplifies any of the N first fine-grid eigenvectors $\boldsymbol{v}_j^h$ if the complementary eigenvector v_{n+1-j}^h is amplified. Conversely, the coarse correction damps any of the N last fine-grid eigenvector $\boldsymbol{v}_{n+1-j}^h$ if the complementary eigenvector $\boldsymbol{v}_j^h$ is damped. In the other case where $2 s_j c_j < kh$, the relation $s_{n+1-j}^2 = c_j^2$ implies that the coarse correction amplifies any of the last N eigenvectors if the complementary is amplified, and damps any of the first N eigenvectors if the complementary is damped. Such a symmetry in the effect of the coarse correction appears in Figure 3.4. Beyond the amplification of certain modes, the blue curves are also characterized by a linear growth due to the increasing coefficient c_j^4 that weights the ratio $\lambda_j^h / \lambda_j^H$ in (3.19).



(a) $kh = 0.3125$ (b) $kh = 0.625$

(c) $kh = 0.9375$ (d) $kh = 1.25$

Figure 3.4: Contraction rates of the coarse correction with respect to kh.

The discussion of this section focused on the problem of the smoother and of the coarse correction on a simple one-dimensional Helmholtz model problem. Typically, stationary smoothers and the coarse correction require the positive definiteness of the initial matrix, which is not true for Helmholtz. Thus, these operators obligate a special attention when designing a multigrid method in the indefinite case. In addition, the near-kernel space of the Helmholtz equation is oscillatory so that the interpolation rules used in elliptic problems are not appropriate for transferring the smallest eigenvectors through the grid hierarchy. While finding a smoother and a coarse correction that enable the convergence of the method is challenging when tackling indefinite problems, designing a fast and practical method makes the problem even more demanding. Before elaborating on the problems of the coarse correction through the concept of pollution introduced in Section 3.2.1, we present the major contributions aimed at designing an appropriate multigrid method for the indefinite Helmholtz equation.

3.1.2 Alternative smoothers for Helmholtz

As introduced above, stationary smoothers are not suitable smoothers for damping the negative eigenvalues that characterize the indefinite Helmholtz equation. The Kaczmarz method [44, 9] is often presented as a good alternative smoother for Helmholtz. One iteration of this smoother is equivalent to a Gauss-Seidel relaxation sweep on normal equations, and therefore leads to the following entry-wise formulation

$$\boldsymbol{x}_i^{(k+1)} = \boldsymbol{x}_i^{(k)} + \overline{a_{ij}} \frac{r_i^{(k)} - \sum_{j<i} a_{ij} x_j^{(k+1)}}{|a_{ii}|^2}, \tag{3.24}$$

where $\overline{a_{ij}}$ denotes the complex conjugate of a_{ij}. While Kaczmarz does not amplify the error, the damping of the oscillatory modes tends to be slower than classical smoothers in the positive case. Subsequently, finding a method that enables fast convergence in the indefinite case motivated the community to investigate more options. Since Jacobi offers both effective and straightforward parallel implementations, a two-step version has been developed in [45]. This extension of Jacobi makes it applicable to the indefinite Helmholtz equation. An extensive presentation of this smoother can also be found in [34]. Let E_{2Jac} be the error propagation matrix of the two-step Jacobi smoother such that

$$E_{2Jac} := \left(I - w_1 D^{-1} A\right)\left(I - w_2 D^{-1} A\right). \tag{3.25}$$

Also, define

$$\delta := (2 - (kh)^2)/h^2. \tag{3.26}$$

The eigenvalues of E_{2Jac} can be formulated as follows

$$\lambda_j\left(E_{2Jac}\right) = f\left(\lambda_j(A)\right), \quad \text{with} \quad f(\lambda) = \left(1 - w_1\frac{\lambda}{\delta}\right)\left(1 - w_2\frac{\lambda}{\delta}\right). \tag{3.27}$$

In fact, the method is called "two-step" because it relies on two separate smoothing iterations, each associated with weights w_1 and w_2 respectively. As discussed previously, the weight can have a strong impact on Jacobi's convergence properties, and is usually employed to speed-up the damping of most oscillatory modes. In the context of solving indefinite equations with this two-step smoother, both weights are mostly chosen to avoid the amplification of negative eigenvalues.

Setting the weights w_1 and w_2 depends on three cases. The first case applies to shifted Laplacian matrices whose diagonal are positive (i.e., $kh < \sqrt{2}$). In that case, the midpoint λ_{mid} defined in (3.7) is positive. The two-step Jacobi variant should damp the half most oscillatory modes the fastest. Conversely, its slowest damping rate should be for near-zero eigenvalues, as in the positive definite case. Both conditions are respectively expressed as follows

$$f\left(\lambda_{\text{mid}}\right) = -f\left(\lambda_n\right) \qquad \text{and} \qquad f'\left(0\right) = 0. \tag{3.28}$$

Moreover, the f function of (3.27) is a polynomial of degree 2 with only one extremal point reached at 0, such that $f(0) = 1$. From both conditions (3.28), one can derive the following weights

$$w_1 = \frac{2\sqrt{2}}{\sqrt{\left(\lambda_1 + \lambda_n\right)^2 + 4\lambda_n^2}} \qquad , \qquad w_2 = -w_1. \tag{3.29}$$

To compare the two-step variant of Jacobi with the original, we assume a reduction factor $\epsilon = 1/3$ of the largest eigenvector after one single-step Jacobi smoothing for a Laplace problem. We consider the largest and most oscillatory eigenvector because we assume that the coarse correction focuses on the negative and geometrically smooth eigenvectors. More details on the provided numbers can be found in [34]. When smoothing the one-dimensional model problem (3.2) by using the two-step

Jacobi variant with weights (3.29), the required number of iterations to reach the same reduction factor is

$$\nu \geq \frac{-\log \epsilon}{\log\left((6-2k^2h^2)/(10-6k^2h^2+k^4h^4)\right)}. \tag{3.30}$$

For $kh = 0.5$, we need to compute $\nu = 3$ iterations to reach the same reduction factor, and $\nu = 5$ iterations for $kh = 1$. While the convergence is guaranteed in this first scenario, the method requires more computations to reach the convergence of the positive case. This phenomenon is illustrated in Figure 3.5a, where orange and blue curves represent respectively the damping factors after a one-step Jacobi iteration with the weight (3.9) and one iteration with the two-step alternative. While the orange curve exceeds one for negative eigenvalues, it demonstrates better damping factors than the blue curve for large positive eigenvalues.

The conditions for setting both weights are slightly different in the second scenario where $\lambda_{\text{mid}} < 0 < \lambda_n$. This time, the effectiveness of the smoother is ensured by enforcing the f function in (3.27) to reach 1 for the largest negative eigenvalue. The curve plateaus around zero by enforcing the derivative of the f function to be null at $\lambda = 0$. The conditions for this second scenario are given by

$$f'(0) = 0 \quad \text{and} \quad f(\lambda_1) = -1, \tag{3.31}$$

which gives the weights

$$w_1 = \frac{\sqrt{2}\,|\delta|}{|\lambda_1|} \quad , \quad w_2 = -w_1. \tag{3.32}$$

Here, the number of iterations required to reach a reduction factor of $\epsilon = 1/3$ is

$$\nu \approx \frac{-\log \epsilon}{2}\left(\frac{k^4h^2}{\pi^2}\right)^2 = \mathcal{O}\left(k^2\right). \tag{3.33}$$

Equation (3.33) highlights how expensive the two-step Jacobi smoother can be in that scenario. In fact, the number of iterations required to damp the largest eigenvector as fast as for a Poisson Equation depends on the square of the wavenumber k. Figure 3.5b) plots how slow the damping of the largest eigenvector is, despite the convergence of the method. This feature is especially problematic because the coarse correction is usually built on the uniform interpolation operator (3.11) that tracks slowly varying modes associated with large negative eigenvalues. Authors in [34] recommend using Chebyshev polynomial or Krylov methods instead. For instance, authors use the latter on intermediate levels of the multigrid hierarchy instead as the problem gets very indefinite.

In the final case where $\delta \leq -2$ (i.e., $\lambda_n < 0$), the problem is negative definite. Therefore, the choice of both weights are aimed at maximizing the damping of eventually near-zero oscillatory modes. Good conditions are given by

$$f(\lambda_1) = 1 \qquad \text{and} \qquad f\left(\frac{\lambda_1}{2}\right) = -f(\lambda_n). \tag{3.34}$$

These conditions lead to the weights

$$w_1 = \frac{2\left(2+\sqrt{2}\right)\delta}{\lambda_1 + \left(2+2\sqrt{2}\right)\lambda_n} \quad , \quad w_2 = \frac{2\left(2-\sqrt{2}\right)\delta}{\lambda_1 + \left(2-2\sqrt{2}\right)\lambda_n}. \tag{3.35}$$

Hence, the required number of iterations for damping the most oscillatory mode is

$$\nu \geq \frac{\log \epsilon}{\log\left(8-k^2h^2\right)^2 / \left(-64+16k^2h^2+k^4h^4\right)} \tag{3.36}$$

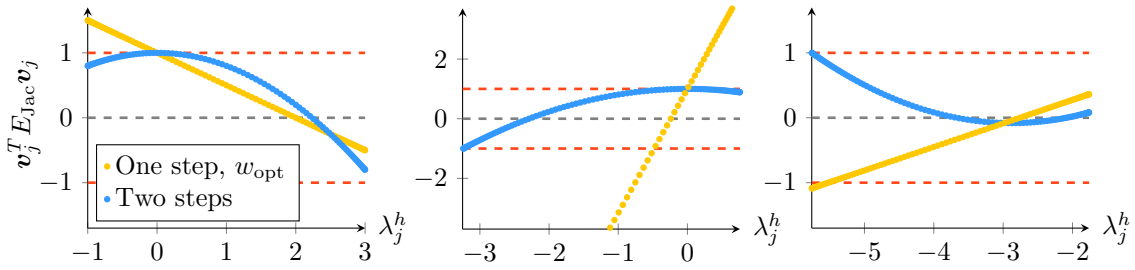
Again, the damping rate of the two-step Jacobi variant is illustrated in Figure 3.5c, where the large eigenvector appears to be damped efficiently. If the discretization coefficient is large enough, the two-step method can be replaced by a classical one-step Jacobi method with weight

$$w = \frac{2-k^2h^2}{3-k^2h^2-2\sin\left(\frac{\pi h}{2}\right)}. \tag{3.37}$$

The number of iterations to be performed in the single step alternative becomes

$$\nu \geq \frac{-\log \epsilon}{\log\left(k^2h^2-3\right)}, \tag{3.38}$$

which converges to one as the discretization coefficient increases.



(a) $kh = 1$ (b) $kh = 1.8$ (c) $kh = 2.4$

Figure 3.5: Damping factors of the Jacobi method and its two-step variant with respect to kh.

3.1.3 Wave-Ray Multigrid & Multiple Coarse Corrections

The first Wave-Ray method for solving Helmholtz is presented in [8] and mainly relies on geometric multigrid ideas. This method has been applied to the First Order System Least-Squares formulation of the Helmholtz equation [56], and later extended with algebraic multigrid ideas [60, 58]. The summary that follows works on the two-dimensional model problem (1.1), as only two wave-propagation directions compose a one-dimensional solution. For more details related to this summary of wave-ray methods, we refer to the simpler implementation proposed in [59].

As mentioned in the previous section, the goal of multigrid is to project difficult eigenvectors associated with small eigenvalues onto an appropriate coarse space. Assuming only these difficult small eigenvectors remain after a few smoothing iterations, the remaining error e_h on the finest level satisfies

$$A\boldsymbol{e}_h \approx 0. \tag{3.39}$$

We remark that the oscillatory propagative waves denoted by $\hat{\boldsymbol{v}}_\theta := \boldsymbol{e}^{\iota k \boldsymbol{S}_\theta(\boldsymbol{x},\boldsymbol{y})}$ are solutions to the homogeneous Helmholtz equation

$$-\Delta\hat{\boldsymbol{v}}_\theta + k^2\hat{\boldsymbol{v}}_\theta = -\Delta\boldsymbol{e}^{\iota k \boldsymbol{S}_\theta(\boldsymbol{x},\boldsymbol{y})} + k^2\boldsymbol{e}^{\iota k \boldsymbol{S}_\theta(\boldsymbol{x},\boldsymbol{y})} = 0, \tag{3.40}$$

with $S_\theta(x, y) = x\cos\theta + y\sin\theta$ and for $\theta \in [0, 2\pi]$. Note that we express $\hat{\boldsymbol{v}}_\theta$ in two dimensions in order to emphasize the infinite number of propagative plain waves that solves the homogeneous equation, whereas the one-dimensional case is characterized by two directions only. The three-dimensional case would require adding the azimuthal angle in the definition of $\hat{\boldsymbol{v}}_\theta$. From both equations (3.39) and (3.40), it comes naturally to write the remaining error as a combination of propagative plain waves, such that

$$\boldsymbol{e}_h = \int_0^{2\pi} \boldsymbol{a}_\theta \hat{\boldsymbol{v}}_\theta d\theta \approx \sum_{j=1}^{n_\theta} \boldsymbol{a}_{\theta_j} \hat{\boldsymbol{v}}_{\theta_j}, \tag{3.41}$$

where n_θ is a positive integer, and $\boldsymbol{a}_{\theta_j}$ corresponds to a smooth envelope function called a "ray". As in the right member of (3.41), each ray is associated with a propagative plain wave $\hat{\boldsymbol{v}}_{\theta_j}$. Applying the homogeneous Helmholtz differential operator to one wave-ray couple gives the equality

$$-\Delta\boldsymbol{a}_{\theta_j}\hat{\boldsymbol{v}}_{\theta_j} + k^2\boldsymbol{a}_{\theta_j}\hat{\boldsymbol{v}}_{\theta_j} = L_{\theta_j}\boldsymbol{a}_{\theta_j}\hat{\boldsymbol{v}}_{\theta_j}, \tag{3.42}$$

where L_{θ_j} is the following Helmholtz differential operator

$$L_{\theta_j} := -\Delta + 2\iota k\left(\sin\theta_{\theta_j}\partial_x - \cos\theta_{\theta_j}\partial_y\right). \tag{3.43}$$

Let $L^h_{\theta_j}$ be a discretization matrix of (3.43). From Equation (3.41), the residual can therefore be approximated by

$$\boldsymbol{r}_h = A\boldsymbol{e}_h \approx \sum_{j=1}^{n_\theta} L^h_{\theta_j}\boldsymbol{a}_{\theta_j}\hat{\boldsymbol{v}}_{\theta_j}. \tag{3.44}$$

The key idea of the Wave-Ray method is to compute each ray $\boldsymbol{a}_{\theta_j}$ independently by projecting them onto a dedicated coarse space. Contrary to the remaining error $\boldsymbol{e}_h$, rays are more likely to be geometrically smooth, and therefore much easier to project onto a coarse space. To summarize, a Wave-Ray cycle starts as a classical multilevel cycle until reaching the separation level indexed by l_s. Multiple ray residuals are then extracted from the residual of the separation level, and treated separately on independent coarse grid hierarchies. The latter step is therefore called Multiple Coarse Corrections.

Letting $\boldsymbol{r}^l$ be the residual of the lth level, the first step is to restrict the residual of the finest level onto a coarser separation level l_s. The restriction is generally performed through an average restriction operator denoted by P_l^{l+1} as in (3.14). Such a restriction operator discards the oscillatory modes, and therefore exhibits the geometrical smoothness on coarse levels. Omitting intermediate smoothing iterations, the restriction of the residual to the separation level is made by applying

$$\boldsymbol{r}^{l_s} = P_{l_s-1}^{l_s}\cdots P_l^{l+1}\cdots P_0^1\boldsymbol{r}^0. \tag{3.45}$$

Note that, until the separation level l_s, coarse matrices result from discretizations of the initial differential operator, including boundary conditions. On the separation level l_s, the second step is to split the residual into n_θ separate ray residuals by introducing proper phase-shifts, such that

$$\begin{aligned}\boldsymbol{r}_{\theta_i}^{l_s} &:= \boldsymbol{r}^{l_s}\boldsymbol{e}^{-\iota\boldsymbol{k}\boldsymbol{S}_{\boldsymbol{\theta_i}}(\boldsymbol{x},\boldsymbol{y})} = \sum_{j=1}^{n_\theta} L_j^{l_s}\boldsymbol{a}_j^{l_s}\hat{\boldsymbol{v}}_j^{l_s}\boldsymbol{e}^{-\iota\boldsymbol{k}\boldsymbol{S}_{\boldsymbol{\theta_i}}(\boldsymbol{x},\boldsymbol{y})} \\ &= L_{\theta_i}^{l_s}\boldsymbol{a}_{\theta_i}^{l_s} + \sum_{j\neq i}^{n_\theta} L_{\theta_j}^{l_s}\boldsymbol{a}_{\theta_j}^{l_s}\hat{\boldsymbol{v}}_{\theta_j}^{l_s}\boldsymbol{e}^{-\iota\boldsymbol{k}\boldsymbol{S}_{\boldsymbol{\theta_i}}(\boldsymbol{x},\boldsymbol{y})} \\ &= \hat{\boldsymbol{r}}_{\theta_i}^{l_s} + \sum_{j\neq i}^{n_\theta} \hat{\boldsymbol{r}}_{\theta_j}^{l_s}\boldsymbol{e}^{-\iota\boldsymbol{k}\left(\boldsymbol{S}_{\boldsymbol{\theta_i}}(\boldsymbol{x},\boldsymbol{y})-\boldsymbol{S}_{\boldsymbol{\theta_j}}(\boldsymbol{x},\boldsymbol{y})\right)}. \end{aligned} \tag{3.46}$$

From both equations (3.44) and (3.46), we see that the residual $\hat{\boldsymbol{r}}_{\theta_i}^{l_s}$ is the right-hand side of the linear system

$$L_{\theta_i}^{l_s}\boldsymbol{a}_{\theta_i}^{l_s} = \hat{\boldsymbol{r}}_{\theta_i}^{l_s}, \tag{3.47}$$

where $\boldsymbol{a}_{\theta_i}^{l_s}$ is the geometrically smooth ray envelope coupled with the propagative plain wave $\boldsymbol{v}_{\theta_i}^{l_s}$. This time, coarse matrices $L_{\theta_i}^{l}$ with $l \geq l_s$ result from independent coarse discretizations of the differential operator (3.43). The sets of coarse points are selected to follow the direction of propagation of the plane wave $\boldsymbol{v}_{\theta_i}^{l_s}$. Examples of coarse variable selection are given in [59]. Also, we note that because of the phase shift, the right member sum is more oscillatory, and subsequently reduced by the next average restriction operations that target geometrical smoothness. Then, phase-shifted residuals $\boldsymbol{r}_{\theta_i}^{l_s}$ are coarsened repeatedly until reaching a small enough level space l_c, such that

$$\boldsymbol{r}_{\theta_i}^{l_c} = P_{l_c-1}^{l_c}\cdots P_{l}^{l+1}\cdots P_{l_s}^{l_s+1}\boldsymbol{r}_{\theta_i}^{l_s}. \tag{3.48}$$

Rays are approximated at the coarsest level by solving the n_θ coarse systems

$$L_{\theta_i}^{l_c}\hat{\boldsymbol{a}}_{\theta_i}^{l_c} = \boldsymbol{r}_{\theta_i}^{l_c}. \tag{3.49}$$

The solution $\hat{\boldsymbol{a}}_{\theta_i}^{l_c}$ of the coarse system (3.48) is an approximation of $\boldsymbol{a}_{\theta_i}^{l_c}$ because the restriction of the residual $\boldsymbol{r}_{\theta_i}^{l_s}$ in (3.48) still contains pollution related to the right member of (3.46). The third step is to interpolate ray approximations up to the separation level l_s,

$$\hat{\boldsymbol{a}}_{\theta_i}^{l_s} := P_{l_s+1}^{l_s}\cdots P_{l+1}^{l}\cdots P_{l_c}^{l_c-1}\hat{\boldsymbol{a}}_{\theta_i}^{l_c}. \tag{3.50}$$

Hence, a good condition of convergence is

$$\boldsymbol{a}_{\theta_i}^{l_s} \approx \hat{\boldsymbol{a}}_{\theta_i}^{l_s}. \tag{3.51}$$

The approximation at the separation level l_s is constructed by merging wave-ray approximation couples as follows

$$\tilde{\boldsymbol{e}}^{l_s} = \sum_{j=1}^{n_\theta} \hat{\boldsymbol{a}}_{\theta_j}^{l_s}\boldsymbol{v}_{\theta_j}. \tag{3.52}$$

Naturally, the approximation of the level l_s is then interpolated back to the finest level such that

$$\tilde{\boldsymbol{e}}^0 = P_1^0 \cdots P_{l+1}^l \cdots P_{l_s}^{l_s-1} \tilde{\boldsymbol{e}}^{l_s}. \tag{3.53}$$

This approach works as a standalone solver and enables a suitable coarse grid correction for Helmholtz. The numerical experiments presented in [59] demonstrate good performances for constant wavenumbers and fine mesh discretization. While the presented approach is built on geometrical multigrid ideas, a multiple Galerkin coarse grid correction method can be found in [58]. One downside is however that it remains challenging to adapt the method to varying wavenumber, and the number of independent coarse corrections per cycle may increase as the matrix becomes more indefinite.

3.1.4 Complex Shifted Laplacian

The indefinite Helmholtz equation is not only challenging for multigrid, but difficult by itself. Smaller wavelength in the continuous problem requires a finer mesh discretization, so that the resulting linear system becomes very large as the wavenumber k increases. For this reason, iterative methods such as GMRES (see Section 2.1.3) are preferred over direct ones when solving that kind of problem as they scale better on modern supercomputers. However, it is well known that their convergence speed depends on the condition number of the initial matrix. Helmholtz matrices may be ill-conditioned depending on the discretization coefficient kh, which tends to decrease the convergence speed. The latter often justifies the use of a preconditioner in the iteration of the method to enhance the convergence, and the speed-up depends on the condition number of the preconditioned matrix. Hence, finding a practical preconditioner that significantly decreases the number of iterations is generally problem-dependent and opened a vast field of study in numerical linear algebra.

In particular for Helmholtz, most classical preconditioners such as the incomplete LU factorization do not provide a substantial speed-up in the convergence of Krylov solvers. This concern motivated the design of a new class of preconditioners called "shifted Laplacian preconditioners" specially dedicated to solving Helmholtz. Helmholtz matrices can be decomposed as follows

$$A = L - k^2 I + B, \tag{3.54}$$

where L denotes the Laplacian matrix, I the identity, and B the boundary condition matrix. This new class of preconditioners for Helmholtz originates from the 1980s [5] and was aimed at enhancing the conjugate gradient method on normal equations (CGNR) by approximating the solution to the Laplacian system $L\boldsymbol{x} = \boldsymbol{b}$ by one SSOR [82] iteration. While the convergence rate of CGNR without preconditioning follows a $\mathcal{O}(h^2)$ law, Bayliss et al. stated that the convergence rate is $\mathcal{O}(1)$ for small mesh size h when solving the Laplacian system exactly. Approximating the solution by one SSOR iteration naturally yields a convergence rate in between both extrema.

In the same paper, the authors indicated that solving the Laplacian system with multigrid enables an h-independent convergence rate. This idea of introducing multigrid cycles to accelerate CGNR has further been developed a year later in [43], and

demonstrated dramatic improvements on the convergence. In the 2000s, Laird and Giles pushed the idea further by adding a positive real shift to the Laplacian [54], giving the first shifted Laplacian preconditioner. The first complex shifted Laplacian (CSL) was finally introduced in [31], and compared with both previous Laplacian and real shifted Laplacian preconditioners.

Let $H_{\gamma,\beta}$ be the CSL preconditioner defined as follows

$$H_{\gamma,\beta} = L + (\gamma + \iota\beta)\, k^2 I + B \quad \text{with} \quad \gamma \geq 0 \quad \text{and} \quad \beta \in \mathbb{R}. \tag{3.55}$$

The matrix $H_{\gamma,\beta}$ is similar to the initial matrix as decomposed in (3.54), except that the wavenumber has a positive real part and an imaginary part.

3.1.4.1 Exact inversion on the simple one-dimensional model problem

Let us tackle the one-dimensional problem (3.1). We set $B = 0$ to satisfy the Dirichlet boundary conditions. This section relies on the analysis developed in [31]. Accordingly, we assume that the CSL preconditioner is inverted exactly to better demonstrate its best case scenario effect on the initial system. We address its approximation by multigrid further in this section. In our case, the iterative method should converge to the solution to the left preconditioned linear system

$$H_{\gamma,\beta}^{-1} A\boldsymbol{x} = H_{\gamma,\beta}^{-1}\boldsymbol{b}. \tag{3.56}$$

This assumption on the explicit inversion of the preconditioner enables a discussion on the best case scenario and will help a further comparison with the multigrid cycle approximation. As defined previously, let $s_j := \sin(\frac{j\pi h}{2})$. Here, both A and $H_{\gamma,\beta}$ have the same eigenvectors, but the latter has eigenvalues

$$\lambda_j\left(H_{\gamma,\beta}\right) = \frac{4}{h^2} s_j^2 + (\gamma + \iota\beta)\, k^2. \tag{3.57}$$

Naturally, it follows that the eigenvalues of the left-preconditioned matrix (3.56) are

$$\lambda_j\left(H_{\gamma,\beta}^{-1} A\right) = \frac{s_j^2 - \left(\frac{kh}{2}\right)^2}{s_j^2 + (\gamma + \iota\beta)\left(\frac{kh}{2}\right)^2}. \tag{3.58}$$

To accelerate the convergence of Krylov iterations, the condition number of the preconditioned system (3.56) should be minimized with respect to the coefficients γ and β defined in (3.56). Since the initial matrix is indefinite, we compute the condition number of the squared left-preconditioned matrix as follows

$$\kappa^2 := \kappa\left(\left(H_{\gamma,\beta}^{-1}A\right)^*\left(H_{\gamma,\beta}^{-1}A\right)\right) = \frac{\lambda_{\max}^h\left(\left(H_{\gamma,\beta}^{-1}A\right)^*\left(H_{\gamma,\beta}^{-1}A\right)\right)}{\lambda_{\min}^h\left(\left(H_{\gamma,\beta}^{-1}A\right)^*\left(H_{\gamma,\beta}^{-1}A\right)\right)}. \tag{3.59}$$

Define $\epsilon := \min_j\left(s_j^2 - \left(\frac{kh}{2}\right)^2\right)$. The condition number (3.59) can be derived by

$$\kappa^2 = \begin{cases} \frac{1}{4}\left(1 + \frac{2\gamma}{\gamma^2+\beta^2}\right)\left((kh)^2/(2\epsilon)\right)^2 & \text{if} \quad \gamma^2 + \beta^2 \leq 1, \\ \frac{1}{4}\left((1+\gamma)^2 + \beta^2\right)\left((kh)^2/(2\epsilon)\right)^2, & \text{if} \quad \gamma^2 + \beta^2 \geq 1 \end{cases}. \tag{3.60}$$

More details on the different steps that led to (3.60) can be found in Appendix A.4 or in [31] as well. One can show that the condition number (3.60) is minimal when $\gamma^2 + \beta^2 = 1$. Hence, the best preconditioner in the real shifted case is reached for $\gamma = 1$. In the complex case with condition $\gamma^2 + \beta^2 = 1$, κ^2 is minimal if $\gamma = 0$, which implies that $\beta = 1$. Let us now compare the three preconditioners $H_0 := H_{0,0}$, $H_1 := H_{1,0}$ and $H_\iota := H_{0,1}$. For ease of notation, let us define the eigenvalues of the one-dimensional Laplacian matrix by

$$\mu_j := 4s_j^2/h^2, \tag{3.61}$$

such that the squared initial matrix eigenvalues are defined by

$$\lambda_j\left(A^*A\right) = \left(\mu_j - k^2\right)^2. \tag{3.62}$$

Using the plain Laplacian preconditioner gives

$$\lambda_j\left(\left(H_0^{-1}A\right)^*\left(H_0^{-1}A\right)\right) = \left(1 - \frac{k^2}{\mu_j}\right)^2. \tag{3.63}$$

In the same way, eigenvalues of the real shifted Laplacian preconditioned matrix are

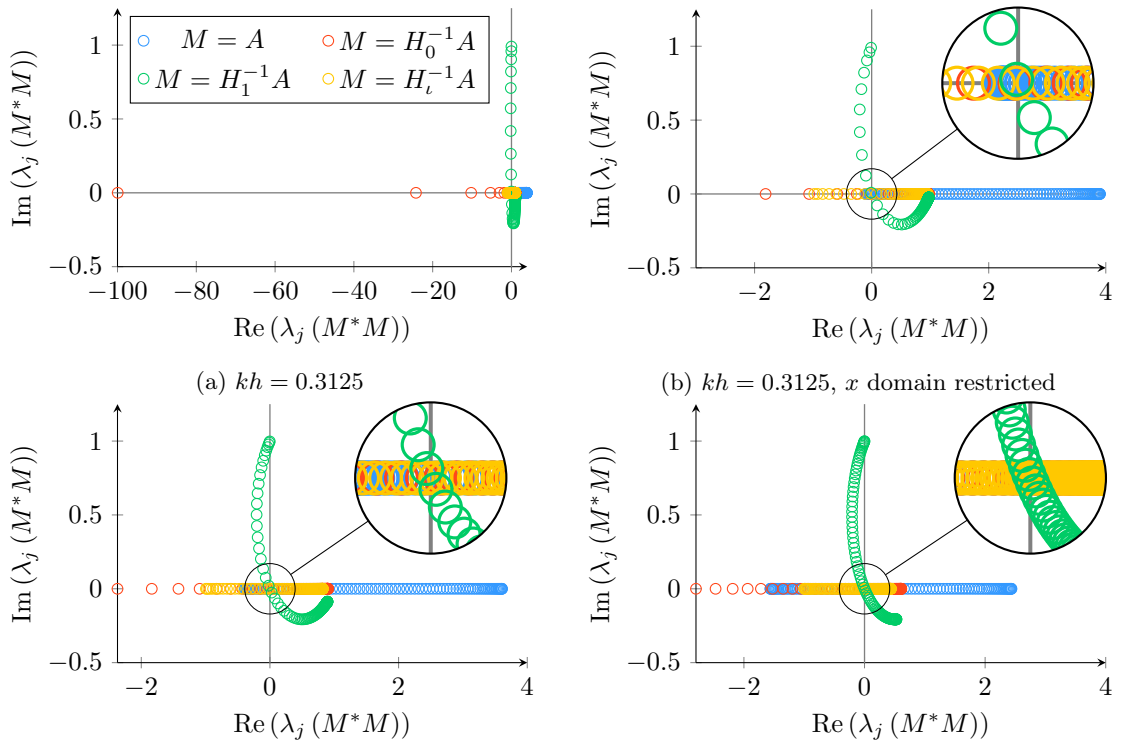
$$\lambda_j\left(\left(H_1^{-1}A\right)^*\left(H_1^{-1}A\right)\right) = \left(1 - \frac{2k^2}{\mu_j + k^2}\right)^2, \tag{3.64}$$

whereas eigenvalues resulting from the complex shifted preconditioner are

$$\lambda_j\left(\left(H_\iota^{-1}A\right)^*\left(H_\iota^{-1}A\right)\right) = 1 - \frac{2\mu_j k^2}{\mu_j^2 + k^4}. \tag{3.65}$$

The Figure 3.6 represents the four spectra of matrices provided by (3.62) (in blue), (3.63) (in red), (3.64) (in green) and (3.65) (in orange) for different values of the discretization coefficient kh. The first figure 3.6a plots the four spectra with no restriction on the x domain, whereas figures 3.6b, 3.6c and 3.6d represent them within a narrower interval to better distinguish their respective behavior around the origin.

Figure 3.6a reveals that the plain Laplacian preconditioner presents the highest eigenvalues in magnitude (see around $x = -100$ or $x = 20$ for instance), which can lead to a large condition number. By contrast, eigenvalues of both shifted Laplacian preconditioned matrices represented by orange and green marks are contained within a bounded interval. Among both types of shifted preconditioners, it appears that the complex shifted preconditioner offers the sparsest concentration of eigenvalues around the origin, which decreases the condition number and therefore speeds up the convergence. The plain Laplacian has the potential to provide a significant help in the limited case where the shift is lower than the smallest eigenvalue (i.e., $0 < k^2 < \mu_1$). Beyond that very particular case, the complex shifted Laplacian generally provides a twice smaller condition number and therefore represents the best option among the three preconditioners. More details in the comparison of condition numbers can be found in Appendix A.4.



(a) $kh = 0.3125$

(b) $kh = 0.3125$, x domain restricted

(c) $kh = 0.625$, x domain restricted

(d) $kh = 1.25$, x domain restricted

Figure 3.6: Spectrum of the preconditioned matrix for different complex shifted Laplacian preconditioners

3.1.4.2 Optimizing the CSL and resolution by Multigrid

The previous development was limited to the one-dimensional model problem for ease of discussion, and also assumed that the inverse of the CSL was known exactly to demonstrate its effectiveness in the best case. Approximating the inverse by an incomplete-LU factorization has been tried in [32] in comparison with geometric multigrid iterations. In [30], the complex shifted Laplacian is approximated using geometric multigrid iterations to speed-up Krylov methods such as BiCGSTAB or GMRES for solving a two-dimensional heterogeneous Helmholtz problem. Algebraic multigrid has later been implemented as a replacement of the geometric setting in [2, 6].

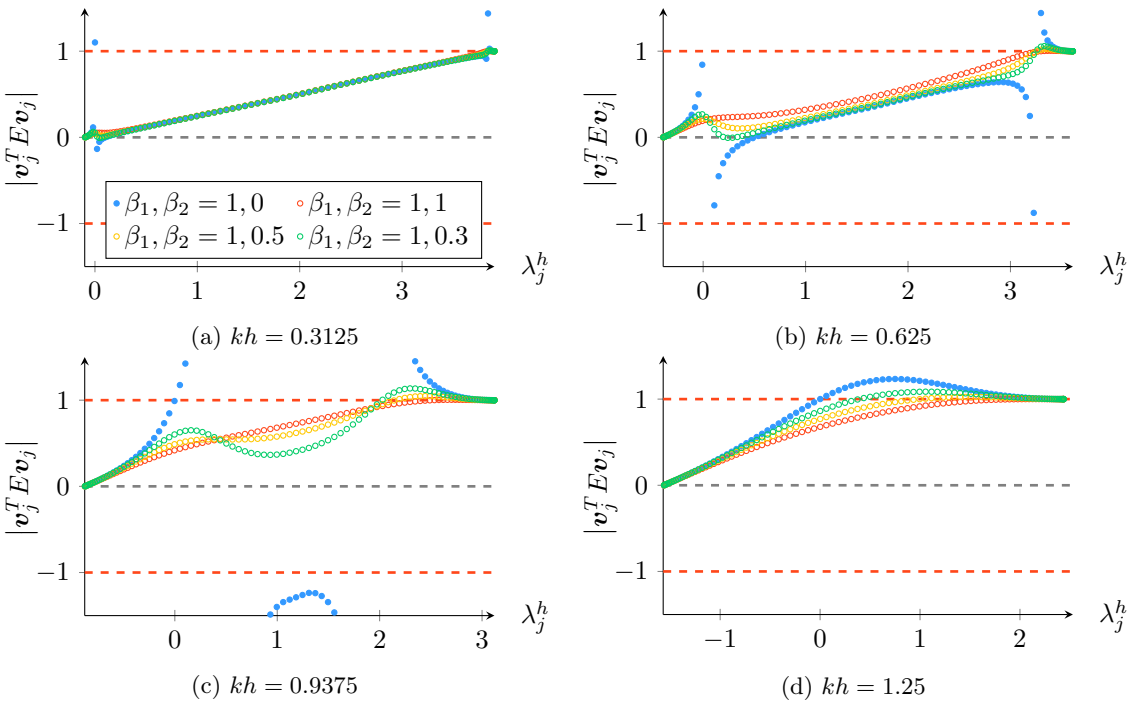
Building more sophisticated multigrid cycles enhances the convergence of these methods, but they are still impacted by the small eigenvalues of the preconditioned system that tend toward zero as the wavenumber increases. Consequently, the condition number increases with the wavenumber, which results in a linearly growing number of iterations. To speed up the convergence of iterative methods when solving Helmholtz, the preconditioner should remain as close as possible from the initial matrix while being easy to solve, with multigrid for instance. The complex shifted preconditioner has been generalized in [30] as follows

$$H_{\beta_1,\beta_2} := L - (\beta_1 - \iota\beta_2)k^2 I\ ,\ (\beta_1, \beta_2) \in \mathbb{R}^2. \tag{3.66}$$

This generalized form now allows the shift to be negative in order to more closely match with the initial indefinite problem. In the most recent development of the method, the shifted preconditioner in fact nearly always include the original negative wavenumber (i.e., $\beta_1 = 1$). Nevertheless, the question of choosing an optimal complex shift β_2 remains central in the method. Letting $H^h_{\beta_1,\beta_2}$ and $H^H_{\beta_1,\beta_2}$ be the fine and coarse discretization of the complex shifted Laplacian respectively, the one-dimensional two-grid analysis applied to the operator (3.66) gives the ratio

$$\frac{\lambda_j\left(H^h_{\beta_1,\beta_2}\right)}{\lambda_j\left(H^H_{\beta_1,\beta_2}\right)} = 1 + \frac{s_j^4}{s_j^2 c_j^2 - \left(\frac{kh}{2}\right)^2 (\beta_1 - \iota\beta_2)}. \tag{3.67}$$

Naturally, the ratio (3.67) of eigenvalues has the same form as in (3.21). Hence, Figure 3.7 shows how the multigrid coarse correction behaves with respect to the eigenvector and depending on the value of β_2. These figures are plotted over the blue curves of Figure 3.4 that pictures the coarse correction without CSL preconditioner.



(a) $kh = 0.3125$ (b) $kh = 0.625$

(c) $kh = 0.9375$ (d) $kh = 1.25$

Figure 3.7: Contraction rate of the coarse correction with respect to kh and β_2 of the complex shifted Laplacian preconditioner.

For all cases, β_1 is set to 1 to match the initial problem, and only β_2 varies. The blue curve corresponds to the initial matrix, whereas the three others correspond to complex shifted matrices. The green curve is associated with the smallest β_2 among the three preconditioners, and seems attracted by the amplified eigenvalues of the initial problem. While a smaller shift matches the initial problem better, large shifted problems are easier to solve. For instance, choosing $\beta_2 = 0$ yields a preconditioner equals to the initial matrix A. Such a preconditioner is therefore the best theoretical preconditioner for Helmholtz. Alternatively, increasing β_2 separates

the preconditioner from the initial problem but enables a practical approximation of the inverse with multigrid. Hence, a trade-off has to be found between both extrema.

This concern has been discussed in [42] and split into two questions. The first asks "*what is the largest shift for which wavenumber-independent convergence is guaranteed?*" [42]. The second asks "*how large a shift is needed for its effective inversion by multigrid?*" [21]. While the former suggests that $\beta_2 \sim \mathcal{O}(k^{-1})$, the answer for the latter is $\beta_2 \sim \mathcal{O}(1)$. Therefore, finding a shift that addresses both questions is particularly difficult. This issue motivated the community to boost the convergence of Krylov methods by "deflating" the small and difficult eigenvalues in addition to the approximation of the CSL inverse by multigrid [67, 68, 67].

These deflation methods have proven to accelerate the overall convergence of the preconditioned Krylov methods for Helmholtz, but the convergence still suffers for large wavenumbers. The authors [26] show that small eigenvalues get closer to zero when using the classical interpolation operators due to a misalignment between fine and coarse grid discretizations. They address this issue by implementing a higher-order interpolation operator based on Bézier curves that is defined as follows

$$\left[P_H^h \boldsymbol{e}^H\right]_i := \begin{cases} \frac{1}{8}\left(\boldsymbol{e}^H_{(i-1)/2} + 6\boldsymbol{e}^H_{(i)/2} + \boldsymbol{e}^H_{(i+1)/2}\right) & \text{if } i \text{ even}, \\ \frac{1}{2}\left(\boldsymbol{e}^H_{(i-1)/2} + \boldsymbol{e}^H_{(i+1)/2}\right) & \text{if } i \text{ odd}, \end{cases} \qquad 1 \leq i \leq n. \quad (3.68)$$

To the best of our knowledge, the first wavenumber-independent multigrid convergence appeared from this setting. The two-grid cycle has later been extended to a multilevel method in [27]. One other substantial improvement from [27] is that the complex shift can get decreased up to $\beta_2 = k^{-1}$ by replacing the unstable usual multigrid smoothers that tends to amplify the negative eigenvalues by a few GMRES iterations. A year later, the first standalone multigrid method [28] emerged. While numerical experiments do not suggest a wavenumber-independent convergence yet, they demonstrate the potential of multigrid in solving the Helmholtz problem when combining the complex shifted Laplacian preconditioner, higher-order interpolation scheme, and GMRES smoothing iterations. The authors also developed the convergence theory of two-grid methods, based on the positive-definiteness of the approximate inverse formed by the coarse correction plus the post-smoothing operation.

3.2 Corruption of the coarse correction in the indefinite case

Now that we have summarized past research on solving the Helmholtz equation with multigrid, it is time to introduce our concept of "pollution" to emphasize why the classical coarse correction appears hopeless for indefinite problems. In fact, we will see that this pollution can corrupt the coarse correction and consequently lead to divergence, even though the interpolation operator P has good approximation properties. In particular, we will see that the contraction or amplification of an eigenvector after applying the coarse correction depends on a mix between small and large eigenvalues, and that the combination of eigenvalues depends on what we call a "block of pollution". The next developments highlight that the pollution

arising from the largest eigenvectors has a strong impact on the coarse correction, especially for small eigenvectors.

3.2.1 Introduction to the concept of pollution

In Chapter 5.1, we show that our interpolation operator should have good approximation properties for the set of smallest eigenvectors denoted by V_c (see (5.6)). The pollution is what makes the difference from the actual target space V_c and its best representation provided by the interpolation operator P. The following theorem links the interpolation error of a small eigenvector $\boldsymbol{v}_i$ of V_c with what we call a "pollution block" denoted by K_f. For what follows, we denote the l_2-orthogonal projection by

$$\Pi\left(P\right) := P\left(P^TP\right)^{-1}P^T. \tag{3.69}$$

Theorem 1. *Let A be an $n \times n$ matrix, and V its orthonormal set of eigenvectors, each associated with the corresponding element of the diagonal eigenvalue matrix Λ. Also, let P be an $n \times n_c$ interpolation operator. Assuming V_c^TP is non-singular, we write the linear decomposition of the post-scaled interpolation operator as $P\left(V_c^TP\right)^{-1} = VK$, where K is the following $n \times n_c$ matrix of coefficients*

$$K := V^TP(V_c^TP)^{-1} = \begin{bmatrix} I_c \\ K_f \end{bmatrix}. \tag{3.70}$$

The block I_c corresponds to the identity matrix of size $n_c \times n_c$, and the block K_f is a $n_f \times n_c$ matrix such that $K_f := V_f^TP(V_c^TP)^{-1}$. The interpolation error of the eigenvector $\boldsymbol{v}_i$ of V_c is given by

$$\boldsymbol{v}_i^T\left(I - \Pi(P)\right)\boldsymbol{v}_i = 1 - \left[\left(I_c + K_f^TK_f\right)^{-1}\right]_{i,i}, \tag{3.71}$$

where $[\cdot]_{j,k}$ denotes the entry (j,k) of the bracketed matrix.

Proof. First, note that post-multiplying P by any non-singular matrix M_c of size $n_c \times n_c$ does not change the l_2-projection

$$\begin{aligned}(PM_c)((PM_c)^T(PM_c))^{-1}(PM_c)^T &= PM_cM_c^{-1}(P^TP)^{-1}M_c^{-T}M_c^TP^T \\ &= P(P^TP)^{-1}P^T = \Pi\left(P\right).\end{aligned} \tag{3.72}$$

In particular for $M_c = (V_c^TP)^{-1}$,

$$\begin{aligned}I - \Pi\left(P\right) &= I - P(P^TP)^{-1}P^T \\ &= I - P(V_c^TP)^{-1}\left(P(V_c^TP)^{-1})^TP(V_c^TP)^{-1}\right)^{-1}(P(V_c^TP)^{-1})^T.\end{aligned} \tag{3.73}$$

Since $P(V_c^TP)^{-1} = VK$, it follows that

$$\begin{aligned}I - \Pi\left(P\right) &= I - (VK)((VK)^T(VK))^{-1}(VK)^T \\ &= I - VK(K^TK)^{-1}K^TV^T.\end{aligned} \tag{3.74}$$

For any eigenvector $\boldsymbol{v}_i$ of A, let $\boldsymbol{e}_i := V^T\boldsymbol{v}_i$ be the canonical unit vector with a one at the i^{th} position and zero elsewhere. Assuming $\boldsymbol{v}_i \in V_c$ ($i \leq n_c$), the

vector $\boldsymbol{c}_i := K^T \boldsymbol{e}_i$ of size n_c is also a unit vector with a one at the ith position. Consequently, the damping factor of $\boldsymbol{v}_i \in V_c$ is

$$\begin{aligned} \boldsymbol{v}_i^T \left(I - \Pi\left(P\right)\right) \boldsymbol{v}_i &= \boldsymbol{v_i}^T V (I - K(K^T K)^{-1} K^T) V^T \boldsymbol{v_i} \\ &= \boldsymbol{e}_i^T (I - K(K^T K)^{-1} K^T) \boldsymbol{e}_i \\ &= 1 - \boldsymbol{c}_i^T (K^T K)^{-1} \boldsymbol{c}_i = 1 - \left[(I_c + K_f^T K_f)^{-1}\right]_{i,i}. \end{aligned} \tag{3.75}$$

□

Since the l_2-projection is unchanged by the post-multiplication of P, we assume for what follows that K has the form (3.70). The block K_f designates what we call "pollution". This block of pollution causes a small difference between an eigenvector $\boldsymbol{v}_i$ of V_c and its best representation in the range of P. The entry $[K_f]_{j,i}$ designates the contribution of the jth large eigenvector of V_f to the interpolation error of the ith smallest eigenvector of V_c. When the ith column of K_f is null, then the interpolation error of $\boldsymbol{v}_i$ equals zero, such that

$$[K_f]_{:,i} = 0 \qquad \Leftrightarrow \qquad \Big(I - \Pi\left(P\right)\Big) \boldsymbol{v}_i = 0. \tag{3.76}$$

However, in practice, a null column is unlikely to be satisfied for Helmholtz, because P should be sparse for cost considerations and the smallest eigenvectors are usually unknown. Moreover, the near-kernel space of the Helmholtz equation is oscillatory. This makes the construction of good interpolation rules more difficult, and tends to pollute the interpolation range. In fact, Theorem 1 indicates that the error of interpolation is probably unavoidable because the columns of K_f are unlikely to be zero. Illustrations of the pollution block K_f are given throughout Chapter 5 for different interpolation operators.

3.2.2 Corruption of the coarse correction in the indefinite case

At this stage, let us demonstrate how the pollution can corrupt the coarse correction in the indefinite case. Consider the contraction of the n_c small eigenvectors V_c, assuming the n_f large eigenvectors V_f are damped by the smoother.

Theorem 2. *Let the matrix K be defined as in* (3.70). *The contraction of an eigenvector $\boldsymbol{v}_i$ of V_c after the coarse correction is given by*

$$\boldsymbol{v}_i^T E \boldsymbol{v}_i = 1 - \lambda_i \left[\left(\Lambda_c + K_f^T \Lambda_f K_f\right)^{-1}\right]_{i,i}. \tag{3.77}$$

Proof. By the same reasoning of the proof for Theorem 1, we note that post-multiplying P by any non-singular matrix M_c of size $n_c \times n_c$ does not change the coarse correction

$$(PM_c)((PM_c)^T A (PM_c))^{-1} (PM_c)^T = P(P^T A P)^{-1} P^T \tag{3.78}$$

Subsequently for $PM_c = P(V_c^T P)^{-1} = VK$, we have

$$\begin{aligned} \Pi_A\left(P\right) &= P(P^T A P)^{-1} P^T A \\ &= (VK)((VK)^T A (VK))^{-1} (VK)^T A \\ &= VK(K^T \Lambda K)^{-1} K^T \Lambda V^T, \end{aligned} \tag{3.79}$$

and the error propagation matrix of the coarse correction can therefore be written

$$E = V(I - K(K^T \Lambda K)^{-1} K^T \Lambda) V^T. \tag{3.80}$$

Defining the Euclidean basis vectors $\boldsymbol{e}_i$ and $\boldsymbol{c}_i$ as in the proof of Theorem 1, it follows that the contraction of $\boldsymbol{v}_i \in V_c$ is

$$\begin{aligned} \boldsymbol{v}_i^T E \boldsymbol{v}_i &= \boldsymbol{v_i}^T V(I - K(K^T \Lambda K)^{-1} K^T \Lambda) V^T \boldsymbol{v_i} \\ &= \boldsymbol{e}_i^T (I - K(K^T \Lambda K)^{-1} K^T \Lambda) \boldsymbol{e}_i \\ &= 1 - \lambda_i \boldsymbol{c}_i^T (K^T \Lambda K)^{-1} \boldsymbol{c}_i = 1 - \lambda_i \left[\left(\Lambda_c + K_f^T \Lambda_f K_f \right)^{-1} \right]_{i,i}. \end{aligned} \tag{3.81}$$

□

Theorem 2 shows that the effect of the coarse correction relies on a combination of the small eigenvalues Λ_c plus the large eigenvalues Λ_f, such that the mix is given by the entries of the pollution K_f. In the SPD case, the effectiveness of the coarse correction is well known. If all eigenvalues are positive, one can remark that

$$\forall i \leq n_c \ , \ 0 \leq \left[\left(\Lambda_c + K_f^T \Lambda_f K_f \right)^{-1} \right]_{i,i} \leq \left[\Lambda_c^{-1} \right]_{i,i} = \lambda_i^{-1} \Rightarrow 0 \leq \boldsymbol{v}_i^T E \boldsymbol{v}_i \leq 1. \tag{3.82}$$

Therefore, the coarse correction always acts as a contraction on $\boldsymbol{v}_i$ regardless of the block of pollution K_f. However, in the indefinite case, the property (3.82) does not hold. In fact, a necessary condition for the coarse correction to be a contraction is

$$\forall i \leq n_c \ , \ \left| \boldsymbol{v}_i^T E \boldsymbol{v}_i \right| \leq 1 \Rightarrow 0 \leq \lambda_i \left[\left(\Lambda_c + K_f^T \Lambda_f K_f \right)^{-1} \right]_{i,i} \leq 2. \tag{3.83}$$

From (3.83), it follows that each diagonal entry should have the same sign as the associated eigenvalue, and be smaller than twice the inverse of the eigenvalue in magnitude. Nothing guarantees such conditions in the case where the small and large eigenvalues have mixed sign. Especially for very small eigenvalues, the mix can easily lead to a diagonal entry of the opposite sign even though K_f is small, because its entries are weighted by the large eigenvalues Λ_f. Therefore, a good interpolation operator can still cause the coarse correction to amplify the error. For very near-zero eigenvalues, even round-off error can lead to divergence in the indefinite case. The following 2×2 example better illustrates how the pollution can cause divergence when A is indefinite.

Example 2. *Let A be a 2×2 matrix, and $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ its eigenvectors respectively associated with eigenvalues $|\lambda_1| < |\lambda_2|$. Let P be an interpolation operator of size 2×1 targeting the smallest eigenvector $\boldsymbol{v}_1$, such that*

$$P = \boldsymbol{v}_1 + \epsilon \boldsymbol{v}_2. \tag{3.84}$$

From definition (3.70)*, the K matrix can be derived by*

$$K = V^T P \left(\boldsymbol{v}_1^T P \right)^{-1} = \left[\boldsymbol{v}_1, \boldsymbol{v}_2 \right]^T \cdot \left[\boldsymbol{v}_1 + \epsilon \boldsymbol{v}_2 \right] = \begin{bmatrix} 1 \\ \epsilon \end{bmatrix}. \tag{3.85}$$

From Theorem 2, the action of the coarse correction on $\boldsymbol{v}_1$ is given by

$$\boldsymbol{v}_1^T E \boldsymbol{v}_1 = 1 - \lambda_1 \left[\left(\Lambda_c + K_f^T \Lambda_f K_f \right)^{-1} \right]_{1,1} = 1 - \frac{\lambda_1}{\lambda_1 + \epsilon^2 \lambda_2} \tag{3.86}$$

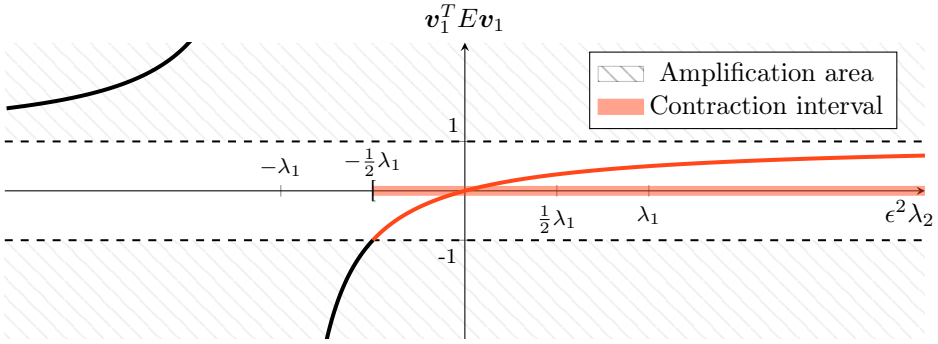


Figure 3.8: Contraction of the coarse correction with respect to the pollution

The Figure 3.8 plots the action of the coarse correction on $\boldsymbol{v}_1$ for $\lambda_1 > 0$ with respect to the pollution block $K_f^T \Lambda_f K_f = \epsilon^2 \lambda_2$. A first observation is that the coarse correction does not amplify the smallest eigenvector if both eigenvalues have the same sign. If both eigenvalues are oppositely signed, then the coarse correction amplifies $\boldsymbol{v}_1$ when $\epsilon^2\lambda_2 < -\lambda_1/2$. Therefore, the condition on the pollution $K_f = \epsilon$ that drives the error of interpolation is particularly difficult to satisfy for a small value of λ_1 and a large value of λ_2.

Figure 3.8 in Example 2 helps understand how improving P affects the coarse correction. Looking at the right segment of the x-axis, $\epsilon^2\lambda_2$ has the same sign as the target eigenvalue λ_1. This scenario is similar to the SPD setting, where improving P by decreasing the pollution coefficient ϵ enhances the contraction of $\boldsymbol{v}_1$ (i.e., $\boldsymbol{v}_1 E \boldsymbol{v}_1$ decreases while remaining positive). The left part of the x-axis corresponds to the counter case where $\epsilon^2\lambda_2$ has the opposite sign of λ_1. Since ϵ gets larger as P worsens, then $\boldsymbol{v}_1 E \boldsymbol{v}_1$ is slightly greater than one when P has terrible approximation properties for $\boldsymbol{v}_1$ and where λ_2 has λ_1 opposite sign. In that case, correcting P potentially brings $\epsilon^2\lambda_2$ within the contraction interval $[-\lambda_1/2\,,\,0]$, but it can also shift the pollution around the critical point where $\epsilon^2\lambda_2 \approx -\lambda_1$. In that particular case, the error explodes and the method diverges extremely fast.

When the matrix gets larger than the previous 2×2 example, the pollution is not defined by a single coefficient ϵ but by the block K_f of (3.70). The same observations still apply for all problem sizes despite this difference of notation, such that decreasing the entries of the pollution block K_f improves the contraction performed by the coarse correction in the SPD case. In that positive context, decreasing the entries of K_f improves the contraction, such that

$$K_f \approx 0 \quad \Rightarrow \quad \left[\left(\Lambda_c + K_f^T \Lambda_f K_f\right)^{-1}\right]_{i,i} \approx \lambda_i^{-1} \quad \Rightarrow \quad \boldsymbol{v}_i^T E \boldsymbol{v}_i \approx 0. \tag{3.87}$$

This feature cannot be generalized to the indefinite case. In fact, good interpolation rules enable the coarse correction to contract the error only if the condition (3.83) is satisfied. Decreasing the entries of K_f still improves the l_2-orthogonal projection, but it does not necessarily have the same effect on the coarse correction. For instance, decreasing the entries of K_f associated with the positive eigenvalues increases the prevalence of negative eigenvalues in the mix $K_f^T \Lambda_f K_f$. In that scenario, the right term of (3.77) in Theorem 2 is more likely to be negative, and the coarse correction prone to amplify eigenvectors associated with positive eigenvalues.

The next theorem derives a more general condition on the spectral radius $\rho(K_f^T \Lambda_f K_f)$ for the coarse correction to be a contraction of the smallest eigenvalues in the indefinite case based on the concept of pollution.

Theorem 3. *If A is indefinite, then*

$$\rho\left(K_f^T \Lambda_f K_f\right) \leq \frac{1}{2}|\lambda_1| \qquad \Rightarrow \qquad \forall \boldsymbol{v}_i \in V_c \ , \ \left|\boldsymbol{v}_i^T E \boldsymbol{v}_i\right| \leq 1 \tag{3.88}$$

Proof. Define $M_K = I_c + \Lambda_c^{-1} K_f^T \Lambda_f K_f$. From the shape of the matrix K defined in (3.80), we have

$$\begin{aligned} V_c^T E V_c &= V_c^T V (I - K(K^T \Lambda K)^{-1} K^T \Lambda) V^T V_c \\ &= I_c - (K^T \Lambda K)^{-1} \Lambda_c \\ &= I_c - (I_c + \Lambda_c^{-1} K_f^T \Lambda_f K_f)^{-1} \Lambda_c^{-1} \Lambda_c \\ &= I_c - M_K^{-1}. \end{aligned} \tag{3.89}$$

Hence, it follows that

$$\forall \boldsymbol{v}_i \in V_c \ , \ \boldsymbol{v}_i^T E \boldsymbol{v}_i = \boldsymbol{e}_i^T V_c^T E V_c \boldsymbol{e}_i = 1 - \boldsymbol{e}_i^T M_K^{-1} \boldsymbol{e}_i. \tag{3.90}$$

where $\boldsymbol{e}_i$ is the ith vector of the Euclidean basis in $\mathbb{R}^{n_c}$. Therefore, $|\boldsymbol{v}_i^T E \boldsymbol{v}_i| \leq 1$ if

$$\forall \boldsymbol{v}_i \in V_c \ , \ -1 \leq \boldsymbol{v}_i^T E \boldsymbol{v}_i \leq 1 \ \Leftrightarrow \ 0 \leq \boldsymbol{e}_i^T M_K^{-1} \boldsymbol{e}_i \leq 2. \tag{3.91}$$

We begin by deriving a condition for the right bound of (3.91), and will show that it also satisfies the left one. Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be two vectors of $\mathbb{R}^n$ linked by the relation $\boldsymbol{x} = M_k \boldsymbol{y}$. The right bound is satisfied if

$$\max_{x \neq 0} \frac{||M_K^{-1} \boldsymbol{x}||}{||\boldsymbol{x}||} = \max_{y \neq 0} \frac{||\boldsymbol{y}||}{||M_K \boldsymbol{y}||} = \left(\min_{y \neq 0} \frac{||M_K \boldsymbol{y}||}{||\boldsymbol{y}||} \right)^{-1} \leq 2. \tag{3.92}$$

Therefore, the condition (3.92) is equivalent to

$$\min_{y \neq 0} \frac{||M_K \boldsymbol{y}||}{||\boldsymbol{y}||} \geq \frac{1}{2}. \tag{3.93}$$

Let $\sigma_i(M)$ be the ith largest singular value of a given matrix M. In a same way, $\lambda_i(M)$ designates the ith largest eigenvalue in magnitude of M (we omit the matrix between parenthesis when referring to the initial matrix A). In addition, let us recall the following triangle inequality $||\boldsymbol{y} + \boldsymbol{z}|| \geq ||\boldsymbol{y}|| - ||\boldsymbol{z}||$, $\forall \boldsymbol{y}, \boldsymbol{z} \in \mathbb{R}^{n_c}$. Thus, we have that

$$\begin{aligned} \min_{y \neq 0} \frac{||M_K \boldsymbol{y}||}{||\boldsymbol{y}||} = \min_{y \neq 0} \frac{||\boldsymbol{y} + \Lambda_c^{-1} K_f^T \Lambda_f K_f \boldsymbol{y}||}{||\boldsymbol{y}||} &\geq \min_{y \neq 0} \left(1 - \frac{||\Lambda_c^{-1} K_f^T \Lambda_f K_f \boldsymbol{y}||}{||\boldsymbol{y}||} \right) \\ &= 1 - \max_{y \neq 0} \frac{||\Lambda_c^{-1} K_f^T \Lambda_f K_f \boldsymbol{y}||}{||\boldsymbol{y}||} \\ &= 1 - \sigma_{n_c}\left(\Lambda_c^{-1} K_f^T \Lambda_f K_f\right). \end{aligned} \tag{3.94}$$

It follows that the condition (3.93) is satisfied if $\sigma_{n_c}(\Lambda_c^{-1}K_f^T\Lambda_f K_f) \leq \frac{1}{2}$. Finally, since $\sigma_{n_c}(\Lambda_c^{-1}K_f^T\Lambda_f K_f) \leq \sigma_{n_c}(K_f^T\Lambda_f K_f)/\sigma_1$ and the singular values coincide with eigenvalues in magnitude because both Λ_c and $K_f^T\Lambda_f K_f$ are Hermitian, the right bound of (3.91) is satisfied if

$$|\lambda_{n_c}\left(K_f^T\Lambda_f K_f\right)| = \rho\left(K_f^T\Lambda_f K_f\right) \leq \frac{1}{2}|\lambda_1|. \tag{3.95}$$

We now address the left bound of (3.91) assuming the condition (3.95) holds. Our goal is to prove that all the diagonal entries of M_K^{-1} are positive. In that way, let $F(M)$ be the field of values of a given matrix M of size n_c such that

$$F\left(M\right) := \left\{\boldsymbol{x}^* M\boldsymbol{x} \mid \forall \boldsymbol{x} \in \mathbb{C}^{n_c}\ ,\ \boldsymbol{x}^*\boldsymbol{x} = 1\right\}. \tag{3.96}$$

If M is Hermitian, one can show that (e.g., [51, chapter 4])

$$\min_{\boldsymbol{x}^*\boldsymbol{x}=1} \boldsymbol{x}^* M\boldsymbol{x} = \lambda_{\min}\left(M\right) \qquad \text{and} \qquad \max_{\boldsymbol{x}^*\boldsymbol{x}=1} \boldsymbol{x}^* M\boldsymbol{x} = \lambda_{\max}\left(M\right). \tag{3.97}$$

Accordingly, let $F(\Lambda_c)$ and $F(K_f^T\Lambda_f K_f)$ be the field of values of Λ_c and $K_f^T\Lambda_f K_f$ respectively. Since A is non-singular, then $0 \notin F\left(\Lambda_c\right)$. Therefore, the spectrum of $\Lambda_c^{-1}K_f^T\Lambda_f K_f$ is included as follows (e.g., [50, chapter 1])

$$\lambda_j\left(\Lambda_c^{-1}K_f^T\Lambda_f K_f\right) \in F\left(K_f^T\Lambda_f K_f\right)/F\left(\Lambda_c\right) \quad , \quad j = 1, \ldots, n_c. \tag{3.98}$$

The set ratio in (3.98) has the usual algebraic interpretation such that

$$\forall \psi \in \frac{F\left(K_f^T\Lambda_f K_f\right)}{F\left(\Lambda_c\right)}\ ,\ -\frac{\max\limits_{\boldsymbol{x}^*\boldsymbol{x}=1}\left|\boldsymbol{x}^* K_f^T\Lambda_f K_f\boldsymbol{x}\right|}{\min\limits_{\boldsymbol{x}^*\boldsymbol{x}=1}\left|\boldsymbol{x}^*\Lambda_c\boldsymbol{x}\right|} \leq \psi \leq \frac{\max\limits_{\boldsymbol{x}^*\boldsymbol{x}=1}\left|\boldsymbol{x}^* K_f^T\Lambda_f K_f\boldsymbol{x}\right|}{\min\limits_{\boldsymbol{x}^*\boldsymbol{x}=1}\left|\boldsymbol{x}^*\Lambda_c\boldsymbol{x}\right|}. \tag{3.99}$$

Furthermore, matrices Λ_c and $K_f^T\Lambda_f K_f$ are Hermitian so the property (3.97) holds for both of them. Because the spectrum belongs to the set ratio as in (3.98), we have

$$-|\lambda_1|^{-1} \cdot \left|\lambda_{n_c}\left(K_f^T\Lambda_f K_f\right)\right| \leq \lambda_j\left(\Lambda_c^{-1}K_f^T\Lambda_f K_f\right) \leq \left|\lambda_{n_c}\left(K_f^T\Lambda_f K_f\right)\right| \cdot |\lambda_1|^{-1}. \tag{3.100}$$

Therefore, assuming the condition (3.95) is satisfied, it follows

$$\lambda_j\left(\Lambda_c^{-1}K_f^T\Lambda_f K_f\right) \geq -\left|\lambda_{n_c}\left(K_f^T\Lambda_f K_f\right)\right| \times |\lambda_1|^{-1} \geq -\frac{1}{2}. \tag{3.101}$$

Adding one to each member of the inequality (3.101) finally gives

$$\lambda_j\left(M_K\right) = \lambda_j\left(I + \Lambda_c^{-1}K_f^T\Lambda_f K_f\right) \geq \frac{1}{2} \tag{3.102}$$

Hence, the condition (3.95) implies that all eigenvalues of M_K are positive. Subsequently, $\det\left(M_K\right) > 0$. The adjugate formula for the inverse of M_K shows that diagonal entries are positive if the determinant of principal sub-matrices are also positive. Denote by $\left[\,\cdot\,\right]_{\Omega_{-i}}$ the principal sub-matrix obtained by deleting the ith row and column of a matrix. Since Λ_c is diagonal, one can show that

$$\left[\Lambda_c^{-1}K_f^T\Lambda_f K_f\right]_{\Omega_{-i}} = \left[\Lambda_c\right]_{\Omega_{-i}}^{-1}\left[K_f^T\Lambda_f K_f\right]_{\Omega_{-i}}. \tag{3.103}$$

As in (3.98), the spectrum is included such that

$$\lambda_j\left([\Lambda_c]^{-1}_{\Omega_{-i}}\left[K_f^T\Lambda_f K_f\right]_{\Omega_{-i}}\right) \in F\left(\left[K_f^T\Lambda_f K_f\right]_{\Omega_{-i}}\right)/F\left([\Lambda_c]_{\Omega_{-i}}\right),\ j=1,\cdots,n_c-1.$$

and therefore the following bound holds

$$\lambda_j\left([\Lambda_c]^{-1}_{\Omega_{-i}}\left[K_f^T\Lambda_f K_f\right]_{\Omega_{-i}}\right) \geq -\left|\lambda_{n_c-1}\left(\left[K_f^T\Lambda_f K_f\right]_{\Omega_{-i}}\right)\right| \times |\lambda_1|^{-1}. \tag{3.104}$$

The matrix $K_f^T\Lambda_f K_f$ being Hermitian, Cauchy's interlace theorem states that

$$\lambda_j\left(K_f^T\Lambda_f K_f\right) \leq \lambda_j\left(\left[K_f^T\Lambda_f K_f\right]_{\Omega_{-i}}\right) \leq \lambda_{j+1}\left(K_f^T\Lambda_f K_f\right),\ j=1,\cdots,n_c-1. \tag{3.105}$$

As a consequence, and from the inequality (3.101), we have

$$\lambda_j\left([\Lambda_c]^{-1}_{\Omega_{-i}}\left[K_f^T\Lambda_f K_f\right]_{\Omega_{-i}}\right) \geq -\left|\lambda_{n_c}\left(K_f^T\Lambda_f K_f\right)\right| \times |\lambda_1|^{-1} \geq -\frac{1}{2}. \tag{3.106}$$

Hence, eigenvalues of principal sub-matrices also satisfy

$$\lambda_j\left([M_K]_{\Omega_{-i}}\right) = \lambda_j\left(I_{n_c-1} + [\Lambda_c]^{-1}_{\Omega_{-i}}\left[K_f^T\Lambda_f K_f\right]_{\Omega_{-i}}\right) \geq \frac{1}{2}. \tag{3.107}$$

Because eigenvalues of the principal sub-matrices are positive, so are the determinants. From the adjugate formula of M_K^{-1}, it follows that

$$\boldsymbol{e}_i^T M_K^{-1}\boldsymbol{e}_i = \left[M_K^{-1}\right]_{i,i} = \frac{\det\left([M_K]_{\Omega_{-i}}\right)}{\det\left(M_K\right)} \geq 0, \qquad i=1,\cdots,n_c \tag{3.108}$$

As a consequence, both left and right bounds of (3.91) are satisfied. Finally,

$$\rho\left(K_f^T\Lambda_f K_f\right) \leq \frac{1}{2}|\lambda_1| \qquad \Rightarrow \qquad \forall \boldsymbol{v}_i \in V_c\ ,\ \left|\boldsymbol{v}_i^T E \boldsymbol{v}_i\right| \leq 1 \tag{3.109}$$

□

The condition provided by Theorem 3 is that the spectral radius of the block $K_f^T\Lambda_f K_f$ should not exceed half of the smallest eigenvalue in magnitude. No assumption can be made on the sign of eigenvalues in the indefinite case, so that the condition prevents the coarse correction from amplifying the error in the case where eigenvalues are oppositely signed. Applied to the previous example 3.8, Theorem 3 states that $|\epsilon^2\lambda_2| < |\lambda_1|/2$. That said, the condition is extremely strict and probably impossible to satisfy in practice for very small eigenvalues. In a practical method, the block K_f will never be sufficiently small for solving all types of indefinite problems because of a potentially near-zero eigenvalue.

Chapter 4

Smoother for Helmholtz

The main attempts in solving Helmholtz with multigrid ideas have been introduced in the previous Chapter 3. By comparison to these methods, our algorithm works purely algebraically and uses standard components of traditional multigrid methods, such as a smoother, an interpolation operator, and a coarse correction. However, each component needs to be adapted to the indefinite and oscillatory nature of the Helmholtz equation. This chapter opens the presentation of our alternative method by discussing the question of a good smoother for Helmholtz.

The Helmholtz equation is characterized by negative eigenvalues that makes the choice of a good smoother difficult. Moreover, traditional smoothers such as Jacobi amplify certain modes. With the aim of developing an algebraic multigrid method for Helmholtz, an alternative smoother has to be designed. Both Kaczmarz and the two-step variant of Jacobi introduced in Chapter 3 are good smoothers that fix the amplification of negative eigenvalues, but often at the cost of a slow convergence when the problem gets more indefinite.

In this chapter, we target a smoother with good convergence properties for eigenvectors associated with large magnitude eigenvalues and independently of their sign. To do so, our smoother will rely on Chebyshev polynomials introduced in Section 2.1.2. In accordance with the complementarity principle, minimizing the overlap between the action of the smoother and of the coarse correction accelerates the convergence of the method. Hence, a smoother whose behavior on the spectrum is a priori known will make the construction of good interpolation rules more convenient. While this feature helps the theoretical understanding of the method, it will also have a practical benefit. In particular, Chapter 5 details how we use the smoother to generate an approximation of the optimal interpolation space. In a similar manner, we will see in Chapter 6 that the smoother can help improve our alternative coarse correction. Another complication is maintaining good smoothing properties in a multilevel setting. Each new coarse matrix is computed by coarsening its fine parent with an interpolation operator that targets small eigenvectors. Therefore, each new coarse matrix in the multilevel hierarchy is more indefinite than its fine parent, until reaching an exact balance between negative and positive eigenvalues. This observation is illustrated in Chapter 7, where Figure 7.6 portrays the eigenvalues of each matrix of a multilevel hierarchy.

Krylov iterations are good polynomial smoothers in the indefinite case but they minimize the global residual norm regardless of the eigenvalues and are right-hand side dependent. Hence, the polynomial changes at each iteration. Even though they can give good convergence in practice, their behavior on the spectrum of the matrix remains difficult to predict. To better motivate our interest in an alternative smoother, Figure 4.1 represents the polynomials extracted from GMRES for two different residuals. Each residual results from a combination of three eigenvectors of the initial matrix. In this example, the size of the Krylov basis is set to 3, as the number of eigenvectors that compose both residuals. We observe that the roots of the polynomials correspond to the associated eigenvalues, such that the polynomial provided by GMRES is the best for one given residual. However, these polynomials are not necessarily appropriate for all residuals. Whereas the blue curve indicates that the associated polynomial does not amplify a single eigenvector, the red curve shows the opposite, such that both eigenvectors associated with eigenvalues around $\lambda = 1$ and $\lambda = 3$ are amplified. This huge difference between both curves is only due to a small change in the linear combination of eigenvectors in the residual. Because GMRES can amplify the eigenvectors for the sake of minimizing the residual, Krylov methods are not convenient in this setting despite their remarkable versatility.

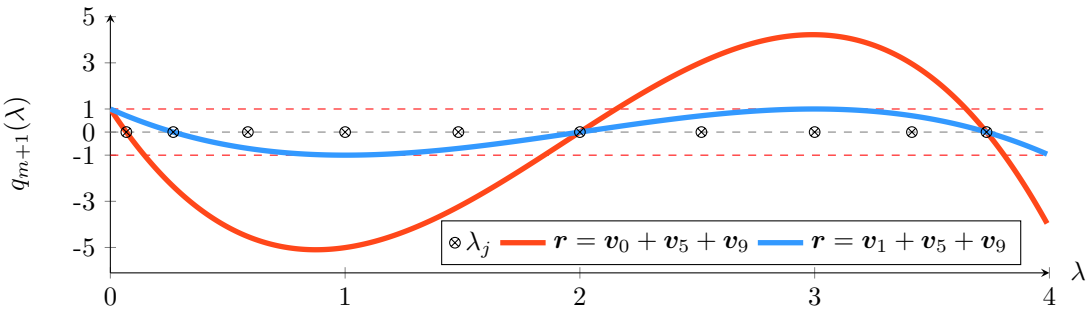
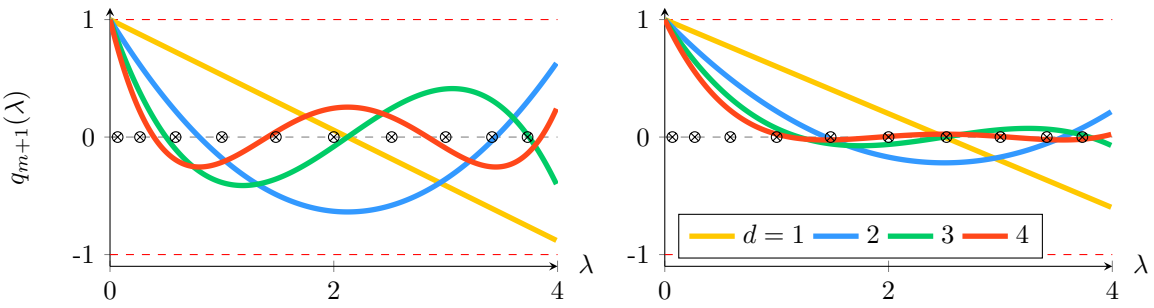


Figure 4.1: Polynomials generated by GMRES for different residual. The degree of each polynomial equals the number of eigenvectors that compose the residual.

By contrast with the right-hand side dependent polynomials that can amplify certain eigenvectors, we prefer to work with a fixed polynomial that decreases the norm of the residual by treating each eigenvector independently. The figures 4.2a and 4.2b represent the damping factors of Chebyshev polynomial smoothers, and show that small eigenvalues are not increased, whereas the large ones are damped. Note that the interval $[a, b]$ in each caption corresponds to the interval in which the Chebyshev roots of our polynomial smoother are selected as best interpolation points. Hence, preventing all eigenvectors from being amplified motivates our choice to resort to Cheybsyev polynomials for what follows.

We fix the problem of negative eigenvalues by designing the Chebyshev polynomial smoother on normal equations. The choice of the interval in which the Chebyshev roots are generated is discussed in the context of designing an algebraic multigrid method for Helmholtz. In particular, one important constraint is to maximize the



(a) $[a, b] = [0.25, 4]$ (b) $[a, b] = [1, 4]$

Figure 4.2: Spectrum of the error propagation matrix for different Chebyshev polynomials

complementarity principle with the coarse grid correction. Therefore, the size of the interval should map the expected proportion of eigenvectors that the Chebyshev polynomial smoother is supposed to damp efficiently. Lastly, numerical experiments will allow us to compare our alternative smoother with GMRES on the eigenvectors of the initial matrix.

4.1 Polynomial smoothers for Helmholtz

In Section 2.1.2.1, we recalled that Chebyshev polynomials provide the best approximation of the inverse function x^{-1} within an interval $[a, b]$. This feature enables the design of the polynomial smoother with the best convergence rate in the interval. For Helmholtz, such a smoother should damp both negative and positive eigenvalues, whereas the interval $[a, b]$ should be continuous. One way to ensure that both positive and negative eigenvectors are damped is to consider a normal equation polynomial smoother. In general, the degree d of the polynomial should be greater than one to damp positive and negative eigenvectors. Resorting to normal equations enables the polynomial to treat eigenvalues with respect to their magnitude rather than their sign, which is equivalent to working with even powers of A if the matrix is Hermitian.

Let $p_d(A^2)$ be a polynomial of degree d that approximates $(A^2)^{-1}$. From Equation (2.20), let $q_{d+1}(A^2)$ be the associated error propagation matrix of the polynomial smoother such that

$$q_{d+1}(A^2) := I - p_d(A^2)A^2. \tag{4.1}$$

Additionally, let $\boldsymbol{v}_i$ be an eigenvector of A associated with the eigenvalue λ_i. Hence,

$$q_{d+1}(\lambda^2)\boldsymbol{v}_i = (1 - p_d(\lambda_i^2)\lambda_i^2)\boldsymbol{v}_i. \tag{4.2}$$

As introduced previously, the polynomial smoother $p_d(A^2)$ is an inverse approximate of $(A^2)^{-1}$ resulting from the polynomial function $p_d(x)$ that approximates the inverse function x^{-1} from $d+1$ interpolation points x_i. The inverse function is particularly difficult to interpolate for values of x around zero. Hence, the interpolation points should be selected in the interval of large eigenvalues. Note that this choice is in accordance with most multigrid methods, where the smoother generally eliminates the large eigenvectors, contrary to the coarse correction that is generally designed

to target the small eigenvectors. The next section details how choosing a relevant interval in which selecting the interpolation points.

4.1.1 Constructing an appropriate target interval

The selection interval for the Chebyshev interpolation points denoted by x_i plays a crucial role in the convergence of the method. Used as a standalone solver, one can set $a = 0$ and choose $b = \lambda_n$ so that the polynomial is the best approximation of the inverse function for the entire spectrum of the matrix. As a multigrid smoother however, the polynomial should maximize the complementarity principle. In other words, the smoother should capture information that the coarse correction does not. As mentioned, a polynomial approximate inverse is naturally bad at damping eigenvectors associated with near-zero eigenvalues. This characteristic is highlighted by (2.26). Even though we assume A^TA to be non-singular, its spectrum is likely to contain near-zero eigenvalues where $q_{d+1}(\lambda)$ is very close to one. In fact, constructing a polynomial aimed at approximating the inverse function x^{-1} is more precise for large values of x. For this reason, we define our Chebyshev polynomial to damp the largest magnitude eigenvalues. To maximize the complementarity between the smoother and the coarse correction, the percentage of damped eigenvalues should approximate the proportion of non-coarse variables. Therefore, the interval $[a, b]$ should satisfy

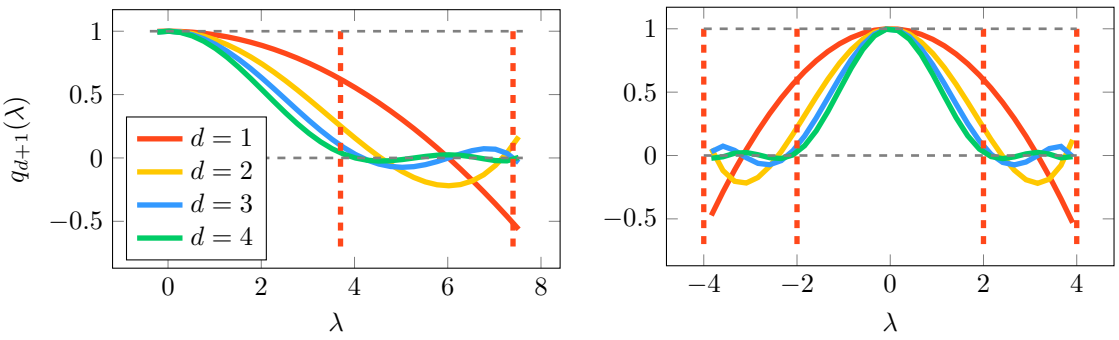
$$\lambda_i \in [-|b|, -|a|] \cup [|a|, |b|] \qquad \Leftrightarrow \qquad \lambda_i^2 \in \left[a^2, b^2\right] \quad , \quad i = n_c, \dots, n_f. \tag{4.3}$$

Whereas the upper-bound is set to include the largest eigenvalue in the interval, the lower-bound a is chosen in a second time to include n_f/n of eigenvalues.

4.1.1.1 Choosing a in proportion with b

One way to determine a good interval without preliminary information [1, 3] is to compute a few power iterations to determine b by an overestimation of the largest eigenvalue λ_n, and choose the lower-bound a according to b. With the perspective of damping the n_f largest eigenvalues in the interval, a rough and simple rule is to set $a := \frac{n_f}{n} b$. For instance, if the coarsening factor is 0.5 (i.e., $n_c = \frac{n}{2}$), then one can choose the lower-bound to be $a = \frac{1}{2} b$. This process is straightforward but assumes that eigenvalues are uniformly separated.

The following figure 4.3 portrays polynomials of different degrees where the interval is generated by this approach. The model problem is the two-dimensional shifted Laplacian SL2D introduced in (1.1). In Figure 4.3a, the shift is $\alpha = 0.625^2$ so we set $|b| = |\lambda_n| = 7.6$. Setting the lower-bound as half the upper-bound gives $|a| = 3.8$ in this example. In Figure 4.3b, $\alpha = 2^2 = 4$, such that $|b| = 4$ and $|a| = 2$. In both cases, polynomials barely oscillate within the interval because the intervals are relatively narrow. Moreover, the lower-bound a is far enough from the origin to not be impacted by Gibbs oscillation phenomena. In other words, the inverse function is accurately approximated by the polynomial p_d within the interval. While largest eigenvalues are easily damped by the smoother, the capture of intermediate ones can be improved by increasing the interval size.



(a) $\alpha = 0.625^2$ (b) $\alpha = 1.5^2$

Figure 4.3: Polynomials generated by setting $|a| = \frac{1}{2}|b|$, with $b = \lambda_n$

We achieve this in what follows by computing the spectral density approximation in order to better estimate the lower-bound a based on the proportion that needs to be damped by the polynomial smoother.

4.1.1.2 Estimating the spectrum of the matrix

Because eigenvalues of multigrid matrices are not necessarily uniformly separated, the above process may lack accuracy. In this section, we present an algebraic option based on a rough approximation of the matrix *spectral density*. More details on this technique can be found in [57]. This spectral density allows us to determine which portion of the spectrum should be damped by the smoother. While it may be reasonable to assume uniform distribution of eigenvalues on the finest level, spectral properties of coarse matrices are difficult to predict in algebraic multigrid. In fact, the purpose of this spectral approximation approach is especially to design an algebraic multilevel method for general indefinite and eventually non-Hermitian problems.

Setting the interval in this case works by first estimating the upper-bound b by a few power iterations as mentioned in Section 4.1.1.1. The lower-bound a is chosen based on the spectral density approximation, so that the probability within the interval $[a, b]$ equals the target proportion, for instance half of the total area in a scenario of exact balance between coarse and non-coarse variables.

Let $\phi(t)$ be the distribution function that represents the probability of finding an eigenvalue at each point of an interval $[-1, 1]$. The spectral density function ϕ is approximated by a linear combination of orthogonal Chebyshev polynomial functions as defined in (2.27), such that

$$\phi(t) = \sum_{j=1}^{\infty} \mu_j T_j(t) \approx \sum_{j=1}^{n_\mu} \mu_j T_j(t). \tag{4.4}$$

Because Chebyshev functions T_j are naturally defined over $[-1, 1]$, the approximation in (4.4) should target the spectral density of the scaled matrix

$$B := \frac{A - cI}{d}, \text{ with } c := \frac{\lambda_{\min} + \lambda_{\max}}{2} \text{ and } \frac{\lambda_{\max} - \lambda_{\min}}{2}. \tag{4.5}$$

In the next section, different scaling approaches are discussed. Yet, we assume that the spectrum of B belongs to the interval $[-1, 1]$. Subsequently, the coefficients μ_j of the distribution function $\phi(t)$ in (4.4) can be derived by a moment matching procedure, such that

$$\mu_j = \frac{2 - \delta_{j1}}{n\pi} \times \text{Trace}(T_j(B)). \tag{4.6}$$

As usual, n designates the matrix size, whereas δ_{j1} denotes the Kronecker symbol. The trace in (4.6) that needs to be computed for each coefficient μ_j of the spectral density function can be estimated by a stochastic process based on a set of n_{vec} random and orthogonal vectors $\boldsymbol{z}_l$, such that

$$\text{Trace}(T_j(B)) = \mathbb{E}[\boldsymbol{z_l}^T T_j(B)\boldsymbol{z_l}] \approx \frac{1}{n_{\text{vec}}} \sum_{l=1}^{n_{\text{vec}}} \boldsymbol{z}_l^T T_j(B)\boldsymbol{z}_l. \tag{4.7}$$

Each entry of $\boldsymbol{z}_l$ in (4.7) is generated randomly following a normal distribution with zero mean and a unit standard deviation. The vectors $T_j(B)\boldsymbol{z_l}$ can be computed successively from the three-term recurrence that characterizes Chebyshev polynomial functions as defined in (2.28). However, it is well known that the accuracy of polynomial approximations is affected by Gibbs oscillations. One common practice to address this concern is to approximate the distribution function of (4.4) by a Chebyshev-Jackson approximation instead. The slight difference is that each coefficient μ_j is modulated by a weight $g_j^{n_\mu}$ as follows

$$\phi(t) \approx \hat{\phi}(t) = \sum_{j=1}^{n_\mu} \mu_j g_j^{n_\mu} T_j(t). \tag{4.8}$$

Defining $\zeta_{n_\mu} := \frac{\pi}{n_\mu+1}$, the regularization weights of (4.8) are given by

$$g_j^{n_\mu} := \frac{\left(1 - \frac{j}{n_\mu+1}\right) \sin\zeta_{n_\mu} \cos k\zeta_{n_\mu} + \frac{1}{n_\mu+1} \cos\zeta_{n_\mu} \sin k\zeta_{n_\mu}}{\sin\zeta_{n_\mu}}. \tag{4.9}$$

Before pursuing this further, let us summarize the spectral density approximation step. The spectral density function is approximated by a linear combination of Chebyshev functions, where each coefficient μ_j is given by (4.6). Each coefficient requires computing a stochastic trace estimation as described in (4.7). Once these μ_j are obtained, they should be weighted by the coefficients $g_j^{n_\mu}$ to finally end with the spectral density approximation denoted by $\hat{\phi}$ in (4.8). The overall spectral density approximation phase is also given in Algorithm 2.

Algorithm 2 Spectral density approximation

$B \leftarrow A/|b|$
for $l = 1, n_{\text{vec}}$ **do**
  $\hat{\boldsymbol{z}}_l \leftarrow \text{RandomVector}(n, \mu = 0, \sigma = 1)$ ▷ $\boldsymbol{z}_l$ are generated randomly following $\mathcal{N}(0, 1)$
  for $j = 1, l$ **do**
    $\boldsymbol{z}_l \leftarrow \boldsymbol{z}_l - \boldsymbol{z}_j^* \boldsymbol{z}_l \cdot \boldsymbol{z}_j$ ▷ Orthonormalization of the random vector
  end for
  $\boldsymbol{z}_l \leftarrow \boldsymbol{z}_l / ||\boldsymbol{z}_l||_2$
  $\hat{\boldsymbol{z}}_l \leftarrow \boldsymbol{z}_l$
  $\boldsymbol{w} \leftarrow B\hat{\boldsymbol{z}}_l - \boldsymbol{z}_l$
  for $j = 1, n_\mu$ **do**
    $\mu_j \leftarrow \mu_j + \hat{\boldsymbol{z}}_l^* B \hat{\boldsymbol{z}}_l$ ▷ Stochastic Trace approximation iteration
    $\boldsymbol{u} \leftarrow \boldsymbol{w}$
    $\boldsymbol{w} \leftarrow \hat{\boldsymbol{z}}_l$
    $\hat{\boldsymbol{z}}_l \leftarrow B\boldsymbol{w} - \boldsymbol{u}$ ▷ Three-term recurrence relation
  end for
end for
for $j = 1, n_\mu$ **do**
  $\mu_j \leftarrow \mu_j \times g_j^{n_\mu} \frac{2 - \delta_{j1}}{n\pi n_{\text{vec}}}$ ▷ μ_j are averaged and weighted by $g_j^{n_\mu}$.
  $\hat{\phi} \leftarrow \mu_j \times \cos(j \arccos(t_i)),\ \forall t_i \in \{-1, 1\}_h$. ▷ $\{-1, 1\}_h$ is a discretized interval of $[-1, 1]$
end for
return $\hat{\phi}$

The next step is to integrate the density function $\hat{\phi}$ in (4.8) from the upper-bound to the left, until reaching the desired area under the curve (i.e., the correct proportion of eigenvalues). The resulting lower-bound in the re-scaled interval $[-1, 1]$ should finally be remapped to the interval of the initial matrix with the aim of finally obtaining the correct value of a^2 in (4.3). This last step depends on the chosen scaling approach for the matrix B, and is therefore discussed in the next sections.

4.1.1.3 Scaling with the initial matrix

This section exposes a first approach for choosing the scaled matrix B in (4.5). Both estimates $\lambda_{\min}$ and $\lambda_{\max}$ should be chosen so that the spectrum of B belongs to the interval $[-1, 1]$ of the spectral density function. Accordingly, in the indefinite case, $\lambda_{\min}$ and $\lambda_{\max}$ should encapsulate negative and positive eigenvalues. Assuming that b is an upper-bound of A eigenvalues, one rough strategy for choosing the scaled matrix B is by setting

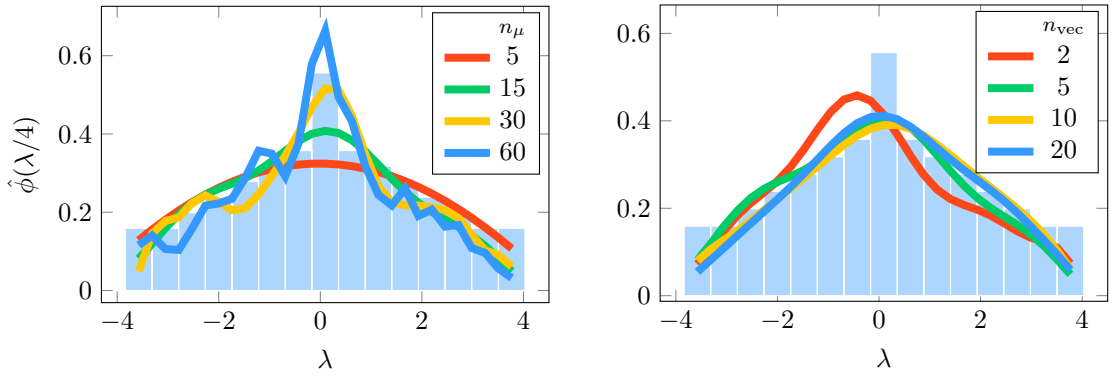
$$\lambda_{\max} := |b| \quad \text{and} \quad \lambda_{\min} := -|b|. \tag{4.10}$$

This choice is especially relevant in the case of a balance between positive and negative eigenvalues, for instance, when α gets closer to 4 in the SL2D model problem (1.1). Injecting (4.10) in (4.5), the scaled matrix B based on the initial matrix A has the form

$$B := \frac{A}{|b|}. \tag{4.11}$$

The Figure 4.5 plots two experiments of the spectral density approximation approach on the SL2D model problem with shift $\alpha = 4$, and using the scaling matrix B of (4.11). In this case, the maximal eigenvalue is $b = 4$. The spectral density

approximations illustrated in Figure 4.7b resort to a fixed number n_{vec} of random vectors $\boldsymbol{z}_l$ in the stochastic trace approximation step (4.7), whereas the number n_μ of Chebyshev functions in (4.4) varies. Conversely, Figure 4.7a represents the same experiment but with n_μ fixed and a varying number n_{vec} of random vectors. While the former shows that larger n_μ fits the variation of the spectral density function better, the latter reveals that increasing n_{vec} gives smoother estimations.



(a) $n_{\text{vec}} = 5$, n_μ varying

(b) n_{vec} varying, $n_\mu = 15$

Figure 4.4: Density of State of the SL2D matrix with shift $\alpha = 2.0^2$

Once the spectral density function $\hat{\phi}$ is approximated with $\hat{\phi}$, one can integrate it to determine a good interval $[-|b|, -|a|] \cup [|a|, |b|]$ that satisfies (4.3). Therefore, the next step consists of applying an area approximation using trapezoid integration to determine the value a, such that a proportion n_f/n eigenvalues are covered by the interval $[-|b|, -|a|] \cup [|a|, |b|]$. As the polynomial smoother should damp the large eigenvalues magnitude-wise, we can sum up the density of negative eigenvalues with the density of positive eigenvalues, such that

$$\forall\, t \in [0,1] \qquad , \qquad \hat{\phi}_+(t) = \hat{\phi}(t) + \hat{\phi}(-t). \tag{4.12}$$

Recall that the upper-bound of $\hat{\phi}_+$ is 1 and maps b. Therefore, we seek the lower-bound t_a such that n_f/n of the area under the curve of $\hat{\phi}_+$ belongs to $[t_a, 1]$. In practice, t_a is computed by integrating $\hat{\phi}_+$ as follows

$$\begin{aligned} t_a &= \underset{t_* \in [0,1]}{\arg\min} \left(\frac{n_f}{n} - \int_{t_*}^{1} \hat{\phi}_+(t) dt \right) \\ &\approx 1 - h_t \times \underset{i_*}{\arg\min} \left(\frac{n_f}{n} - \sum_{i=0}^{i_*} \frac{\hat{\phi}_+(1 - h_t i) + \hat{\phi}_+(1 - h_t i - h_t))}{2} \times h_t \right), \end{aligned} \tag{4.13}$$

where h_t is a stepping size. Lastly, the estimated lower-bound t_a has to be remapped on the initial interval to return the correct value for $|a|$. From (4.5), we have

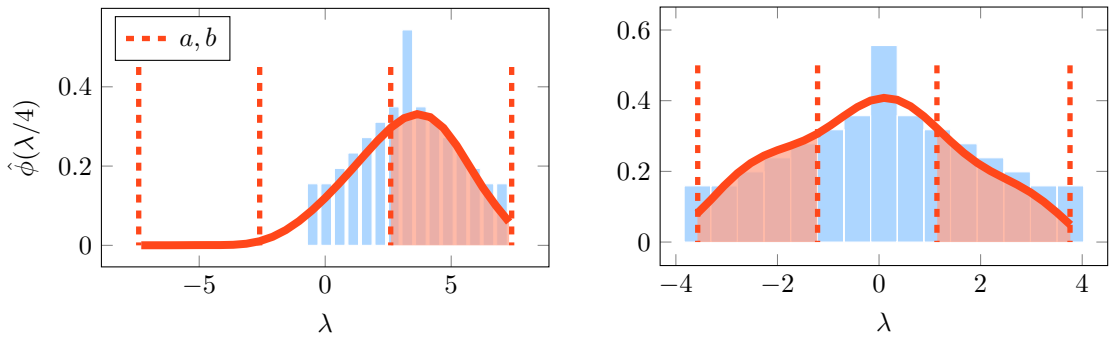
$$|a| = |b| \cdot t_a. \tag{4.14}$$

The interval $[a^2, b^2]$ therefore constitutes a purely algebraic interval for the roots of the Chebyshev polynomial smoother based on A^2. The lower-bound estimation is summarized by Algorithm 3, and illustrated in Figure 4.5.

Algorithm 3 Construction of the interval $[a, b]$ based on spectral density approximation

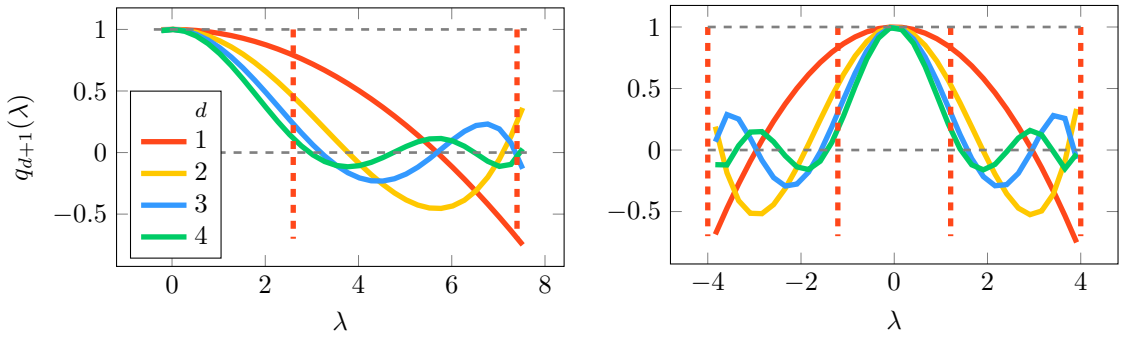
$\hat{\phi} \leftarrow$ ComputeSpectralDensity(A,b) ▷ From Algorithm 2
$\hat{\phi}_+ \leftarrow \hat{\phi}(t) + \hat{\phi}(-t)\ \forall t_i \in \{0,1\}_{h_t}$. ▷ $\{0,1\}_{h_t}$ is a discretized interval of $[0,1]$
end for
area $\leftarrow 0, t \leftarrow 1$
while area $< {}^{n_f}/_{n}$ **do**
    $t_a \leftarrow t - h_t$
    area $\leftarrow$ area $+ h_t/2 \times \left(\hat{\phi}_+(t_a) + \hat{\phi}_+(t)\right)$ ▷ Integration using trapezoid formulas
    $t \leftarrow t - h_t$
end for
$|a| = {}^{|b|t_a}/_{2}$ ▷ Lower-bound t_a needs to be re-scaled to the initial matrix spectrum

Integrating the spectral density function offers a wider interval by enabling a more accurate estimation for the lower-bound a^2 than selecting a in proportion with b. In Figure 4.5a, $|a| = 2.6$ for $\alpha = 0.625^2$ whereas $|a| = 1.2$ for $\alpha = 2.0^2$ in Figure 4.7a. These numbers are smaller than the lower-bound estimates portrayed by figures 4.3a and 4.3b. Despite more important oscillations, the resulting polynomials portrayed in 4.6 prove to damp a larger proportion of eigenvalues.



(a) $\alpha = 0.625^2$ (b) $\alpha = 2.0^2$

Figure 4.5: Density of State of the SL2D matrix by setting $\lambda_{\min} = -|b|$ in (4.5)



(a) $\alpha = 0.625^2$ (b) $\alpha = 1.5^2$

Figure 4.6: Spectrum of q_{d+1} with roots selected in $[a, b]$ in (4.5)

Let us discuss one downside of this scaling approach based on the initial matrix. It is well known for the model problem SL2D that none of its eigenvalues belongs to

$[-|b|, -\alpha]$. However, the area that the density curve covers within this interval is not strictly zero as it should be. This is due to the density function that smoothly grows from $-|b|$ to the actual non-zero probability area which begins at $-\alpha$. In other words, the rough estimation $\lambda_{\min} = -|b|$ for the definition of the scaled matrix in (4.5) creates a gap that impacts the shape of the curve, and subsequently degrades the lower-bound estimation. For instance, the blue histogram in the background of Figure 4.8a is symmetric contrary to the curve in the foreground. As the consequence, the lower-bound estimate $|a|$ is slightly shifted from the middle peak. This rough estimation of the largest negative eigenvalue $\lambda_{\min}$ is probably enough in the context of designing a good polynomial smoother for Helmholtz as highlighted by Figure 4.6.

One geometrical approach for fixing this issue when solving the Helmholtz SL2D model problem is to set the lower bound as the smallest eigenvalue and the upper bound as the largest eigenvalue, such that $\lambda_{\min} = -\alpha$ and $\lambda_{\max} = 8 - \alpha$. In that case, the scaled matrix of (4.5) would be

$$B := \frac{A - (4-\alpha)I}{4} = \frac{1}{4} \cdot (A - \mathrm{Diag}(A)) \tag{4.15}$$

A second algebraic strategy for solving the gap issue caused by the useless portion $[-|b|, -\alpha]$ is to work on normal equations, but at the price of doubling the number of matrix vector products in the overall spectral density approximation. The next section discusses this second approach.

4.1.1.4 Scaling with the normal equation matrix

Scaling on the normal equation matrix $A^T A$ is convenient to determine the density function in magnitude, but at the cost of doubling the number of matrix vector products in the spectral density approximation phase. In that case, the lower-bound $\lambda_{\min}$ can simply be set to 0 if the matrix is squared. The upper-bound b of the Chebyshev nodes interval can still be computed by a few power iterations as in the previous scaling approach. Here, we assume that b corresponds to an upper-bound of the largest eigenvalue of the normal equation matrix $A^T A$. Estimating the lower-bound a using the spectral density approximation still requires to rescale the normal equation matrix. In this second scaling approach, we set

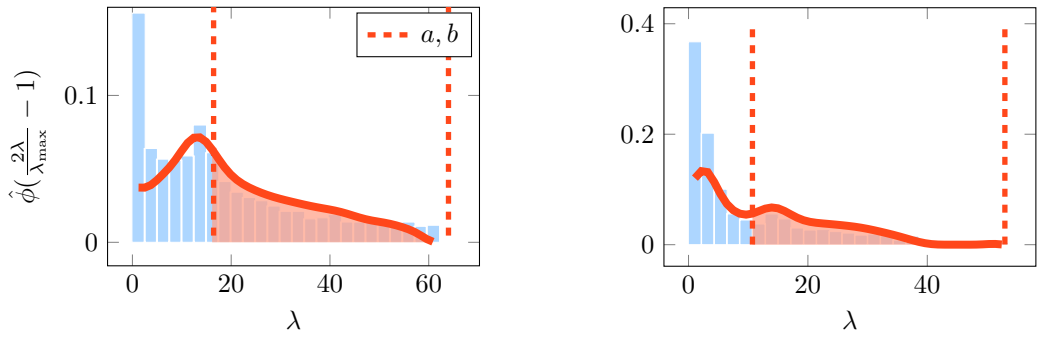
$$\lambda_{\min} = 0 \qquad \text{and} \qquad \lambda_{\max} = b \tag{4.16}$$

such that the scaled matrix B of (4.5) becomes

$$B = \frac{2A^* A}{b} - I. \tag{4.17}$$

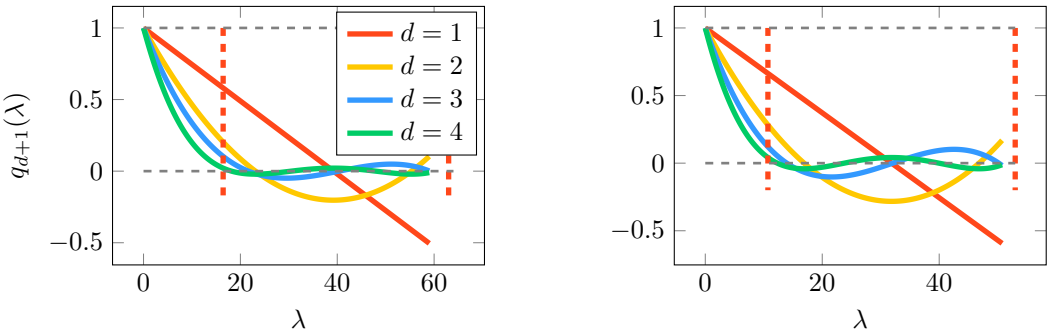
Since the eigenvalues of B are positives, its spectral density approximation does not need to be summed up with the negative portion as in (4.12). The lower-bound estimations for the same shifts as for Figure 4.3 are illustrated in 4.7. These numerical experiments do not emphasize any incidence of the shift on the accuracy of the spectral density approximation. The red curve fits the variation of the histogram accurately except for near-zero eigenvalues. In comparison with the previous figure,

Figure 4.8 plots different Chebyshev polynomials after estimating the lower-bound with the spectral density setting. Especially, these figures reveal how large the interval of eigenvalues is compared to the remaining portion that should be treated by the coarse correction, while containing 50% of the spectrum only.



(a) $\alpha = 0.625^2$ (b) $\alpha = 1.5^2$

Figure 4.7: Interval estimation of $[a, b]$ for $A^T A$



(a) $\alpha = 0.625^2$ (b) $\alpha = 1.5^2$

Figure 4.8: Spectrum of q_{d+1} with roots selected in $[a, b]$ for $A^T A$

Both scaling approach can be used to determine a good interval $[a^2, b^2]$ for selecting the Chebyshev interpolation points. Thereafter, the Lagrangian formula (2.24) enables the computation of the coefficients of the polynomial $p_d(A^T A)$ that approximates the inverse of $A^T A$.

4.1.2 Numerical experiments on the smoother

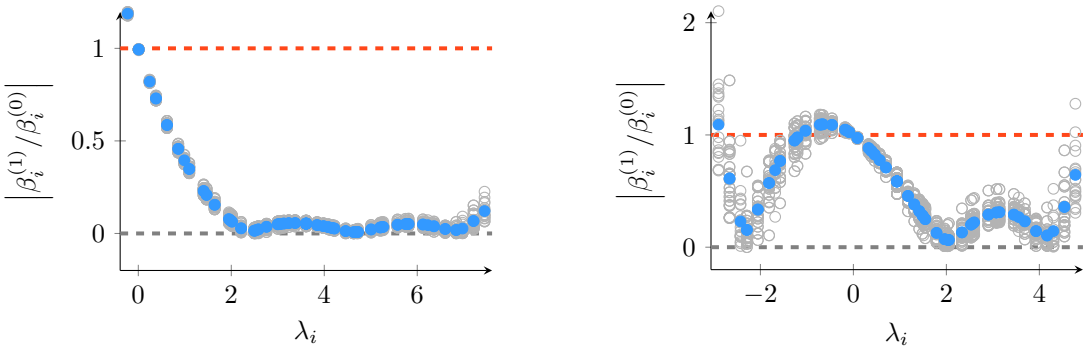
Used as a standalone solver, the main goal when designing an iterative method is to minimize its convergence rate. However, in our case, the smoother is implemented within the multigrid cycle, and is therefore coupled with an additional coarse grid correction. As discussed previously, the keystone that drives convergence of a multigrid method is the complementarity between both operators. For fast convergence, the smoother should be effective where the coarse correction is not and vice versa. To emphasize this feature of our Chebyshev polynomial, let us first analyze the effect of the GMRES method on each eigenvector of the initial matrix.

Since Krylov methods are right-hand side dependent, it is not insightful to extract the eigenvalues of its error propagation matrix. Let $\boldsymbol{r}^{(0)}$ be the initial residual. We compute the residual $\boldsymbol{r}^{(1)}$ after one single smoothing step. Both residuals can be decomposed as a linear combination of eigenvectors as follows

$$\boldsymbol{r}^{(0)} = \sum_{i=1}^{n} \beta_i^{(0)} \boldsymbol{v}_i \quad \text{and} \quad \boldsymbol{r}^{(1)} = \sum_{i=1}^{n} \beta_i^{(1)} \boldsymbol{v}_i. \tag{4.18}$$

For each eigenvector $\boldsymbol{v}_i$, Figure 4.9 plots the ratios $|\beta^{(1)}/\beta^{(0)}|$ with respect to the eigenvalues for two different shifts. In this first experiment, three Krylov vectors are generated to construct the basis. This numerical experiment is run 25 times. Each gray mark represents the effect of the method on one particular eigenvector and for each of the 25 initially random residuals $\boldsymbol{r}^{(0)}$. The blue marks correspond to the average of the gray marks for each eigenvector.

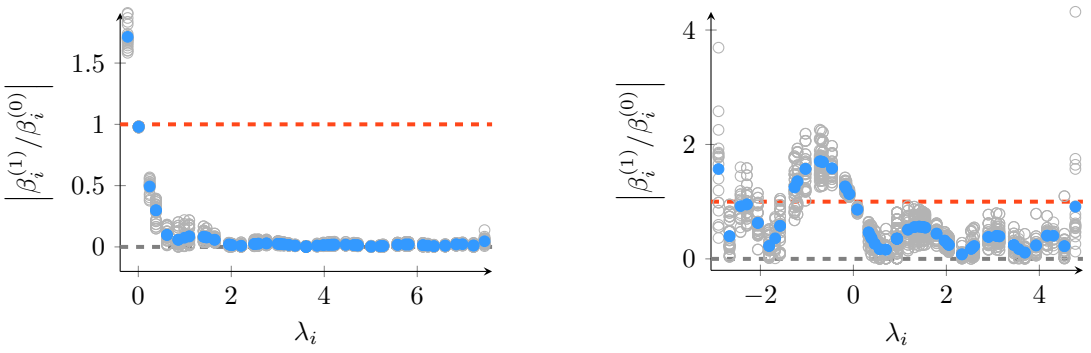
The left Figure 4.9a that corresponds to the shifted case $\alpha = 0.625^2$ illustrates how efficient the Krylov iterations are in damping the large positive eigenvectors. However, the smallest magnitude eigenvalue remains untouched, and the negative small eigenvalue on its left gets amplified. In a similar way, the right Figure 4.9b shows that the damping factors of large positive eigenvalues oscillate around zero, and that the largest negative eigenvalue is also amplified when increasing the shift to $\alpha = 1.75^2$.



(a) $\alpha = 0.625^2$ (b) $\alpha = 1.75^2$

Figure 4.9: Effect of $\nu = 1$ smoothing step of GMRES(3) on the eigenvectors of the SL2D problem for two different shifts.

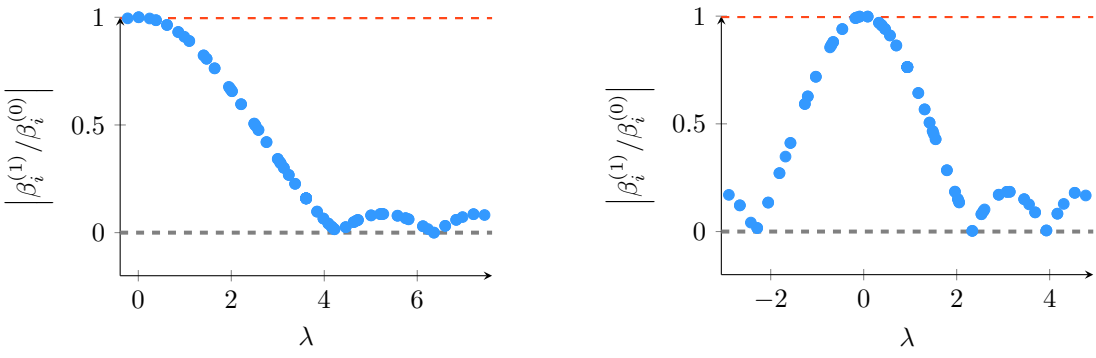
Similar observations can be made in Figure 4.10 when increasing the size of the minimization space to 6 Krylov vectors. Moreover, Figure 4.10b reveals that not only the largest negative eigenvalues are amplified, but also the intermediate ones. In fact, nothing guaranties a symmetric behavior of the Krylov method between the negative and positive eigenvalues. Due to the right-hand side dependency, it remains difficult to predict its effect in all cases and for each eigenvector, especially for the coarse matrices of a multilevel hierarchy. Subsequently, the coarse correction may need to hit several separated intervals of the spectrum to satisfy the complementarity principle when using a Krylov method as a smoother.



(a) $\alpha = 0.625^2$ (b) $\alpha = 1.75^2$

Figure 4.10: Effect of $\nu = 1$ smoothing step of GMRES(6) on the eigenvectors of the SL2D problem for two different shifts.

The smoother will play an important role in the design of good interpolation rules and in the coarse correction process. We refer to chapters 5 and 6 for more details on these topics. The fact that GMRES eventually amplifies certain regions of the spectrum motivated us to design an alternative smoother whose spectral behavior is a priori known and guaranteed to not amplify any eigenvector. The Chebyshev polynomial smoother behaves the same independently of the right-hand side, and shares the same eigenvectors with the initial matrix. Subsequently, it follows that the ratios $\beta_i^{(1)}/\beta_i^{(0)}$ correspond to the eigenvalues of the error propagation matrix $q_{d+1}(A^TA)$ in absolute values. As a consequence, the gray marks are all the same and hidden behind the average blue marks in Figure 4.11. In this experiment, one iteration of the Chebyshev polynomial smoother with normal equations is applied, and the degree is set to 3. Moreover, the degree of the error propagation matrix is 3 with respect to A^TA, which corresponds to a polynomial of degree 6 with respect to A in the Hermitian case.



(a) $\alpha = 0.625^2$ (b) $\alpha = 1.75^2$

Figure 4.11: Effect of $\nu = 1$ smoothing step of our normal equations polynomial smoother with degree 6 on the eigenvectors of the SL2D problem for two different shifts.

The Chebyshev polynomial smoother with normal equations prevents any amplification of the eigenvectors as discussed in Section 2.1.2.2. This convenient feature enables a better knowledge of which space the coarse correction should target, and

will help the construction of the interpolation operator in Chapter 5. This benefit may be at the price of smaller damping factors, as it is apparent when comparing Figure 4.11a with Figure 4.10a. However, it enables the convergence of the smoother in the indefinite case, and is therefore extremely useful in cases where the proportion of negative and positive eigenvalues tends to be the same. In particular, comparing 4.11b with 4.10b highlights that the polynomial smoother seems to offer similar damping properties to GMRES, while preventing each eigenvector from being amplified. These smoothing properties help in the multilevel case, as coarser matrices in the multigrid hierarchy tend to be very indefinite as illustrated in Figure 7.6 for instance.

However, when the proportion of positive and negative eigenvalues gets unbalanced, for instance, for small shifts, certain roots are "wasted" on the side where only a few eigenvalues are located. In the future, it would be interesting to avoid normal equations to allow more flexible smoothing properties. Different polynomials may still prevent the amplification of certain eigenvectors while offering even better convergence properties.

4.2 Intermediate Conclusion

In this chapter, we introduced a Chebyshev polynomial smoother with normal equations to damp the large magnitude eigenvalues independently of their signs. The Chebyshev framework provides the best polynomial inverse approximate of the initial matrix for a given interval. Hence, the choice of the interval is important, and can be defined in several ways. While geometrical information can help, we also introduced a purely algebraic approach based on a spectral density approximation of the initial matrix.

Used as a stand-alone solver, the convergence of our polynomial smoother can be outperformed by Krylov methods, but its effect on the spectrum is known in advance and ensures that no eigenvector is amplified in the smoothing process. As discussed in chapters 5 and 6, these features are key in the design of a good interpolation operator, but also in the contraction rate of an adapted coarse correction for Helmholtz.

Chapter 5

Interpolation rules for Helmholtz

The complementarity between the smoother and the coarse correction drives the convergence of multigrid methods. In Chapter 4, we introduced a smoother that damps the large eigenvalues of the Helmholtz matrix magnitude-wise. Accordingly, the coarse correction should capture the small magnitude eigenvalues. The design of an appropriate interpolation operator P is especially important because the coarse correction works by projecting the residual onto the range of interpolation. Therefore, the span of P should approximate the space that the smoother damps the least, while keeping a practical structure that counts as few non-zero entries as possible.

Authors introduced several approximation properties that P should satisfy to maximize the effectiveness of the coarse correction. The optimal framework introduced in Chapter 2 provides the range of the best theoretical interpolation operator for a given smoother (2.85). In our case, the optimal interpolation operator prescribed by the theory should span the oscillatory space of eigenvectors associated with small magnitude eigenvalues. In what follows, and as in Chapter 3, we designate this space by V_c. However, the optimal framework does not address the question of practical implementation. In a different manner, the ideal theory provides a more practical form of interpolation based on a set of coarse and fine variable operators. Such an ideal interpolation operator is usually dense and subsequently impractical. However, it allows approximations aimed at satisfying a trade-off between sparsity and good interpolation rules.

The ideal interpolation operator P_* based on the classical coarse and fine variable operators R^T and S (2.75) can yield good convergence in practice. However, the variable operators are not designed with the perspective of addressing Helmholtz. As a consequence, it is difficult to predict the efficiency of an approximation P of the classical ideal interpolation operator P_*. Alternatively, we introduce a new coarse variable operator $\hat{R}^T$ based on a least-squares minimization strategy. Thereafter, we construct its orthogonal fine variable counterpart $\hat{S}$ such that $\hat{R}\hat{S} = 0$. Here, the process of approximating the ideal interpolation $\hat{P}_*$ is more predictable because $\hat{R}^T$ and $\hat{S}$ are better initial approximations of the theoretical complementarity required. In this second approach, $\hat{P}$ denotes the approximation of $\hat{P}_*$.

Throughout this chapter, we discuss the approximation properties of our interpola-

tion operator along with its complexity and its effect on the concept of "pollution" introduced in 3.2. With the aim of evaluating the approximation properties of the different operators in this chapter, we use the l_2-orthogonal projection $\Pi(P)$ defined in (3.69). We say that a vector $\boldsymbol{v}$ is "close" to the range of P if $\boldsymbol{v}^T(I - \Pi(P))\boldsymbol{v} \approx 0$. Conversely, we say that $\boldsymbol{v}$ is "poorly approximated" by P when $\boldsymbol{v}^T(I - \Pi(P))\boldsymbol{v} \approx 1$.

5.1 Guidance of the optimal theory

To design a good interpolation operator for Helmholtz, we start by looking at the optimal interpolation range prescribed by the theory. As prescribed by Lemma 1, given a smoothing matrix M^{-1} that approximates A^{-1}, the optimal interpolation operator $P_\#$ satisfies

$$\text{Range}(P_\#) = \text{span}\left(\{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{n_c}\}\right), \tag{5.1}$$

where $\boldsymbol{u}_i$ is the ith eigenvector associated with the ith largest eigenvalue μ_i in magnitude of the generalized eigenvalue problem

$$A\boldsymbol{u}_i = \mu_i M\boldsymbol{u}_i. \tag{5.2}$$

The range of optimal interpolation in (5.1) corresponds to the space spanned by eigenvectors $\boldsymbol{u}_i$ associated with the smallest eigenvalues μ_i in magnitude, which corresponds to the space that the smoother damps the least. In our case, we use a Chebyshev polynomial smoother of degree d, such that $p_d(A^2)$ approximates the inverse matrix A^{-2}. Therefore, the generalized eigenvalue problem (5.2) becomes

$$A^2\boldsymbol{u}_i = \mu_i \left(p_d(A^2)\right)^{-1} \boldsymbol{u}_i. \tag{5.3}$$

Recall that $\boldsymbol{v}_i$ is the eigenvector associated with the ith largest magnitude eigenvalue λ_i. Because both A^2 and $p_d(A^2)$ have the same eigenvectors, we have $\boldsymbol{u}_i = \boldsymbol{v}_i$. From (5.3), it follows that

$$\mu_i = \lambda_i^2 p_d(\lambda_i^2). \tag{5.4}$$

In addition, recall that $q_{d+1}(A^2) = I - p_d(A^2)A^2$ is the error propagation matrix of $p_d(A^2)$. Hence, eigenvalues of the generalized eigenvalue problem (5.3) are finally given by

$$\mu_i = 1 - q_{d+1}(\lambda_i^2). \tag{5.5}$$

The polynomial smoother damps an eigenvector $\boldsymbol{v}$ slowly if its associated eigenvalue $\lambda \approx 0$, such that $q_{d+1}(\lambda^2) \approx 1$. Therefore, the most difficult eigenvector for our polynomial smoother is $\boldsymbol{v}_1$. As a consequence, the smallest eigenvalue of the generalized eigenvalue problem (5.3) is $\mu_1 = 1 - q_{d+1}(\lambda_1^2) \approx 0$ assuming $\lambda_1 \approx 0$. From (5.2), the optimal theory states that $\boldsymbol{v}_1$ is the most important to include in the range of the optimal interpolation operator. The same reasoning applies for all of the other eigenvectors of V_c as well.

To summarize, the optimal range finally corresponds to the space spanned by the small eigenvectors of V_c, including the near-kernel space, since the polynomial smoother damps the complementary subspace V_f faster. In other words, the optimal interpolation operator satisfies

$$\text{Range}(P_\#) = \text{span}\left(V_c\right). \tag{5.6}$$

5.2 Classical framework

The optimal theory prescribes the range of the best interpolation operator, but no insight on its practical form. Conversely, the ideal theory does not prescribe the optimal interpolation operator, but the best interpolation operator that can be derived from a given set of coarse and fine variable operators. Still, the ideal interpolation operator minimizes the interpolation error of eigenvectors in proportion with the inverse of associated eigenvalues. Hence, even the "classical" ideal interpolation operator P_* of (2.77) should have good approximation properties of the near-kernel space of V_c. However, the quality of an approximation P of P_* is less evident. New variable operators based on a least-squares minimization strategy will be introduced in the next section to address this issue. But first, let us discuss on the properties satisfied by approximations of P_*.

5.2.1 Classical variable operators and ideal interpolation operator

The splitting between coarse and fine variables has been done since the early days of multigrid method research. In classical multigrid settings, the coarse and fine variable splitting is simply based on an initial selection of n_c $\mathcal{C}$-points and n_f $\mathcal{F}$-points. In classical AMG, the selection of the coarse variables is aimed at tracking the geometrical smoothness of the near-kernel space of elliptic problems by way of the strength of connection rule (2.56). In our case, the near-kernel space is oscillatory. Hence, we measure the connection between the entries magnitude-wise and define the strong connection groups as follows

$$\mathcal{S}_i := \left\{ \, j \, , \; |a_{ij}| \geq \theta \max_{k \neq i} \left(|a_{ik}| \right) \, \right\}. \tag{5.7}$$

The n_c selected $\mathcal{C}$-points are those with the most strongly connected points, and their neighbors form the group of $\mathcal{F}$-points. Let the rows and the columns of A be permuted based on their $\mathcal{C}$-points and $\mathcal{F}$-points affiliation. We assume that A has the form (2.76). Then, recall that the classical coarse and fine variable operators are respectively defined by

$$R^T = [0 \; I_c]^T \qquad \text{and} \qquad S = [I_f \; 0]^T. \tag{5.8}$$

The ideal interpolation operator works by removing the S-related space that the smoother should handled from range(R^T) to better target the information that the coarse correction should capture, such that

$$P_* := \Big(I - \Pi_A \left(S \right) \Big) R^T = \Big(I - S \left(S^T A S \right)^{-1} S^T A \Big) R^T. \tag{5.9}$$

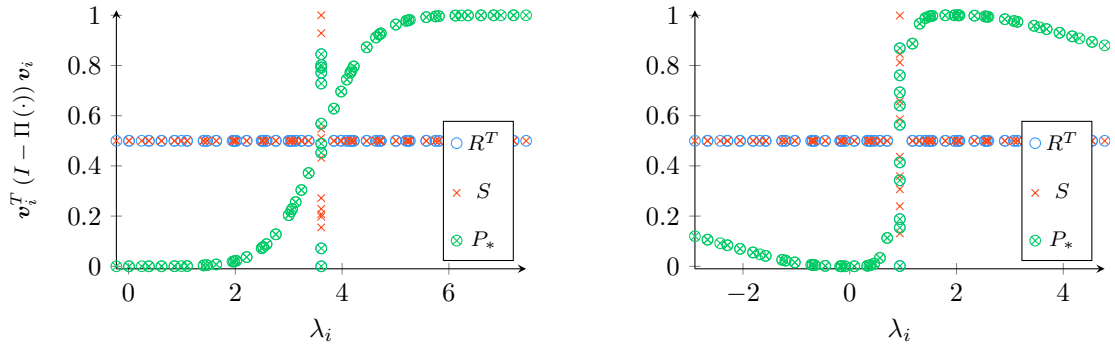
When using the variable operators (5.8), let us recall that the ideal interpolation operator has the form

$$P_* = \begin{bmatrix} -A_{ff}^{-1} A_{fc} \\ I_c \end{bmatrix}. \tag{5.10}$$

Assuming A defines a norm (which is not true in the indefinite case), constructing each column of P_* is equivalent to solving the n_c minimization problems

$$[P_*]_{:,i} = R^T_{:,i} - \boldsymbol{s}_i \qquad \text{with} \qquad \boldsymbol{s}_i := \underset{\tilde{\boldsymbol{s}} \in \text{Range}(S)}{\arg\min} \, ||R^T_{:,i} - \tilde{\boldsymbol{s}}||_A \quad i = 1, \ldots, n_c. \tag{5.11}$$

For this reason, choosing a relevant set of coarse and fine variable operators is crucial in the design of a multigrid method. For instance, Figure 5.1 presents the error of interpolation of every eigenvector $\boldsymbol{v}_i$ using the measure (3.69) for both operators R^T and S. One clear observation is that none of the classical variable operators distinguishes the smallest eigenvectors of V_c from the largest of V_f. In other words, the classical coarse variable operator R^T does not have good approximation property for the oscillatory near-kernel space that characterizes Helmholtz. Conversely, the range of the classical fine variable operator S does not approximate the space for which the polynomial smoother introduced in Chapter 4 is the most effective.



(a) $\alpha = 0.625^2$ (b) $\alpha = 1.75^2$

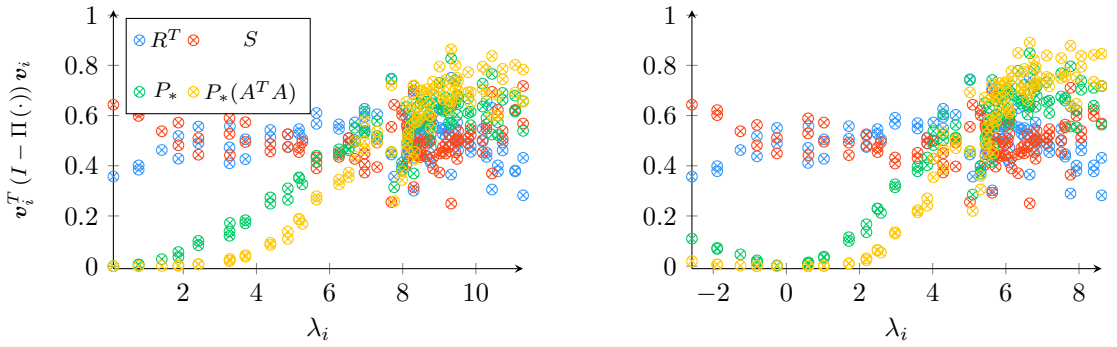
Figure 5.1: Error of the l_2-projection onto the range of classical variable operators R^T and S and of the ideal interpolation operator P_* for two different shifts

Nevertheless for Helmholtz, the classical definitions in (5.8) may still lead to a good ideal interpolation operator since P_* is the optimal minimizer of the quantity μ_M in (2.65). Because this quantity is weighted by the inverse of an A-norm, the classical ideal interpolation operator focuses on the smallest eigenvectors very well, as illustrated in Figure 5.1.

In the next section, we address one approximation approach of P_* based on the set (5.8). When applying a red-black $\mathcal{C}/\mathcal{F}$ splitting for the 5-point stencil matrix (1.2), then the block A_{ff} is diagonal. In that particular case, P_* is practical. To prevent our approximations from benefiting from this special property, we consider the 9-point stencil (1.3) in this chapter. The analysis remains the same, the only difference is the structure of the initial matrix A that challenges our approximations better in the 9-point stencil case.

Hence, Figure 5.2 plots the l_2-projection error of the different operators in the 9-point stencil case. Again, the variable operators have approximately the same effects on each eigenvector, whereas the resulting ideal interpolation operator offers good approximation properties of the near-kernel space. As highlighted in (5.11), applying the left operator of the ideal formula removes the information contained in the range of $\hat{S}$ by minimizing an approximation error in A-norm. Even though such a norm does not exist in the indefinite case, ignoring this problem may still give interesting results in practice as depicted by the green curves. However, plugging the normal equations guarantees the minimization principle of the ideal interpolation

operator. We discuss this additional step in Section 5.2.3. In Figure 5.2, the orange curves depict a smaller error when resorting to normal equations.



(a) Classical setting R^T and S - $\alpha = 0.625^2$ (b) Classical setting R^T and S - $\alpha = 1.75^2$

Figure 5.2: Error of the l_2-projection onto the range of classical variable operators and ideal interpolation operator for the model problem SL2D-9S with and without normal equations

5.2.2 Ideal approximation based on the SPAI approach

One well-known practical issue of the ideal interpolation operator (5.9) is related to the generally dense inverse matrix of the fine block A_{ff} in (5.10). To circumvent this problem, an approximation based on sparsity constraints should be applied, as a good interpolation operator P should also count the fewer non-zero entries possible. A first approach to control its density is to compute a Sparse Approximate Inverse (SPAI) of A_{ff}. These SPAI techniques have originally been developed by Federickson and Brenson [41, 76]. In fact, they generally consist of computing the closest sparse approximation of a matrix inverse under constraints on the output pattern $\mathcal{P}$ of non-zero entries. The inverse approximation minimizes the quantity $||I - AM||_F$ subject to $\mathcal{P}$, where M corresponds to the constrained approximate inverse. In our case, we can compute an approximation of A_{ff}^{-1} using these SPAI techniques and inject it in (5.10). This approach can also be useful in the reduction setting as it also gives an approximation for M_f in (2.72), as in [84].

A second approach is to approximate the $n_f \times n_c$ matrix $A_{ff}^{-1} A_{fc}$ of (5.9) at once. This approach relies on the generalized SPAI technique developed [46] which minimizes the quantity of general form $||B - AW||_F$, where B and A respectively stand for the right member A_{fc} and A_{ff} in our case.

When resorting to SPAI techniques, the classical ideal interpolation operator can be approximated by

$$P = \begin{bmatrix} W \\ I_c \end{bmatrix}, \quad \text{with} \quad W := \begin{cases} \underset{\tilde{W}}{\arg\min} & ||A_{fc} - A_{ff}\tilde{W}||_F \\ \text{subject to} & \tilde{W}_{i,j} \in \mathcal{P} \end{cases}, \tag{5.12}$$

where $\mathcal{P}$ corresponds to the pattern of non-zero entries. We discuss the choice of the pattern further. Hence, the approximation based on (5.12) can give excellent results in the classical setting if A_{ff}^{-1} is also sparse or diagonally dominant. For instance,

we mentioned that using a red-black $\mathcal{C}/\mathcal{F}$ splitting on 5-point stencil matrices leads to a purely diagonal block A_{ff}. The interpolation operator constructed by way of (5.12) is ideal. More sophisticated $\mathcal{C}/\mathcal{F}$ splitting heuristics have been investigated to enable a convenient form of A_{ff} in the general case, as in [61, 83].

In practice, the computation of the sparse block W is made by solving a least-squares minimization problem for each column. The entire process is summarized by Algorithm 4 and more details can be found in [84].

Algorithm 4 Sparse Approximate Inverse for the construction of $\hat{P}$

for $j = 1, n_c$ **do**
  $\mathcal{J} \leftarrow \left\{ i \mid (i,j) \in \mathcal{P}_m \left(R_f^T \right) \right\}$
  $\mathcal{I} \leftarrow \left\{ i \mid (i,j) \in \mathcal{P} \left(\left[\hat{S}^T A \hat{S} \right]_{:,\mathcal{J}} \right) \right\}$
  $Q_j, R_j \leftarrow \text{QR Factorization} \left(\left[\hat{S}^T A \hat{S} \right]_{\mathcal{I},\mathcal{J}} \right)$
  $[W]_{\mathcal{I},j} \leftarrow R_j^{-1} Q_j^T \left[\hat{S}^T A \hat{R}^T \right]_{\mathcal{I},j}$
end for
return W

As the accuracy of P depends on the sparsity constraints enforced on W, we investigate the quality of approximation based on different augmented sparsity patterns. Hence, define m the number of new entries that are added on the top and bottom of each initial non-zero entry of a given pattern, as illustrated in Figure 5.3.

$$\begin{pmatrix} x & 0 & 0 & x & x \\ 0 & x & 0 & 0 & 0 \\ 0 & 0 & x & 0 & 0 \\ 0 & 0 & 0 & x & 0 \\ x & x & 0 & 0 & x \end{pmatrix}$$

(a) m = 0

$$\begin{pmatrix} x & x & 0 & x & x \\ x & x & x & x & x \\ 0 & x & x & x & 0 \\ x & x & x & x & x \\ x & x & 0 & x & x \end{pmatrix}$$

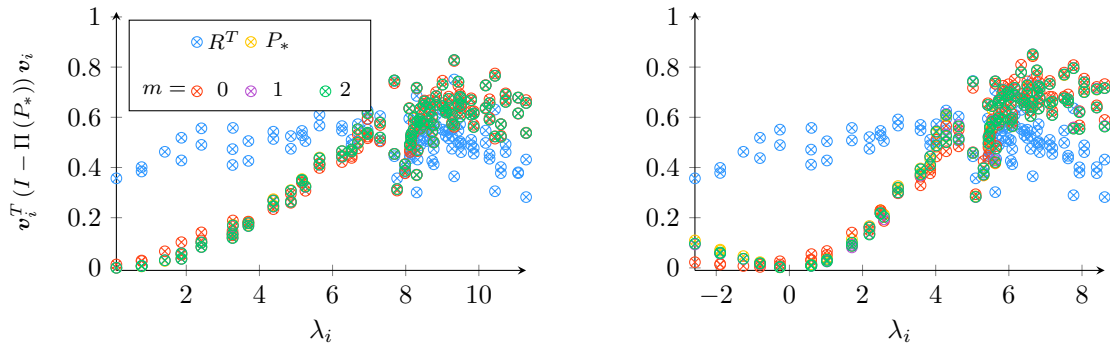
(b) m = 1

$$\begin{pmatrix} x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \end{pmatrix}$$

(c) m = 2

Figure 5.3: Augmentation of the pattern with respect to m

We control the sparsity pattern of W by augmenting the pattern of $A_{ff}A_{fc} + A_{fc}$ denoted by $\mathcal{P}_m(A_{ff}A_{fc} + A_{fc})$. In [84], the pattern $\mathcal{P}_0(A_{ff}A_{fc} + A_{fc})$ is used to construct the approximation of P_*. Figure 5.4 highlights the effect of approximation on the l_2-projection error. In particular, augmenting the pattern seems to have a strong impact, such that $m = 1$ is sufficient to provide an almost perfect approximation of the ideal interpolation. Lastly, we do not notice any significant difference between $m = 1$ and $m = 2$ in this case, certainly because P is already close to P_* for $m = 1$.



(a) Classical setting R^T and S - $\alpha = 0.625^2$ (b) Classical setting R^T and S - $\alpha = 1.75^2$

Figure 5.4: Error of the l_2-projection onto the range of the classical ideal approximations using the SPAI approach for the model problem SL2D-9S with respect to m - without normal equations

5.2.3 Normal Equations

In the SPD case, the construction of P_* results from the minimization of an error in A-norm (5.11). Such a norm does not exist in the indefinite case. Figure 5.4 shows that ignoring this concern can still provide good approximation properties of V_c in practice. However, we noticed a better convergence of our multilevel experiments in Chapter 7 when using the normal equations in the approximation of the ideal interpolation operator. Moreover, Figure 5.2 shows that it decreases the error of the l_2-projection. Hence, we also consider the normal equations in this chapter, such that the ideal interpolation operator has the form

$$P_* := \left(I - \Pi_{A^TA}(S)\right) R^T = \left(I - S\left(S^TA^TAS\right)^{-1} S^TA^TA\right) R^T. \tag{5.13}$$

This time, constructing each column of P_* is equivalent to solving the following minimization problems regardless of the indefinite nature of the initial matrix A, such that

$$[P_*]_{:,i} = R^T_{:,i} - \boldsymbol{s}_i \quad \text{with} \quad \boldsymbol{s}_i := \underset{\tilde{\boldsymbol{s}} \in \text{Range}(S)}{\arg\min} ||R^T_{:,i} - \tilde{\boldsymbol{s}}||_{A^TA} \quad i = 1, \ldots, n_c. \tag{5.14}$$

Assuming the same coarse-fine permutation of the initial matrix as in (2.76), the resulting normal equations matrix has the form

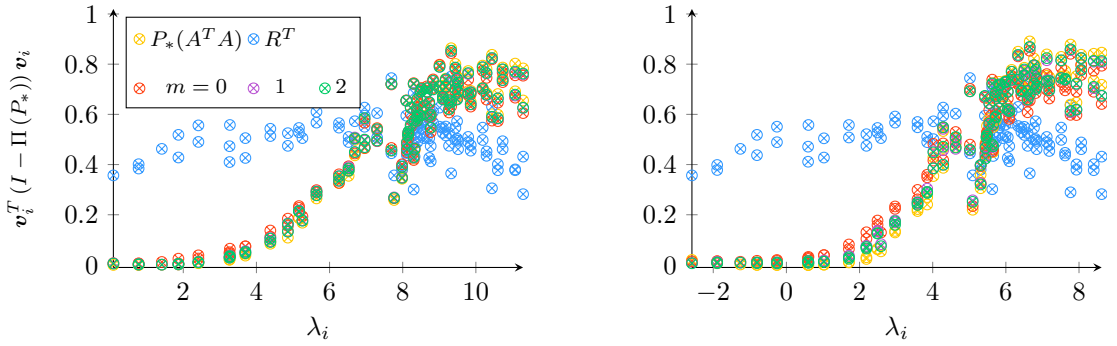
$$A^TA = \begin{bmatrix} A_{ff}^TA_{ff} + A_{cf}^TA_{cf} & A_{ff}^TA_{fc} + A_{cf}^TA_{cc} \\ A_{fc}^TA_{ff} + A_{cc}^TA_{cf} & A_{cc}^TA_{cc} + A_{fc}^TA_{fc} \end{bmatrix}. \tag{5.15}$$

Substituting the normal equations matrix (5.15) in the former minimization problem (5.12) gives

$$W := \begin{cases} \underset{\tilde{W}}{\arg\min} & ||A_{ff}^T\left(A_{fc} - A_{ff}\tilde{W}\right) + A_{cf}^T\left(A_{cc} - A_{cf}\tilde{W}\right)||_F \\ \text{subject to} & \tilde{W}_{i,j} \in \mathcal{P} \end{cases}. \tag{5.16}$$

where W is the fine variable block of interpolation in P. Equation (5.16) reveals that approximating the classical ideal interpolation operator with normal equations is

equivalent to solving two minimization problems of the form (5.12) simultaneously. Here again, sub-matrices involved in the normal equation minimization problem (5.16) are nicely structured as in the original minimization problem (5.12). Although the structure of A^TA is probably more tricky, it may be reasonable to expect an appropriate sparse block W with a convenient the $\mathcal{C}/\mathcal{F}$ splitting. Figure 5.5 plots the l_2-projection error of each eigenvector using the SPAI approximation with normal equations. The sparsity patterns enforced on W are the same as in Figure 5.4.



(a) Classical setting R^T and S - $\alpha = 0.625^2$

(b) Classical setting R^T and S - $\alpha = 1.75^2$

Figure 5.5: Error of the l_2-projection onto the range of the classical and least-squares ideal approximations with the SPAI approach for the model problem SL2D-9S with respect to τ - with normal equations

While 5.2 shows that the l_2-projection error of the ideal interpolation P_* is better with the normal equations, Figure 5.5 reveals that using normal equations yields a stronger difference between the approximation P and the ideal interpolation operator P_*. In fact, the structure of A^TA is more difficult and it is not surprising that the approximate inverse return by solving (5.16) is subsequently less accurate.

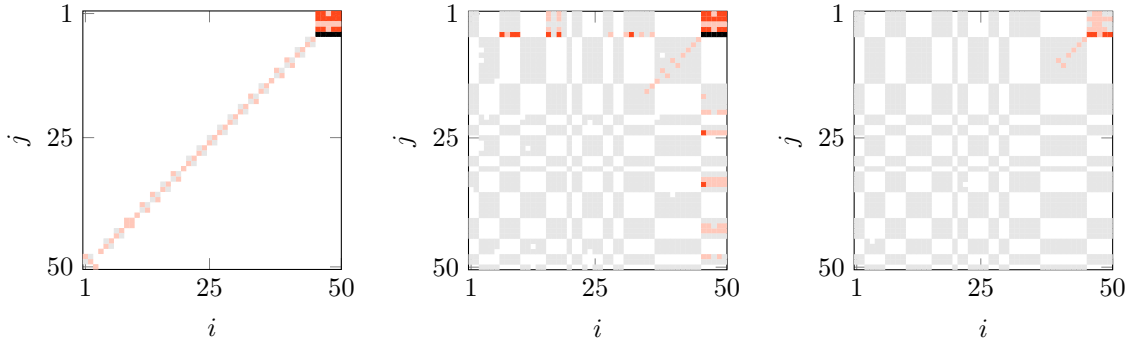
5.2.4 Effect on the pollution

In this section, we discuss the effect of the ideal approximation step through the concept of pollution introduced in Section 3.2. In Section 3.2.2, we have seen that the pollution associated with large eigenvalues has the strongest impact on the coarse correction, especially for eigenvectors associated with small eigenvalues in magnitude. Moreover, the contraction rate depends on a mix between negative and eigenvalues through the block of pollution K_f. In what follows, we analyze what type of pollution the ideal approximation phase decreases promptly. Throughout this section, the target ideal interpolation has normal equations.

5.2.4.1 The 5-point stencil case

Let us first analyze the pollution K_f when applying the previous approach to the 5-point stencil problem (1.2). Figure 5.6 represents the entries of K_f with respect to the pattern augmentation degree m and for the shift $\alpha = 0.625^2$. Alternatively, Figure 5.7 corresponds to the shift $\alpha = 1.75^2$. Both Figures 5.6a and 5.7a represent the entries of K_f when no approximation of the ideal interpolation operator is applied, such that $P = R^T$. Black pixels designate max amplitude entries, red pixels

correspond to intermediate amplitudes (between 10% and 1% of the largest amplitude), pink pixels represent small amplitudes (between 1% and 0.1%), and gray pixels denote extremely small entries (below 0.1%). Naturally, white pixels represent strictly zero entries. The x-axis corresponds to the indexes of V_c, whereas the y-axis corresponds to the indexes of V_f. For instance, the entry (j, i) of K_f in (3.70) gives the pollution arising from the large eigenvector $\boldsymbol{v}_j \in V_f$ in the l_2-projection of the small eigenvector $\boldsymbol{v}_i \in V_c$. Note that, because we use normal equations, P is an approximation of P_*, even in the 5-point stencil case.



(a) R^T (b) P, $m = 0$ (c) P, $m = 1$

Figure 5.6: Entries of the pollution block K_f with respect to τ and for $\alpha = 0.625^2$ - The 5-point stencil case

First, the diagonal shape of K_f in Figure 5.6a reveals that the coarse variable operator R^T interacts in a very particular way with the small eigenvectors of 5-point stencil matrix. Large and small eigenvectors are probably linked by a relation of the form (3.10) in the two-dimensional problem as well.

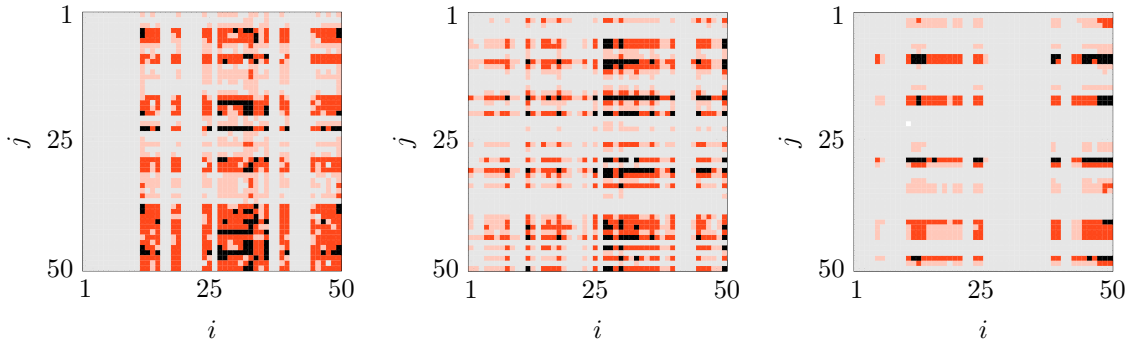
Beyond that point, Figure 5.6 shows that approximating the ideal interpolation operator decreases the number of pink pixels. More precisely, the pollution of the smallest eigenvectors is eliminated when setting m to 0 although a few red pixels appear on the top. The red pixels are cleaned by increasing m to 1, except on the top right-hand corner where the cluster of red pixels remains.

The evolution of the pollution is more difficult to clarify for the larger shift $\alpha = 1.75^2$. For sure, setting $m = 1$ decreases the pollution entries globally, without a clear distinction between the pollution entries associated with small and large eigenvectors of V_f however.

5.2.4.2 The 9-point stencil case

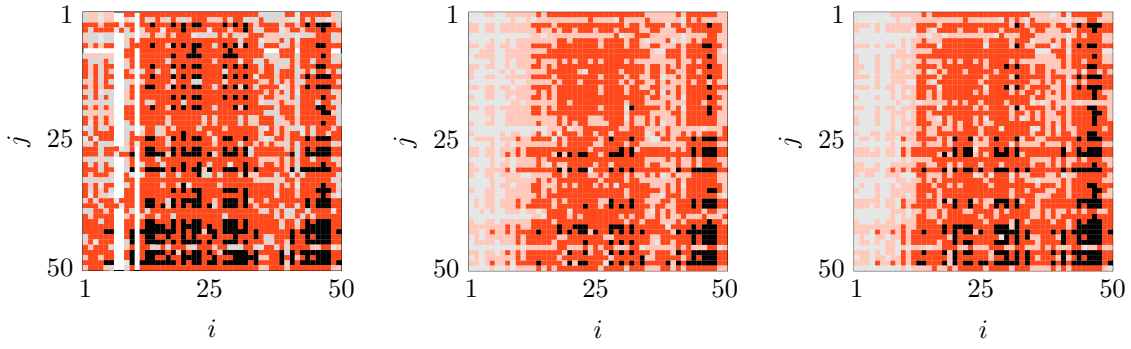
Let us repeat the same experiments on the 9-point stencil model problem (1.3).

In both Figures 5.8 and 5.9, we observe that improving the approximation of P_* tends to decrease the pollution of the smallest eigenvectors first. The most polluted area for $m = 1$ corresponds to the largest eigenvectors of V_c.



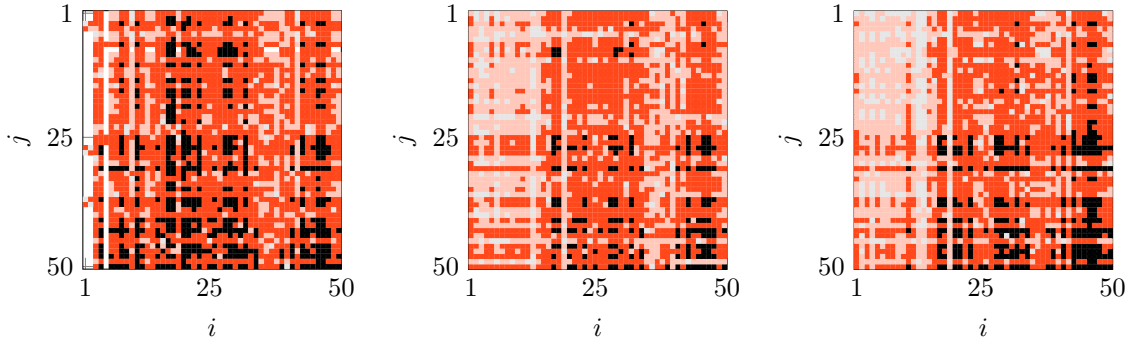
(a) R^T (b) P, $m = 0$ (c) P, $m = 1$

Figure 5.7: Entries of the pollution block K_f with respect to m and for $\alpha = 1.75^2$ - The 5-point stencil case



(a) R^T (b) P, $m = 0$ (c) P, $m = 1$

Figure 5.8: Entries of the pollution block K_f with respect to m and for $\alpha = 0.625^2$ - The 9-point stencil case



(a) R^T (b) P, $m = 0$ (c) P, $m = 1$

Figure 5.9: Entries of the pollution block K_f with respect to m and for $\alpha = 1.75^2$ - The 9-point stencil case

We conclude this section by noting that approximating the ideal interpolation operator tends to decrease the pollution of the smallest eigenvectors of V_c first. This feature is especially important as the smallest eigenvectors are the most sensitive to the pollution, for the reasons explained in Section 3.2.2.

5.2.5 Complexity

While approximating the ideal interpolation operator improves the coarse variable operator by decreasing its pollution, a trade-off has to be found with the resulting fill-in of the coarse grid operator. In what follows, we only study the sparsity pattern of the interpolation operators and of the coarse matrices for $\alpha = 0.625^2$ as increasing the shift do not change the complexity significantly.

5.2.5.1 Complexity of the interpolation operator

stencil	R^T	$m=0$	$m=1$	$m=2$	P_*
5-pts	0.5	2.3	2.3	2.3	2.3
9-pts	0.5	3.8	6.6	8.6	13.8

(a) Without normal equations

R^T	$m=0$	$m=1$	$m=2$	P_*
0.5	2.3	4.8	7.4	25.5
0.5	3.8	6.8	8.8	24.8

(b) With normal equations

Table 5.1: Average number of non-zero entries per row with respect to m when approximating the ideal interpolation operator with the SPAI approach

Naturally, Table 5.1 shows that increasing m leads to more non-zero entries in P. Moreover, the ideal interpolation operator P_* with normal equations is denser. However, we used the same sparsity pattern to control the fill-in of P when approximating P_* with or without normal equations. As a consequence, the sparsity of P remains the same in both cases. Extending the pattern of non-zero entries when the target ideal interpolation has normal equations should improve P, but at the cost of higher complexity.

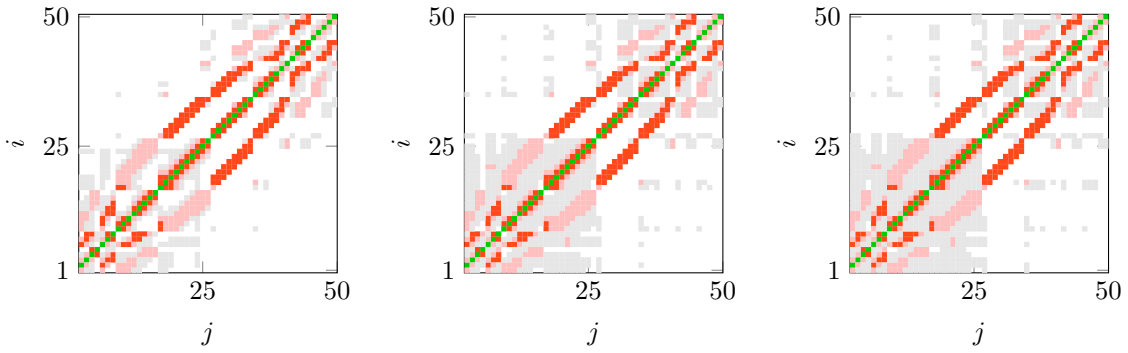
5.2.5.2 Complexity of the coarse grid

Recall that $A_c = P^T AP$ is the coarse grid operator computed through the Galerkin triple matrix product, as introduced in (2.41). Dark green and red pixels in Figure 5.10 respectively designate positive and negative entries whose magnitude are greater than 10% of the largest magnitude entry. Light green and pink pixels respectively designate positive and negative entries with magnitude between 10% and 1% of the largest. Gray pixels designate remaining non-zero entries, regardless of their signs.

In accordance with Table 5.1, both figures show that the density increases with m. Nevertheless, the loss of sparsity is mostly due to the appearance of very near-zero entries represented in gray, whose magnitude are lower than $10^{-3} \times ||P||_\infty$. This feature hints that an additional sparsification approach may be applicable to reduce the complexity of the coarse matrices.

5.3 Least-squares minimization framework

The process of approximating P_* is generally difficult to predict, because the classic variable operators R^T and S are not approximations of the theoretical complementarity required. In this section, we design a least-squares coarse variable operator



(a) $m = 0$ (b) $m = 1$ (c) $m = 2$

Figure 5.10: Fill-in of the coarse matrix A_c with respect to m for the 9-point stencil problem and with normal equations

denoted by $\hat{R}^T$ that targets the potentially oscillatory set of small eigenvectors V_c. As we assume no available information on the near-kernel space, we approximate it by way of a set of test vectors denoted by T. As prescribed by the theory, we thereafter define the fine variable operator $\hat{S}$ to satisfy the l_2-orthogonality with $\hat{R}^T$. The least-squares variable operators $\hat{R}^T$ and $\hat{S}$ respectively target V_c and V_f and enable a new and more predictable approximation $\hat{P}$ of the ideal interpolation operator $\hat{P}_*$.

In this second framework, we design the ideal approximation $\hat{P}$ by restricting the minimization space in (5.11) to a few columns of $\hat{S}$ only. Contrary to the classic ideal approximation P that works better when the fine inverse block has a convenient structure, $\hat{P}$ is designed by removing smoother related information contained in the columns of $\hat{S}$ from the range of $\hat{R}^T$ (which initially approximates the span of V_c through the set T). This approach improves the complementarity between the smoother and the coarse correction.

5.3.1 Approximating V_c by a set of test vectors

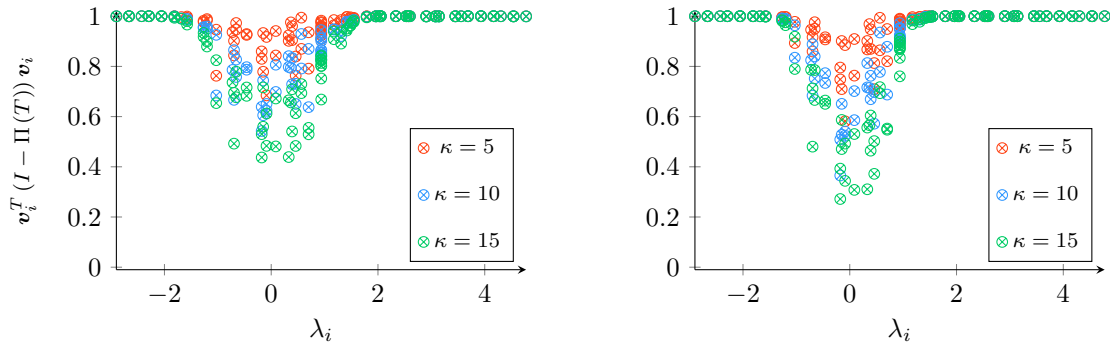
In a pure algebraic setting, the near-kernel space of the Helmholtz equation is unknown. Using the smallest eigenvectors V_c to construct the optimal interpolation is not practical, as it requires computing the n_c smallest eigenvectors exactly. Instead, we compute a cheaper approximation of V_c by way of a set of vectors generated by smoothing a number κ of initially random vectors.

We approximate the oscillatory and potentially large near-kernel space by smoothing a set of random vectors with the polynomial smoother developed in Chapter 4. We recall that the error propagation matrix of the Chebyshev polynomial smoother is $q_{d+1}^{\nu}(A^2)$. Define the columns of T by a set of κ smoothed random vectors $\boldsymbol{z}_l$, such that

$$T_{:,l} = q_{d+1}^{\nu}(A^2)\boldsymbol{z}_l \quad \text{with} \quad \boldsymbol{z}_l = \mathsf{random}(n) \quad , \quad l = 1, \ldots, \kappa, \tag{5.17}$$

where $T_{:,l}$ designates the lth column of T (i.e., the lth test vector). Naturally, increasing κ enlarges the set, whereas adding more smoothing steps tends to ac-

centuate the prevalence of small eigenvectors in the span of T as the polynomial smoother damps the largest ones faster.



(a) $\nu = 2$ (b) $\nu = 4$

Figure 5.11: Error of the l_2-projection onto range(T) with respect to κ and ν with $\alpha = 1.75^2$

The Figure 5.11 plots the error between each eigenvector and its l_2-projection (3.69) in the range of T with respect to the number κ of test vectors and the number ν of smoothing steps. As expected, the layering of red, blue, and green marks shows that increasing κ improves the approximation of the smallest eigenvectors. Additionally, adding more smoothing steps emphasizes the smallest eigenvectors. Marks associated with large eigenvectors are very close to one in both figures 5.11a and 5.11b, which means that T is quasi-orthogonal with V_f. Even though a small number of test vectors cannot represent the entire and potentially large near-kernel space accurately, the subspace still provides a rough approximation of the smallest eigenvectors of V_c and excludes the largest ones. These features will be helpful in the construction of appropriate interpolation rules for Helmholtz.

5.3.2 Construction of the least-squares variable operators

The set T of smoothed random vectors forms an approximation of the vector space that good interpolation rules should produce in theory. Hence, we construct the new coarse variable operator $\hat{R}^T$ by applying a least-squares minimization strategy on it. This approach starts by initializing the set of points $\mathcal{C}/\mathcal{F}$. We denote by n_c and n_f their respective sizes. For Helmholtz, such a dichotomy is inspired by the classical strength of connection rule (2.56) that originally tracks geometrically smooth information. Because the near-kernel space of Helmholtz is potentially oscillatory, strong connection groups $\mathcal{S}_i$ are only constructed to evaluate the importance of each point to the others based on the magnitude of matrix entries. The points strongly connected to the ith point belong to the group (5.7). From these strong connection groups (5.7), the selection of $\mathcal{C}$ remains the same as in the classical AMG set-up phase.

For what follows, let us now assume rows of T are permuted in terms of coarse and fine points, such that

$$T = \begin{bmatrix} T_f \\ T_c \end{bmatrix}. \tag{5.18}$$

In theory, the set of test vectors T is exactly in the range of the coarse variable operator $\hat{R}^T$ if there exists a set of coarse vectors $T_c^{'}$ of size $n_c \times \kappa$ that satisfies

$$T = \hat{R}^T T_c^{'}. \tag{5.19}$$

This property is for instance enforced in the smoothed aggregation method [75, 74, 12] by splitting the set between disjoint aggregates over the entire domain to initiate a tentative block interpolation operator. In most applications targeted by smoothed aggregation methods, the near-kernel space is known in advance and can be plugged into T directly. In our case, the set of vectors only provides a rough approximation of the near-kernel space, so that enforcing the strict equality of (5.19) is not a requirement. Moreover, the size κ of the set is arbitrary and potentially too large for a practical implementation of the smoothed aggregation method.

Subsequently, we opt for a least-squares minimization strategy that will construct each row of the new coarse variable operator $\hat{R}^T$ as the best average interpolation rules for T. The following development is inspired from [7]. Ordering the points based on their affiliation to the groups of $\mathcal{C}$-points and $\mathcal{F}$-points, a good form of coarse variable operator is

$$\hat{R}^T = \begin{bmatrix} R_f^T \\ I_c \end{bmatrix}. \tag{5.20}$$

The $\mathcal{C}$-points are interpolated to the finer level by way of a simple injection rule corresponding to the $n_c \times n_c$ identity block I_c. The block R_f^T designates the interpolation rules of $\mathcal{F}$-points. From (5.20), the interpolation of $\mathcal{C}$-points with the coarse variable operator naturally satisfies

$$T_c = I_c T_c. \tag{5.21}$$

Equation (5.21) highlights that interpolation of $\mathcal{C}$-points is exact, due to the identity block I_c. Our approach consists of defining a sparse and relevant block R_f^T in $\hat{R}^T$ that minimizes the interpolation error of $\mathcal{F}$-points in the set T. A good coarse variable operator for the propagation of the set of test vectors T satisfies

$$T_f \approx R_f^T T_c. \tag{5.22}$$

Accordingly, we seek a practical coarse variable operator that minimizes the Euclidean distance between the T_f and $R_f^T T_c$ under the constraint of a non-zero pattern on the block R_f^T.

To this end, let i be an $\mathcal{F}$-point and $\hat{\boldsymbol{r}}_i$ be the vector containing the non-zero elements of the ith row of $\hat{R}^T$. The idea is to construct each $\mathcal{F}$-point interpolation rule by minimizing the squared difference between $\mathcal{F}$-values of T and their interpolation from strongly connected $\mathcal{C}$-points designated by $\mathcal{C}_i$, such that

$$\mathcal{C}_i := \mathcal{S}_i \cap \mathcal{C}. \tag{5.23}$$

Subsequently, the non-zero entries in the ith row of the coarse variable operator are given by the indexes of the set $\mathcal{C}_i$. Denote by $T_{i,:}$ a row vector containing the ith

values of each test vector, and $T_{\mathcal{C}_i,l}$ a vector containing the values in $T_{:,l}$ of the $\mathcal{C}_i$ in (5.23). Then, we define the ith row of the coarse variable operator by

$$\forall i \in \mathcal{F} \quad , \quad \hat{\boldsymbol{r}}_i = \underset{\hat{\boldsymbol{r}} \in \mathbb{C}^{\mathrm{card}(\mathcal{C}_i)}}{\arg\min} \sum_{l=1}^{\kappa} w_l \left(T_{i,l} - \hat{\boldsymbol{r}} \cdot T_{\mathcal{C}_i,l}\right)^2 =: \underset{\hat{\boldsymbol{r}} \in \mathbb{C}^{\mathrm{card}(\mathcal{C}_i)}}{\arg\min} \mathcal{L}_i(\hat{\boldsymbol{r}}), \tag{5.24}$$

where w_l are scaling weights. For instance, good weights are $w_l = 1/|\lambda_l|$ if T contains eigenvectors. In our case, the contribution of eigenvectors is statistically the same for each column of T, as they arise from the same smoothing process of initially random vectors. In what follows, we fix $w_l = 1$ in (5.24). Finding the minimum of the convex loss function $\mathcal{L}_i$ is equivalent to solving

$$\nabla \mathcal{L}_i(\hat{\boldsymbol{r}}_i) = 0. \tag{5.25}$$

Equation (5.25) can be rewritten element-wise

$$\frac{\partial \mathcal{L}_i(\hat{\boldsymbol{r}}_i)}{\partial \hat{\boldsymbol{r}}_{ij}} = \sum_{l=1}^{\kappa} 2w_l(T_{i,l} - \hat{\boldsymbol{r}}_i \cdot T_{\mathcal{C}_i,l}) T_{\mathcal{C}_{ij},l} = 0 \quad , \quad j = 1, \ldots, \mathrm{card}(\mathcal{C}_i). \tag{5.26}$$

Finally, (5.26) leads to a system of linear equations to solve for each $\mathcal{F}$-point

$$\hat{\boldsymbol{r}}_i T_{\mathcal{C}_i} W T_{\mathcal{C}_i}^T = T_i W T_{\mathcal{C}_i}^T, \tag{5.27}$$

where the non-zero elements of the ith row of $\hat{R}^T$ are contained in the solution $\hat{\boldsymbol{r}}_i$. The matrix is full rank and the solution in (5.27) is unique if we have at least $\kappa = \max_i \{\mathrm{Card}(\mathcal{C}_i)\}$ locally linearly independent test vectors. Even if it is statistically always the case when starting from random candidate vectors, the matrix singularity can be detected during the factorization of $T_{\mathcal{C}_i} W T_{\mathcal{C}_i}^T$. In that special case, a pseudo-inverse can be computed to find an optimal solution in the least squares sense. For what follows, we set the maximal size of $\mathcal{C}_i$ to 4 to control the sparsity of $\hat{R}^T$.

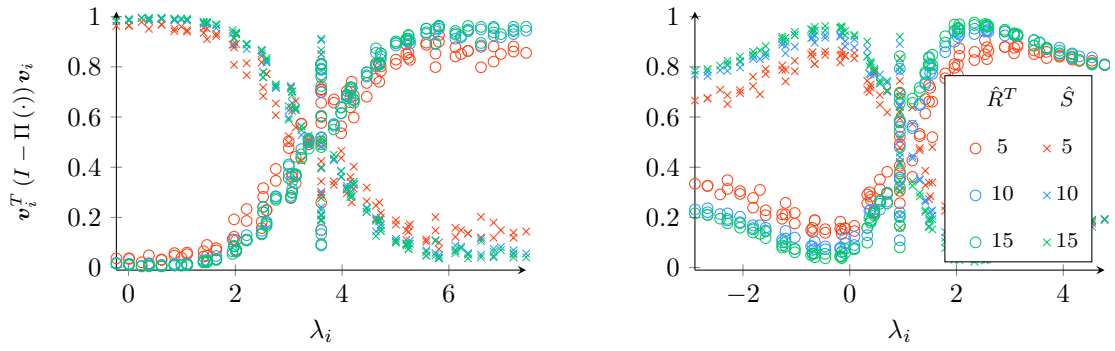
As detailed previously, the ideal interpolation operator is constructed by projecting-out the $\hat{S}$-related information that should be handled by the smoother from the range of $\hat{R}^T$. This has the effect to maximize the complementarity between the smoother and the coarse grid correction that depends on $\hat{P}$. Now that we have a least-squares coarse variable operator $\hat{R}^T$ for Helmholtz, the form of $\hat{S}$ has to be defined. Following the theory [38], the least-squares coarse and fine variable operators should satisfy

$$\mathrm{range}\left(\hat{R}^T\right) \oplus \mathrm{range}\left(\hat{S}^T\right) = n_c + n_f = n \qquad \text{and} \qquad \hat{R}\hat{S} = 0. \tag{5.28}$$

In fact, there exists an infinite number of fine variable operators $\hat{S}$ that satisfies (5.28). However, the resulting ideal interpolation operator remains the same regardless of the form of $\hat{S}$ as long as (5.28) is satisfied. Subsequently, one straightforward l_2-orthogonal set of variable operators is

$$\hat{R}^T = \begin{bmatrix} R_f^T \\ I_c \end{bmatrix} \quad \text{and} \quad \hat{S} = \begin{bmatrix} I_f \\ -R_f \end{bmatrix}, \quad \text{such that} \quad \hat{R}\hat{S} = \begin{bmatrix} R_f & I_c \end{bmatrix} \begin{bmatrix} I_f \\ -R_f \end{bmatrix} = 0. \tag{5.29}$$

Recall that the set T is composed of small eigenvectors of V_c due to the Chebyshev polynomial smoother that primarily damps V_f and $\hat{R}^T$ is designed to interpolate it correctly. By orthogonality, the space spanned by $\hat{S}$ is mostly composed of large eigenvectors that are easily damped by the smoother. Therefore, the aim of the ideal framework in this oscillatory context is to improve the coarse variable operator by removing the information related to these large eigenvectors, as they are already captured by the Chebyshev polynomial smoother of Chapter 4. The Figure 5.12 illustrates the interaction of eigenvectors in the range of $\hat{R}^T$ (in blue) and $\hat{S}$ (in red) with respect to the number κ of test vectors in T. This figure can be compared to the analogous Figure 5.1 that plots the same quantity but for the classical variable operators defined in (5.8).



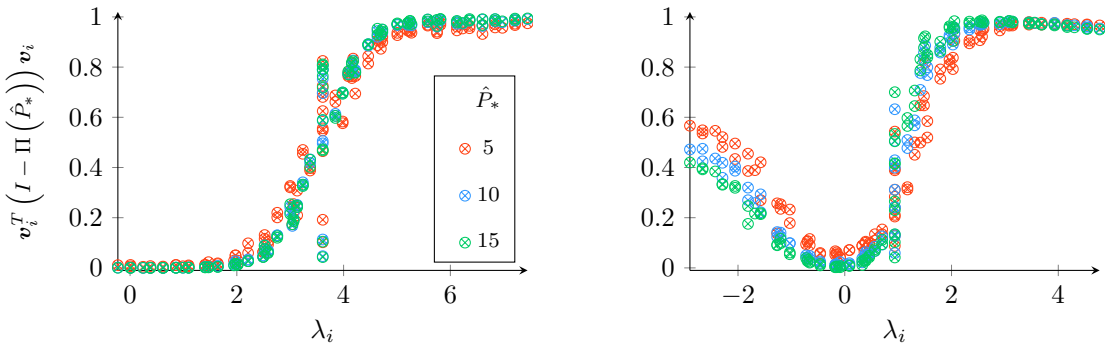
(a) $\alpha = 0.625^2$ (b) $\alpha = 1.75^2$

Figure 5.12: Error of the l_2-projection onto the range of the least-squares variable operators $\hat{R}^T$ and $\hat{S}$ with respect to the number κ of test vectors

By contrast to the classical setting, Figure 5.12 shows that the range of $\hat{R}^T$ contains a good approximation of the smallest eigenvectors of V_c. Whereas the prevalence of small and large eigenvectors are the same in the range of the classical coarse variable operator R^T, the least-squares counterpart $\hat{R}^T$ gives a much better approximation of V_c. Conversely, the fine variable operator has good approximation properties for large eigenvectors of V_f. A last and expected observation is that the interpolation error of $\hat{R}^T$ for small eigenvectors of V_c decreases as more test vectors are added in T. The same observation applies for $\hat{S}$ with the large eigenvectors of V_f.

Let $\hat{P}_*$ be the ideal interpolation operator resulting from the least-squares variable operators $\hat{R}^T$ and $\hat{S}$. Figure 5.13 represents the error of interpolation of $\hat{P}_*$ with respect to the number κ of test vectors in T.

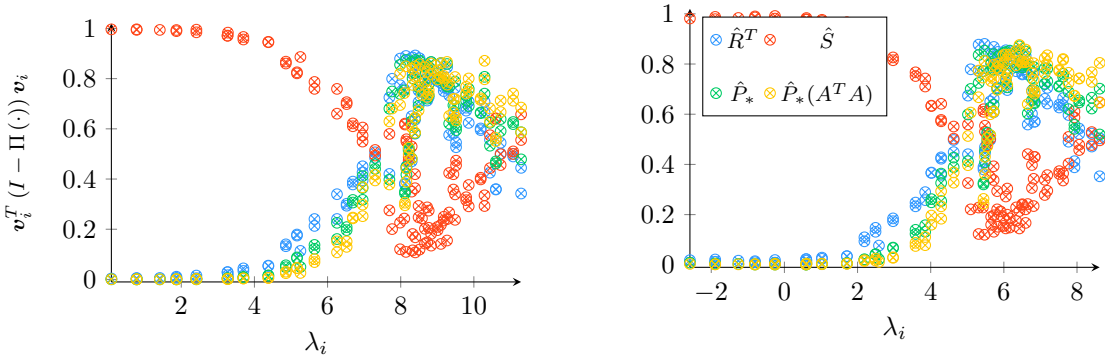
First, note that the error for small eigenvectors decreases with κ, as it does for $\hat{R}^T$ in Figure 5.12 as well. Analogous to Figure 5.1 depicting the error of P_*, the least-squares ideal interpolation operator $\hat{P}_*$ appears slightly less accurate for the smallest eigenvalue. This inconvenience is due to the large eigenvectors that interact with $\hat{R}^T$ for small values of κ but tends to vanish as more test vectors are generated. In all cases, intermediate small positive eigenvectors are better approximated and large negative ones are better ignored.



(a) $\alpha = 0.625^2$ (b) $\alpha = 1.75^2$

Figure 5.13: Error of the l_2-projection onto the range of the least-squares ideal interpolation operator $\hat{P}_*$ with respect to the number κ of test vectors

In what follows, the number of test vectors used in the construction of $\hat{R}^T$ is always fixed to $\kappa = 15$. Let us also recall that a red-black selection of $\mathcal{C}/\mathcal{F}$-points to 5-point stencil matrices leads to a diagonal block A_{ff} in the classical case. Therefore, we also considered the 9-point stencil problem (1.3) to challenge our methods. Accordingly, Figure 5.14 plots the l_2-projection error of eigenvectors for $\hat{R}^T$, $\hat{S}$, and $\hat{P}_*$ for two shifted 9-point stencil matrices.



(a) Least-squares setting $\hat{R}^T$ and $\hat{S}$ - $\alpha = 0.625^2$ (b) Least-squares setting $\hat{R}^T$ and $\hat{S}$ - $\alpha = 1.75^2$

Figure 5.14: Error of the l_2-projection onto the range of least-squares variable operators and ideal interpolation operator for the model problem SL2D-9S with and without normal equations

As discussed previously, the classical variable operators (5.8) ignore eigenvalues associated with eigenvectors. This observation appears distinctly in Figure 5.2. Despite this observation, P_* still minimizes the quantity (2.65), and therefore appears as a good interpolation operator for the smallest eigenvector in Figure 5.2. However, the error grows fast as the magnitude of the eigenvalue increases, even if the associated eigenvector belongs to V_c in general.

Alternatively in Figure 5.14, we observe that the least-squares variable operators have a special treatment for each eigenvector, such that the range of $\hat{R}^T$ approximates the span of V_c better and the range of $\hat{S}$ approximates the span of V_f better. It follows that the least-squares ideal interpolation operator $\hat{P}_*$ is not only good for

the smallest eigenvector, but for all the smallest eigenvectors of V_c.

In other words, both classical and least-squares ideal interpolation operators seem to be good interpolation operators of the smallest eigenvector, but $\hat{P}_*$ appears more efficient for a broader portion of V_c. This feature is obvious when comparing both figures 5.2b and 5.14b around $|\lambda_i| \approx 2$ for instance. Lastly, orange marks represent the error when using normal equations in the definition of ideal interpolation. Comparing Figure 5.14 with 5.2, it seems that normal equations benefits P_* better than $\hat{P}_*$ in this case.

5.3.3 Ideal approximation based on the subspace restriction approach

The SPAI approach introduced in Section 5.2.2 benefits from the underlying structure of the fine block. In this least-squares framework, it is not clear if the fine block $\hat{S}^T A \hat{S}$ has convenient structure that could help it. Moreover, the least-squares coarse and fine variable operators are designed to designate what the coarse correction and the smoother should respectively capture. Restricting the search space of the minimization problem (5.11) to a few columns of $\hat{S}$ still provides an ideal approximation $\hat{P}$ of $\hat{P}_*$ aimed at improving the complementarity between the coarse correction and the smoother. In fact, constructing the approximation $\hat{P}$ by restricting the columns of $\hat{S}$ is equivalent to computing the ideal $\hat{P}_*$ but with less fine-related information. Additionally, the fine variable operator $\hat{S}$ is sparse. Subsequently, restricting the search space of $\hat{S}$ controls the sparsity of the ideal approximation $\hat{P}$.

In that way, let X_j be the injection operator of ones and zeros of size $n_f \times n_j$ that selects n_j columns of $\hat{S}$, $\hat{S}X_j$, and such that $n_j \leq n_f$. Also, let $\mathcal{I}$ be the set of column indexes for which X_j contains a one entry. It follows that applying the operator X_j to $\hat{S}$ gives

$$\hat{S}X_j = \left[\hat{S}\right]_{:,\mathcal{I}}. \tag{5.30}$$

As shown in Algorithm 5, it is not necessary to form the injection matrix X_j explicitly in practice. Nevertheless, we use this term for ease of notation and to better connect the approximation with the minimization principle that drives this approach. In fact, the only difference between the exact ideal interpolation and its approximation arises from the restriction of the search space given by X_j. Note that the restricted search space is still l_2-orthogonal with the least-squares coarse variable operator, such that

$$\hat{R}\hat{S}X_j = 0, \tag{5.31}$$

as required by the ideal theory. Similarly to (5.11), let $\boldsymbol{s}_j$ be defined by

$$\boldsymbol{s}_j := \hat{S}X_j \left(X_j^T \hat{S}^T A \hat{S} X_j\right)^{-1} X_j^T \hat{S}^T A \hat{R}_{:,j}^T. \tag{5.32}$$

Since indefinite matrices do not generate a norm, we are not guaranteed a minimization property for $\boldsymbol{s}_j$. The effect of injecting the normal equations matrix in (5.34) will be discussed in Section 5.3.4. That said, the columns of the least-squares ideal approximation are computed by

$$\hat{P}_{:,j} = \hat{R}_{:,j}^T - \boldsymbol{s}_j = \hat{R}_{:,j}^T - \hat{S}X_j\boldsymbol{\rho}_j, \tag{5.33}$$

where $\boldsymbol{\rho}_j$ is the solution to the $n_j \times n_j$ linear system

$$X_j^T \hat{S}^T A \hat{S} X_j \boldsymbol{\rho}_j = X_j^T \hat{S}^T A \hat{R}_{:,j}^T. \tag{5.34}$$

In this subspace restriction approach, the trade-off between sparsity and accuracy is enabled by choosing the columns of $\hat{S}$ based on the entries of $\hat{S}^T A \hat{R}_{:,j}^T$. In fact, each entry corresponds to the A-inner product between a column of the fine variable operator and the jth column of the coarse variable operator. A large entry designates a column of $\hat{S}$ that contributes a lot to the solution to the problem (5.32). The column selection phase iterates until the entries associated with the selected columns represent a percentage τ of the entire set of non-zero entries. At each iteration, the column of $\hat{S}$ associated with the largest entry of $\hat{S}^T A \hat{R}_{:,j}^T$ is selected, which is equivalent to extending X_j with the Euclidean basis vector with one at the index of the chosen column and zeros elsewhere. Because the columns with the largest entries in $\hat{S}^T A \hat{R}_{:,j}^T$ are selected first, the set of selected columns is the smallest set that satisfies

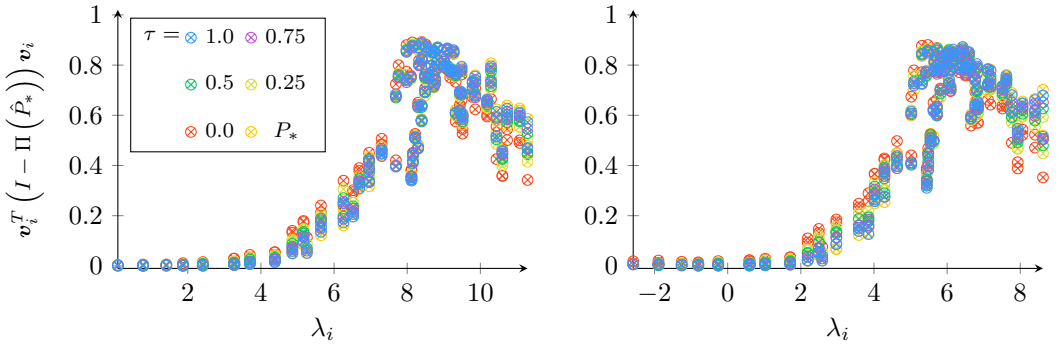
$$||X_j \hat{S}^T A \hat{R}_{:,j}^T||_2^2 \geq \tau \times ||\hat{S}^T A \hat{R}_{:,j}^T||_2^2, \quad \text{with} \quad \tau \in [0, 1]\,. \tag{5.35}$$

When τ increases, more and more $\hat{S}$-related information of V_f that the smoother already damps is removed from the range of $\hat{R}^T$, such that $\hat{P}$ becomes a better interpolation operator for what should be treated on the coarse level. Although setting $\tau = 1$ selects all the columns associated with the non-zeros of $\hat{S}^T A \hat{R}_{:,j}^T$, the remaining columns associated with zero entries are omitted. As a consequence, the matrix $X_j^T \hat{S}^T A \hat{S} X_j$ still corresponds to a principle sub-matrix of $\hat{S}^T A \hat{S}$. This second ideal approximation approach is recapped in Algorithm 5.

Algorithm 5 Subspace restriction approach to approximate the ideal interpolation operator

1: **for** $j = 1, n_c$ **do**
2: $\quad \mathcal{I} \leftarrow$ Sort Rows Ascending Order $\left(\hat{S}^T A \hat{R}_{:,j}^T\right)$
3: $\quad$ **While** $||\hat{S}^T A \hat{R}_{\mathcal{I},j}||_2 > \tau \times ||\hat{S}^T A \hat{R}_{:,j}||_2$ **do**
4: $\quad\quad \mathcal{I} \leftarrow \mathcal{I} \setminus \{\mathcal{I}_1\}$ $\quad \triangleright\ \mathcal{I}_1$ contains the smallest entry in $\hat{S}^T A \hat{R}_{\mathcal{I},j}$
5: $\quad$ **end while**
6: $\quad \rho_j \leftarrow \left(\left[\hat{S}^T A \hat{S}\right]_{\mathcal{I},\mathcal{I}}\right)^{-1} \cdot \left[\hat{S}^T A \hat{R}_{:,j}^T\right]_{\mathcal{I}}$
7: $\quad \left[\hat{P}\right]_{:,j} \leftarrow \hat{R}_{:,j}^T - [S]_{:,\mathcal{I}} \cdot \rho_j$
8: **end for**
9: **return** $\hat{P}$

Figure 5.15 indicates increasingly better approximation properties of $\hat{P}$ as τ grows. We note how close $\hat{P}$ for $\tau = 1.0$ is from the ideal interpolation operator $\hat{P}_*$. The complexity of these interpolation operators are discussed in Section 5.3.6.1.



(a) Least-squares setting R^T and S - $\alpha = 0.625^2$ (b) Least-squares setting R^T and S - $\alpha = 1.75^2$

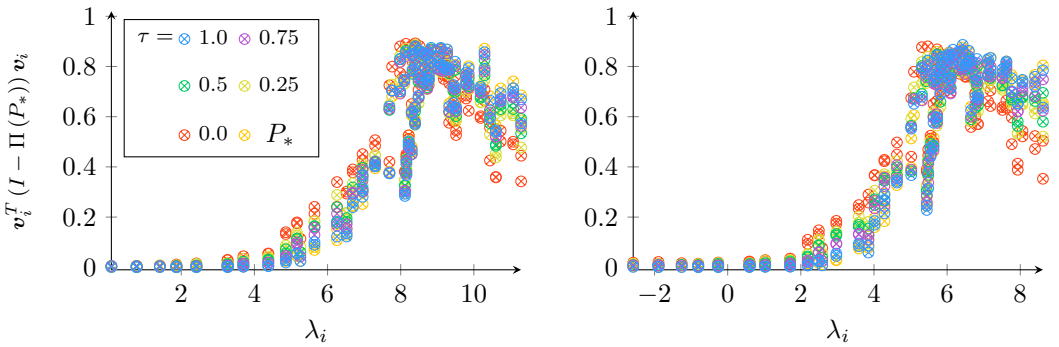
Figure 5.15: Error of the l_2-projection onto the range of the least-squares ideal approximations with the subspace restriction approach for the model problem SL2D-9S with respect to τ - without normal equations

5.3.4 Normal equations

Injecting normal equations in this ideal approximation approach ensures that the resulting ideal approximation $\hat{P}$ satisfies minimization properties, such that injecting A^TA in (5.32) leads to

$$\boldsymbol{s}_j := \underset{\tilde{\boldsymbol{s}}\in\text{Range}(\hat{S}X_j)}{\arg\min} ||\hat{R}^T_{:,j} - \tilde{\boldsymbol{s}}||_{A^TA} = \hat{S}X_j\left(X_j^T\hat{S}^TA^TA\hat{S}X_j\right)^{-1}X_j^T\hat{S}^TA^TA\hat{R}^T_{:,j}. \quad (5.36)$$

This time, each row of $\hat{S}^TA^TA\hat{R}^T_{:,j}$ corresponds to the A^TA-inner product of every column of $\hat{S}$ with the jth column of $\hat{R}^T$. Therefore, the number of selected columns is likely to grow when using normal equations, which adds more fill-in in the resulting $\hat{P}$ and also increases the computational cost of the overall method. Figure 5.16 is analogous to 5.15 but with normal equations. The gain does not appear clearly in this case, but numerical experiments in Section 7.2 appeared very important when running the multilevel experiments of Chapter 7.



(a) Least-squares setting R^T and S - $\alpha = 0.625^2$ (b) Least-squares setting R^T and S - $\alpha = 1.75^2$

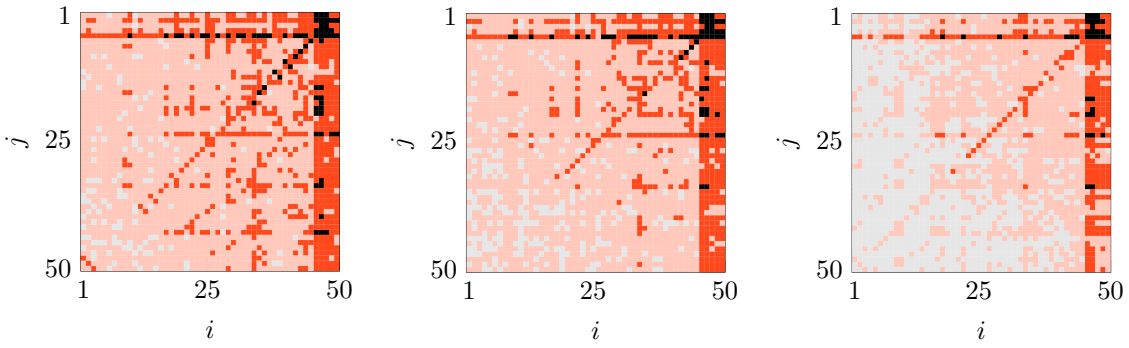
Figure 5.16: Error of the l_2-projection onto the range of the classical and least-squares ideal approximations with the subspace restriction approach for the model problem SL2D-9S with respect to τ - with normal equations

5.3.5 Effect on the pollution

Let us discuss the effect of the ideal approximation phase in this least-squares framework. The colors of the pixels in the following figures are described in Section 5.2.4. Again, below experiments result from both 5-point stencil and 9-point stencil experiments. Lastly, we target the ideal interpolation operator $\hat{P}_*$ with normal equations to ensure its minimization properties (see (5.14) and (5.36)).

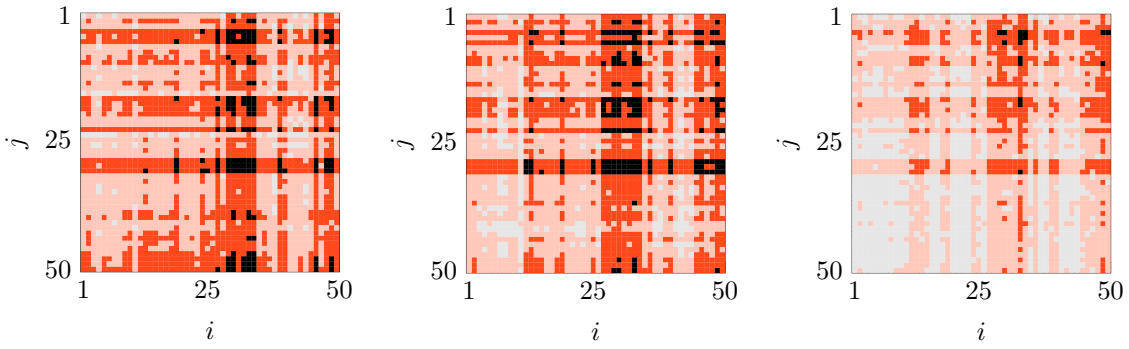
5.3.5.1 The 5-point stencil case

We begin with the 5-point stencil case. Figure 5.17 and 5.18 map the entries of the pollution block K_f for three different values of τ and two values of α. A first and obvious observation for both shifts is that the left bottom hand corners get cleaned from red and pink pixels as τ increases. In other words, the pollution of the smallest eigenvectors of V_c arising from the largest eigenvectors of V_f decreases the fastest as $\hat{P}$ gets better.



(a) $\tau = 0.0$ ($\hat{P} = \hat{R}^T$) (b) $\tau = 0.5$ (c) $\tau = 1.0$

Figure 5.17: Entries of the pollution block K_f with respect to τ and for $\alpha = 0.625^2$ - The 5-point stencil case



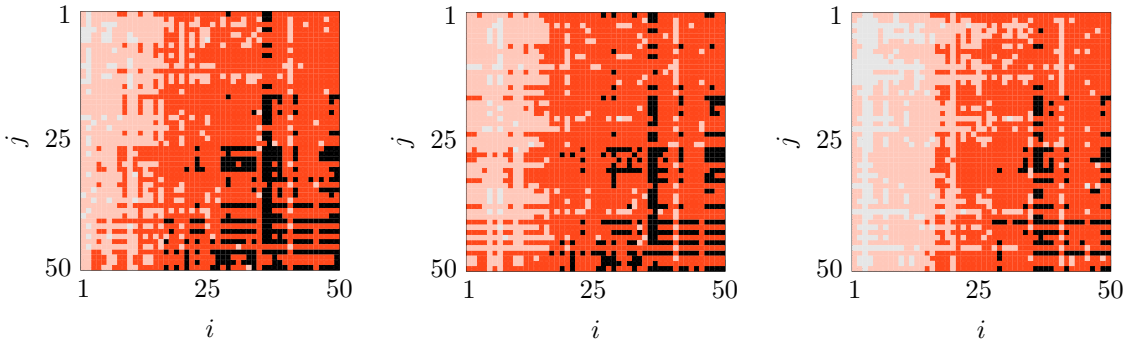
(a) $\tau = 0.0$ ($\hat{P} = \hat{R}^T$) (b) $\tau = 0.5$ (c) $\tau = 1.0$

Figure 5.18: Entries of the pollution block K_f with respect to τ and for $\alpha = 1.75^2$ - The 5-point stencil case

Furthermore, the pollution seems more important for $\alpha = 1.75^2$. However, setting $\tau = 1.0$ decreases the pollution to an equivalent order of magnitude in both cases.

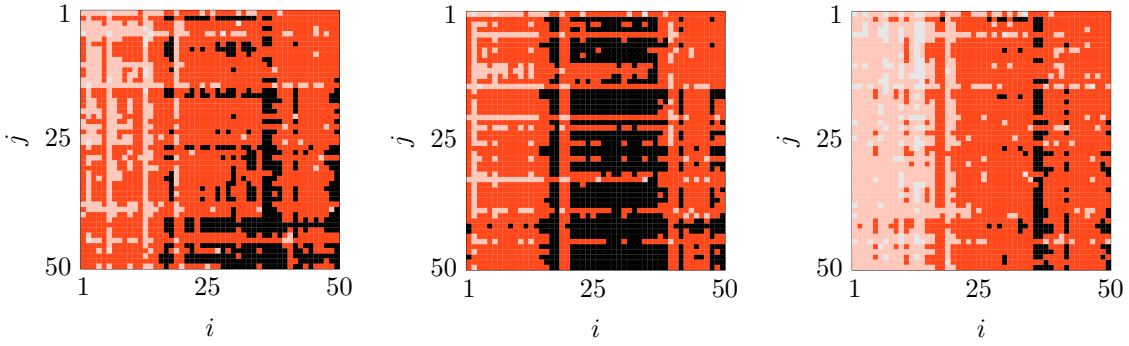
5.3.5.2 The 9-point stencil case

Alternatively, Figure 5.19 and Figure 5.20 represent the entries of K_f in the 9-point stencil case. Again, the pollution for $\alpha = 1.75^2$ is more important. While the effect of the ideal approximation phase is unclear for $\tau = 0.5$, setting $\tau = 1.0$ decreases the pollution of the smallest eigenvectors drastically. Still, Figure 5.16 shows that the l_2-projection error decreases when increasing τ to 0.5, although it does not appear distinctly when looking at the pollution in figures 5.19b and 5.20b. In comparison with the previous Figure 5.8 and Figure 5.9 of the classical setting, approximating $\hat{P}$ seems to have a better impact in this least-squares setting. In fact, both least-squares variable operators approximate the subspaces that the coarse correction and the smoother should capture respectively. This feature allows us to construct an interpolation operator that improves the complementarity principle even further as τ grows. It is the main benefit of the subspace restriction approach, and is certainly what makes the convergence of the multilevel methods experiments presented in Chapter 7 better than with the SPAI based approaches of Section 5.2.



(a) $\tau = 0.0$ ($\hat{P} = \hat{R}^T$) (b) $\tau = 0.5$ (c) $\tau = 1.0$

Figure 5.19: Entries of the pollution block K_f with respect to τ and for $\alpha = 0.625^2$ - The 9-point stencil case



(a) $\tau = 0.0$ ($\hat{P} = \hat{R}^T$) (b) $\tau = 0.5$ (c) $\tau = 1.0$

Figure 5.20: Entries of the pollution block K_f with respect to τ and for $\alpha = 1.75^2$ - The 9-point stencil case

5.3.6 Complexity

As in Section 5.2.5, let us now discuss the complexity of the interpolation operator and of the coarse grid operator with respect to τ. For each numerical experiment, the shift is set to $\alpha = 0.625^2$.

5.3.6.1 Complexity of the interpolation operator

Again, we open the discussion by studying the number of non-zero entries of $\hat{P}$ contained in Table 5.2. Because the maximal size of $\mathcal{C}_i$ in (5.24) is set to 4 for both stencil scenarios, it follows that the number of non-zero entries in $\hat{R}^T$ is approximately the same. Surprisingly, Table 5.2 shows that $\hat{P}$ is denser in the 5-point stencil case. In the 9-point stencil case, the sparsity remains approximately the same with or without normal equations for small values of τ. Conversely when setting $\tau = 1.0$, $\hat{P}$ turns much denser in the 9-point stencil case. The vector

$$\boldsymbol{b}_j := \hat{S}^T A \hat{R}^T_{:,j} \tag{5.37}$$

in (5.35) surely contains more entries in the 9-point stencil case, but the distribution of entries in magnitude is probably less uniform than in 5-point stencil case. In fact, Algorithm 5 selects the columns of $\hat{S}$ such that a proportion τ of the values in $\boldsymbol{b}_j$ in magnitude are selected. Therefore, fewer columns may be selected for a same value of τ in one case, even though its sparsity pattern has more non-zero. Alternatively, setting $\tau = 1.0$ selects all the columns of $\hat{S}$ associated with a non-zero entry of $\boldsymbol{b}_j$. Because the 9-point stencil leads to denser matrix, $\hat{P}$ has more non-zero entries as well for $\tau = 1.0$.

stencil	$\hat{R}^T$	$\tau = 0.3$	$\tau = 0.6$	$\tau = 1.0$	$\hat{P}_*$
5-pts	2.3	7.25	10.6	14.8	50
9-pts	2.2	5.8	8.7	17.3	49.5

(a) Without normal equations

$\hat{R}^T$	$\tau = 0.3$	$\tau = 0.6$	$\tau = 1.0$	$\hat{P}_*$
2.3	6.8	9.9	14.8	50
2.2	5.9	9.3	24.7	49.5

(b) With normal equations

Table 5.2: Average number of non-zero entries per row with respect to τ when approximating the ideal interpolation operator with the subspace restriction approach

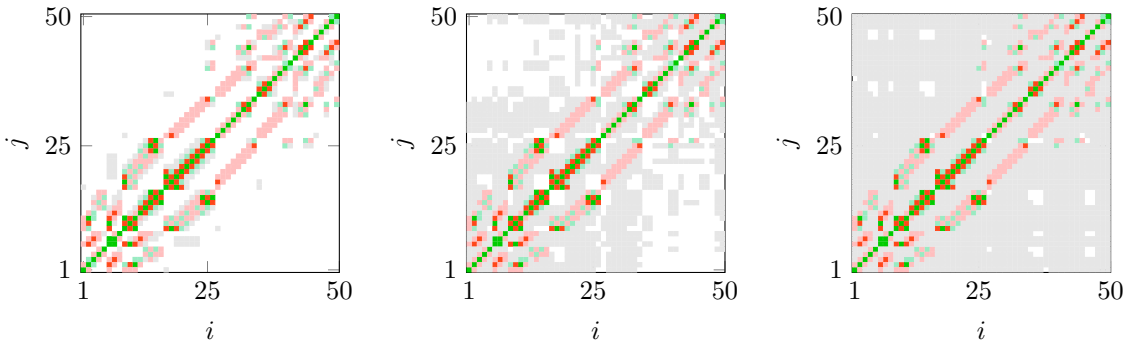
By comparing Table 5.2 with Table 5.1, it appears that the subspace restriction approach with the least-squares variable operators generally leads to more fill-in of the interpolation operator than the first based on the classical set of variable operators. Improving the sparsity of $\hat{P}$ is a topic of future research.

5.3.6.2 Complexity of the coarse grid

The Figure 5.21 portrays the sparsity pattern of the coarse matrix resulting from the least-squares ideal approximations with the subspace restriction approach. The initial matrix results from the 9-point stencil (1.3), and the target ideal interpolation has normal equations. The Figure 5.21 is the counterpart of Figure 5.10, so we refer

to Section 5.2.5 for the colors of the pixels. This time, the Galerkin coarse matrix is computed by way of the triple matrix product

$$\hat{A}_c = \hat{P}^T A \hat{P}. \tag{5.38}$$



(a) $\tau = 0.0$ (b) $\tau = 0.5$ (c) $\tau = 1.0$

Figure 5.21: Fill-in of the coarse matrix $\hat{A}_c$ with respect to τ for the 9-point stencil problem and with normal equations

Here again, the fill-in of the coarse matrix mostly results from small entries, and it may be possible to approximate them by Non-Galerkin matrices. Sparser Non-Galerkin matrices have been developed to reduce the complexity of coarse operators in AMG [37]. The heuristics of this approach are based on the geometrical smooth assumption of near-kernel space, as in most elliptic problems. Extending Non-Galerkin coarsening to the oscillatory near-kernel space of the Helmholtz equation is a topic of further research. In the indefinite case, the Galerkin triple matrix is not enough to guarantee the coarse correction to contract the error in all cases. Subsequently, we present a new alternative coarse correction in Chapter 6 that properly contracts the error. Injecting a Non-Galerkin matrix as a sparse approximation of the Galerkin coarse grid might degrade the convergence, but will never lead to an amplification of the error with our alternative coarse correction.

5.4 Intermediate Conclusion

In traditional multigrid methods originally aimed at solving elliptic problems, the interpolation operator relies on the geometrical smoothness of the near-kernel space. Such a convenient assumption does not exist for Helmholtz. The ideal interpolation operator with the classical variable operators can be good for Helmholtz, but it is unclear if an approximation of it can give sufficient results in the multilevel case because the variable operators are not designed to represent the complementarity between our polynomial smoother and the coarse correction.

By plugging the polynomial smoother developed in Chapter 4 into the optimal framework recalled in Lemma 1, it appears that the range of interpolation should approximate the oscillatory space of smallest eigenvectors denoted by V_c. Hence, we looked for an appropriate coarse variable operator that has good approximation

properties of the eigenvectors of V_c, and, by orthogonality, a fine variable operator that approximates the eigenvectors of V_f. Hence, it led to the development of a least-squares approximation of the ideal interpolation operator. This operator is a better interpolation operator in the general case because the coarse variable operator is designed to approximate the span of V_c. In addition, the coarse variable operator is improved by removing smoother related information contained within the range of the least-squares fine variable operator.

Lastly, we have shown that the ideal approximation phase focuses on decreasing the pollution of the smallest eigenvectors arising from the large eigenvectors. This type of pollution is the most problematic on the contraction rate of the coarse correction, and can lead to divergence. In the next chapter, we illustrate this issue and present an alternative coarse correction that fixes the divergence scenarios.

Chapter 6

Coarse Correction for Helmholtz

In previous chapters, we first introduced a Chebyshev polynomial smoother aimed at damping eigenvectors independently of their signs, and then presented an interpolation operator whose range approximates the oscillatory near-kernel space of the Helmholtz equation. However, one more important issue inherent to the indefinite nature of Helmholtz remains. In the SPD case, the discretization matrix A defines a norm. As a consequence, applying the coarse correction to the right-hand side b is equivalent to computing the best approximation that minimizes its difference with the solution in A-norm, as in (2.42). Such an A-norm does not exist in the indefinite case. In the context of constructing appropriate interpolation rules for Helmholtz, the lack of variational properties in the A-orthogonal projection motivated us to use the normal equations in the ideal interpolation definition, even though working with A can provide good results in practice. The same concern applies for the coarse correction. Using A may result in good convergence rates, but it can also lead to slow convergence. Worse still, the coarse correction may even amplify the error and lead to divergence. This issue is formalized through the new concept of "pollution" in Chapter 3.2. In particular, we have shown that even a good interpolation operator can lead to a coarse correction that amplifies the error. Although numerical experiments of Chapter 7 reveal that improving the ideal approximation yields faster convergence, it is not enough for ensuring that the coarse correction operates as a contraction on the error. Subsequently, an alternative coarse correction is necessary for indefinite cases.

We begin this chapter by illustrating the effect of the coarse correction when using the least-squares ideal approximation operators developed in Section 5.3. In particular, we will see that divergence can happen in certain cases, even though the interpolation operator satisfies good approximation properties. With the goal of addressing the divergence concern, we present an alternative coarse correction that guarantees convergence without assumption on the nature of the matrix.

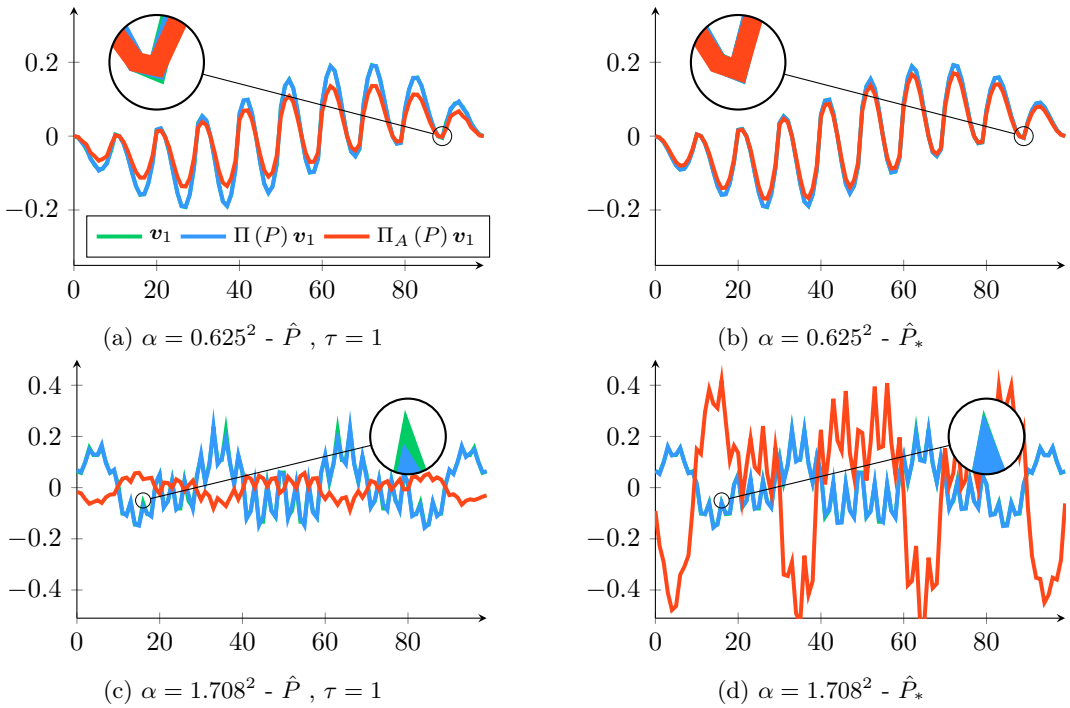
6.1 Alteration of the coarse correction in the indefinite case

While both smoothers and interpolation operators are now designed to face two inconvenient properties of the Helmholtz equation, signed eigenvalues and oscillatory near-kernel space, the effectiveness of the classical coarse correction is not guaranteed

in an indefinite context. Even worse, the classical coarse correction can amplify the error, although the interpolation operator has good approximation properties, and lead to divergence. Before discussing an alternative coarse correction, let us highlight how the matrix indefiniteness can corrupt the classical coarse correction with a simple illustration.

6.1.1 Illustration of the error amplification

Figure 6.1 plots the smallest eigenvector $\boldsymbol{v}_1$ (in green) of a two-dimensional shifted Laplacian matrix, its best representation within the range of interpolation (in blue) and the result of the coarse correction when applied to $\boldsymbol{v}_1$ (in red).



(a) $\alpha = 0.625^2$ - $\hat{P}$, $\tau = 1$

(b) $\alpha = 0.625^2$ - $\hat{P}_*$

(c) $\alpha = 1.708^2$ - $\hat{P}$, $\tau = 1$

(d) $\alpha = 1.708^2$ - $\hat{P}_*$

Figure 6.1: Smallest eigenvector $\boldsymbol{v}_1$ vs. its l_2-projection $\Pi(P)\boldsymbol{v}_1$ vs. its coarse correction $\Pi_A(P)\boldsymbol{v}_1$, for $\hat{P}$ and $\hat{P}_*$ and with respect to α

Numerical experiments related to Figure 6.1 are made through a 5-point stencil discretization of the shifted Laplacian matrix. The coarse correction is built on the least-squares ideal approximation introduced in Section 5.3. In this experiment, the columns of $\hat{S}$ are chosen by setting $\tau = 1$. On the right side, both figures 6.1b and 6.1d rely on the exact least-squares ideal interpolation operator denoted by $\hat{P}_*$. Both top figures correspond to the case where $\alpha = 0.625^2$, whereas the two bottom figures are for $\alpha = 1.708^2$. As expected, the smallest eigenvector gets more oscillatory as the shift α grows.

A first observation is that blue and green curves align almost perfectly in all four of Figure 6.1, meaning that the least-squares interpolation operator introduced in Chapter 5 offers a good approximation to the potentially oscillatory smallest eigenvector. Even though the l_2-orthogonal projection of $\boldsymbol{v}_1$ within the range of $\hat{P}_*$ fits

better, $\hat{P}$ still demonstrates excellent approximation properties for both shifts. In figures 6.1a and 6.1b, where $\alpha = 0.625^2$, the red coarse correction vectors are relatively close to their respective green smallest eigenvector. The slight difference between both is only a matter of amplitude. As we would naturally expect, the coarse correction built on $\hat{P}_*$ returns a better coarse correction vector than the coarse correction with P_*. However, the second shifted case $\alpha = 1.708^2$ tells another story. In figures 6.1c and 6.1d, the oscillations of the coarse correction vectors and of the smallest eigenvectors are synchronized, however their directions are reversed. While the difference between blue and green curves indicates a very small interpolation error, the smallest eigenvector is not contracted by the coarse correction, but amplified. This time, the amplification error is even more important when using the ideal interpolation operator $\hat{P}_*$ although the l_2-orthogonal projection of the smallest eigenvector is better than its counterpart $\hat{P}$.

The same experiments applied to the classical ideal interpolation are discussed in Section A.7.

6.1.2 On how improving the interpolation can increase the error amplification

In Chapter 5, we introduced the concept of "pollution". Let us recall the definition of the K matrix priorly given in (3.70), such that

$$K := V^T P (V_c^T P)^{-1} = \begin{bmatrix} I_c \\ K_f \end{bmatrix}. \tag{6.1}$$

In particular, Theorem 1 connects the error of interpolation with the pollution block K_f. We also recall that an entry (j, i) of K_f corresponds to the contribution of the jth large eigenvector of V_f to the l_2-orthogonal projection onto range(P) of the ith small eigenvector of V_c. Let us recall from the Section 5.2.4 and Section 5.3.5 that improving the ideal approximation tends to decrease the magnitude entries of K_f in proportion with their associated eigenvalues. However, we also discussed that improving the interpolation can decrease the pollution and bring it to a critical point where the error explodes extremely fast. This phenomenon is illustrated in Figure 3.8 on a simple 2×2 example. In our case, we observe this feature by comparing the red curves of Figure 6.1c and Figure 6.1d. Here, using the exact ideal interpolation operator amplifies the error even more importantly than an approximation of it. The following lemma helps understand this concern in the more general case.

Lemma 2. *Let ϕ_i be defined as follows*

$$\phi_i := \lambda_i - \left(\left[\left(\Lambda_c + K_f^T \Lambda_f K_f \right)^{-1} \right]_{i,i} \right)^{-1}. \tag{6.2}$$

The coarse correction contracts an eigenvector $\boldsymbol{v}_i$ of V_c if and only if

$$\left| \boldsymbol{v}_i^T E \boldsymbol{v}_i \right| < 1 \qquad \Leftrightarrow \qquad \frac{\phi_i}{\lambda_i} < \frac{1}{2}. \tag{6.3}$$

Proof. First, injecting (6.2) in (3.77) gives

$$\boldsymbol{v}_i^T E \boldsymbol{v}_i = 1 - \frac{\lambda_i}{\lambda_i - \phi_i}. \tag{6.4}$$

Subsequently, the coarse correction operates as a contraction of $\boldsymbol{v}_i$ if

$$-1 < 1 - \frac{\lambda_i}{\lambda_i - \phi_i} < 1 \quad \Leftrightarrow \quad 0 < \frac{\lambda_i}{\lambda_i - \phi_i} < 2. \tag{6.5}$$

From Equation (6.5), we finally ends with the necessary condition

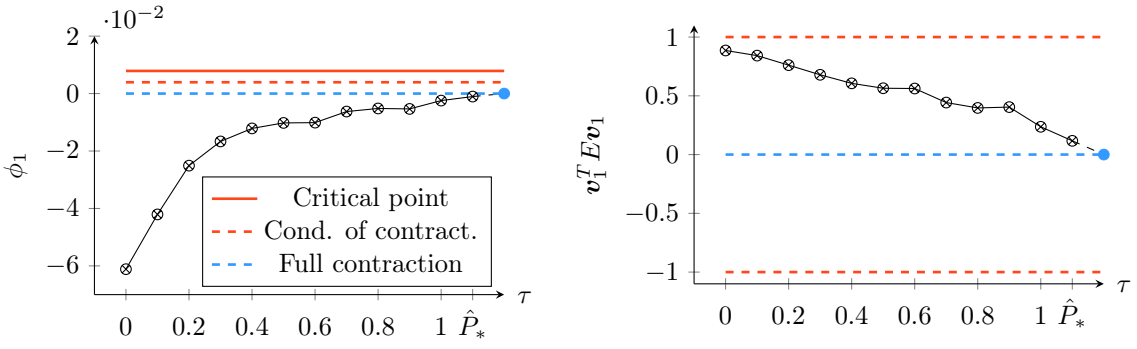
$$\frac{1}{2} < 1 - \frac{\phi_i}{\lambda_i} \quad \Leftrightarrow \quad \frac{\phi_i}{\lambda_i} < \frac{1}{2}. \tag{6.6}$$

□

From Lemma 2, the coarse correction contracts $\boldsymbol{v}_i$ if and only if

$$\left|\boldsymbol{v}_i^T E \boldsymbol{v}_i\right| < 1 \quad \Leftrightarrow \quad \begin{cases} \phi_i \in \left(-\infty, \dfrac{\lambda_i}{2}\right) & \text{if } \lambda_i > 0 \\ \phi_i \in \left(\dfrac{\lambda_i}{2}, +\infty\right) & \text{if } \lambda_i < 0 \end{cases}. \tag{6.7}$$

Moreover, (6.4) highlights that the contraction of $\boldsymbol{v}_i$ is zero if the difference ϕ_i is zero, but also reveals the critical point $\phi_i = \lambda_i$ for which the error explodes. To elaborate on this discussion, Figure 6.2a represents ϕ_1 and Figure 6.2b represents $\boldsymbol{v}_1^T E \boldsymbol{v}_1$ for several approximations of the ideal interpolation operator and $\alpha = 0.625^2$. The x-axis represents the degree of approximation τ plus the ideal case.

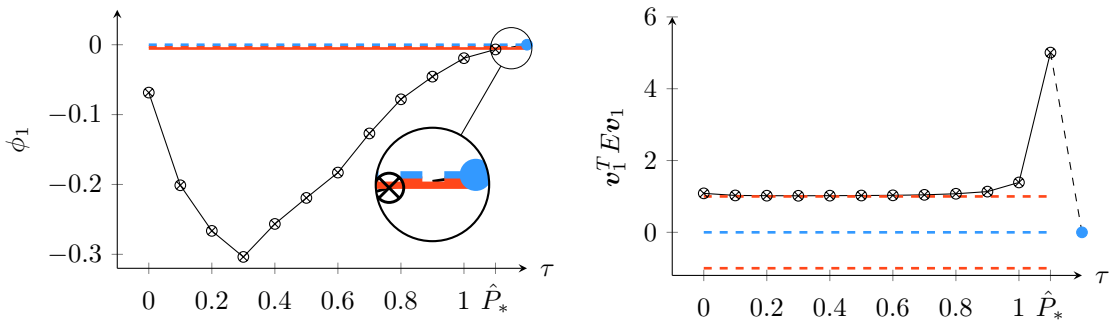


(a) ϕ_1 with respect to τ and $\hat{P}_*$

(b) $\boldsymbol{v}_1^T E \boldsymbol{v}_1$ with respect to τ and $\hat{P}_*$

Figure 6.2: ϕ_1 and $\boldsymbol{v}_1^T E \boldsymbol{v}_1$ with respect to $\hat{P}$ and for $\alpha = 0.625^2$

In Figure 6.2, the eigenvalue associated with $\boldsymbol{v}_1$ is $\lambda_1 = 0.00788199$. A first observation is that decreasing the magnitude of ϕ_1 by improving $\hat{P}$ improves the capture of $\boldsymbol{v}_1$ in this case. This feature was already highlighted by comparing both previous figures 6.1a and 6.1b, where the ideal interpolation operator had better approximation properties and consequently enabled the coarse correction to better approximate the smallest eigenvector $\boldsymbol{v}_1$.



(a) ϕ_1 with respect to τ and $\hat{P}_*$

(b) $\boldsymbol{v}_1^T E \boldsymbol{v}_1$ with respect to τ and $\hat{P}_*$

Figure 6.3: ϕ_1 and $\boldsymbol{v}_1^T E \boldsymbol{v}_1$ with respect to $\hat{P}$ and for $\alpha = 1.708^2$

By comparison, Figure 6.3 correlates ϕ_1 with $\boldsymbol{v}_1^T E \boldsymbol{v}_1$ in the case where the shift equals $\alpha = 1.708^2$, and where $\lambda_1 = -0.0054$. In that case, none of the interpolation operators enables the coarse correction to contract the smallest eigenvector. As a consequence, the method diverges in all cases. The left Figure 6.3a indicates an improvement of the approximation properties of the interpolation operator for large values of τ, as ϕ_1 converges to the full contraction point $\phi_1 = 0$. However, the full contraction point represented by the dashed blue line is higher than the critical exploding point designated by the solid red line at $\phi_1 = \lambda_1$. In other words, getting closer to the condition of contraction $\phi_1 = \lambda_1/2$ requires crossing the critical point $\phi_1 = \lambda_1$.

While designing a better interpolation operator in the positive case always implies a better contraction rate, it eventually brings ϕ_i closer to the critical point in the indefinite case. This phenomenon explains why the ideal coarse correction of Figure 6.1d returns an amplified reversed approximation of $\boldsymbol{v}_1$ relative to its counterpart of Figure 6.1c.

As illustrated in Figure 6.1, a good interpolation operator with small K_f can still cause divergence although it satisfies good approximation properties. The classical coarse correction appears hopeless for indefinite problems.

6.2 The alternative coarse correction

As discussed in the previous section, the classical coarse correction is not equivalent to a minimization problem in the indefinite case, and improving P will never be enough to remedy this loss of equivalence. Moreover, because the interpolation operator developed in Chapter 5 targets the smallest eigenvectors of each level, every coarser matrix is more indefinite than its fine parent. Then, as the number of coarse levels increases, the balance between negative and positive eigenvalues reaches an equilibrium, and makes the effectiveness of the classical coarse correction difficult to predict. Nevertheless, Figure 6.1 shows that the interpolation operator has good approximation properties for the oscillatory near-kernel space. In particular, the Figure 6.1 suggests that only the direction of the coarse correction vector has to be changed; the shape is correct. Hence, a coarse correction that amplifies or flips the

smallest eigenvectors can still provide pertinent information for solving the system.

In this section, we propose to minimize the approximation error in a proper norm for indefinite problems and within a space composed of vectors returned by the classical coarse correction. Moreover, to decrease the eigenvector pollution, each coarse correction vector is smoothed by the polynomial smoother of Chapter 4.

6.2.1 General considerations on GMRES

The *Generalized Minimal RESidual* (GMRES) method [64, 66] approximates the solution in a Krylov subspace by minimizing the residual in the Euclidean norm. The method can solve any class of matrix system since the norm is valid independent of the context, which is of particular interest for the indefinite case. Let us first define some notation before introducing the alternative coarse correction. Let W_p be the $n \times p$ rectangular matrix containing the p orthonormalized Krylov vectors such that

$$\text{range}\,(W_p) = \text{span}\left\{\boldsymbol{b}, A\boldsymbol{b}, A^2\boldsymbol{b}, \ldots, A^{p-1}\boldsymbol{b}\right\}. \tag{6.8}$$

Each column of W_p is orthonormalized following a Gram-Schmidt process. The coefficients of the orthonormalization are stored in the rectangular Hessenberg matrix $\bar{H}_p$ of size $p+1 \times p$. The square matrix H_p is of size $p \times p$ and obtained from $\bar{H}_p$ by deleting its last row. Both matrices W_p and H_p are linked by

$$AW_p = W_{p+1}\bar{H}_p \qquad \text{and} \qquad W_p^T A W_p = H_p, \tag{6.9}$$

which leads to the following equality

$$\min_{\tilde{\boldsymbol{x}} \in \text{range}(W_p)} \|\boldsymbol{b} - A\tilde{\boldsymbol{x}}\|_2 = \min_{\boldsymbol{\rho}_p \in \mathbb{C}^p} \|\boldsymbol{b} - AW_p\boldsymbol{\rho}_p\|_2 = \min_{\boldsymbol{\rho}_p \in \mathbb{C}^p} \|W_p^T\boldsymbol{b} - H_p\boldsymbol{\rho}_p\|_2. \tag{6.10}$$

In practice, GMRES takes advantage of the convenient Hessenberg shape of $\bar{H}_p$ to construct an upper triangular matrix by applying Given's rotations. The minimization of the residual then relies on a backward substitution. The relation (6.9) can be generalized [22] to any arbitrary subspace $W_p = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_p]$ such that

$$\underset{\tilde{\boldsymbol{x}} \in \text{range}(W_p)}{\arg\min} \; ||\boldsymbol{b} - A\tilde{\boldsymbol{x}}||_2 = W_p H_p^{-1} Z_p^T \boldsymbol{b} \qquad \text{with} \qquad AW_p = Z_p H_p, \tag{6.11}$$

and where Z_p denotes the orthonormalized basis of AW_p. Note that the Arnoldi relation (6.11) does not define any particular recurrence relation since W_p is arbitrary and not necessarily designed by successive matrix vector products. In addition, the only matrix that needs to be orthonormal in the generalized setting is Z_p.

6.2.2 Minimization within a space of coarse correction vectors

As mentioned in the introduction of this section, the interpolation operator has good approximation properties for the oscillatory near-kernel space. Even though the small eigenvectors that constitute each coarse correction vector are likely to be oriented in the wrong direction or amplified because of the pollution effect introduced in Section 6.1, they still provide useful information about the near-kernel space. Hence, the major idea of the alternative coarse correction is to recycle the

potentially wrong oriented coarse correction vectors to design a subspace in which the residual is minimized in Euclidean norm.

Let W_p be the set of coarse correction vectors of the pth iteration linked by the Arnoldi relation (6.11) with its orthonormal counterpart Z_p. Accordingly, let $\boldsymbol{w}_j \in W_p$ and $\boldsymbol{z}_j \in Z_p$ denote the jth vectors of the set W_p and Z_p respectively. For ease of discussion, we introduce this alternative coarse correction with a two-level method. The multilevel case will be addressed in the next chapter with numerical experiments.

At each iteration p, the classical coarse correction returns a new coarse correction vector that is smoothed by the Chebyshev polynomial smoother presented in Chapter 4. This new smoothed coarse correction vector is thereafter added to the previous set as follows

$$W_p = [W_{p-1} \quad , \quad \boldsymbol{w}_p] \text{ with } \boldsymbol{w}_p = q_{d+1}^{\nu}(A^2) P \left(P^T A P\right)^{-1} P^T \boldsymbol{r}^{(p)}, \tag{6.12}$$

where $\boldsymbol{r}^{(p)}$ designates the residual at the pth iteration. Once the coarse correction vector is smoothed and that W_p is formed, the set Z_p is extended as follows

$$Z_p = [Z_{p-1} \quad , \quad \boldsymbol{z}_p] \text{ with } \boldsymbol{z}_p = \frac{1}{h_{p,p}} \left(A\boldsymbol{w}_p - \sum_{j=1}^{p-1} h_{j,p} \cdot \boldsymbol{z}_j \right), \tag{6.13}$$

where coefficients $h_{j,p}$ result from the orthonormalization process of the new vector $A\boldsymbol{w}_p$. These coefficients are stored in the squared upper triangular matrix

$$H_p = \left[\begin{array}{ccc|c} & & & h_{1,p} \\ & H_{p-1} & & \vdots \\ & & & h_{p-1,p} \\ \hline 0 & \cdots & 0 & h_{p,p} \end{array} \right] \quad \text{with} \quad h_{j,p} = \begin{cases} \langle \boldsymbol{z}_j, \boldsymbol{z}_p \rangle & \text{if } j < p \\ ||\boldsymbol{z}_p||_2 & \text{if } j = p \end{cases}. \tag{6.14}$$

Even though the notation is similar to the Arnoldi relation of Krylov methods, recall that W_p and Z_p do not correspond to Krylov subspaces, but are filled successively as the multigrid cycle is iterated. Krylov subspaces lead to a particular form of the Arnoldi relation (6.11), where H_p is Hessenberg because Z_p has one more column than W_p, and that a particular recurrence relation links both subspaces (i.e., range(W_p) is a subspace of range(Z_{p+1}) in Krylov methods). In our case, Algorithm 6 presents the alternative two-level cycle, and can be compared with Algorithm 1 which recaps the classical two-level cycle.

While Equation (6.12) gives the recursive form of W_p, one can derive a more general formula by letting $E_{TG}(W_p)$ be the two-grid method with the alternative coarse correction at the pth iteration, such that

$$E_{TG}\left(W_p\right) = q_{d+1}^{\nu}\left(A^2\right) \cdot \left(I - W_p H_p^{-1} Z_p^T A\right) \cdot q_{m+1}^{\nu}\left(A^2\right). \tag{6.15}$$

Therefore, the general form of W_p at the pth iteration is

$$W_p = \left[\boldsymbol{w}_1, \ldots, \boldsymbol{w}_p\right], \tag{6.16}$$

Algorithm 6 Two-level pth cycle with the alternative coarse correction

Inputs : $\boldsymbol{b}$ right-hand side, $\tilde{\boldsymbol{x}}$ approximation of $\boldsymbol{x}$, $\boldsymbol{r} = b - A\tilde{\boldsymbol{u}}$ residual
M smoother, P interpolation operator
for $j = 1, \nu$ **do**
$\tilde{\boldsymbol{x}} \leftarrow \tilde{\boldsymbol{x}} + p(A^2)\boldsymbol{r}$
$\boldsymbol{r} \leftarrow \boldsymbol{b} - A\tilde{\boldsymbol{x}}$
end for
$\boldsymbol{r}_{\mathcal{C}} \leftarrow P^T\boldsymbol{r}$
$\boldsymbol{e}_{\mathcal{C}} \leftarrow \text{Solve}(P^TAP, \boldsymbol{r}_{\mathcal{C}})$
$\boldsymbol{w} \leftarrow q_{d+1}^{\nu}(A^2)P\boldsymbol{e}_{\mathcal{C}}$
$\hat{\boldsymbol{w}}, H_p \leftarrow \text{Orthonormalize}(\boldsymbol{w}, Z_{p-1})$
$W_p, Z_p \leftarrow [W_{p-1}\ ,\ \boldsymbol{w}], [Z_{p-1}\ ,\ \hat{\boldsymbol{w}}]$
for $j = 1, \nu$ **do**
$\tilde{\boldsymbol{x}} \leftarrow \tilde{\boldsymbol{x}} + p(A^2)\boldsymbol{r}$
$\boldsymbol{r} \leftarrow \boldsymbol{b} - A\tilde{\boldsymbol{x}}$
end for
$\tilde{\boldsymbol{x}} \leftarrow \tilde{\boldsymbol{x}} + W_pH_p^{-1}Z_p^T\boldsymbol{r}$
$\boldsymbol{r} \leftarrow \boldsymbol{b} - A\tilde{\boldsymbol{x}}$
Output : $\tilde{\boldsymbol{x}}$ approximation of $\boldsymbol{x}$ at the end of the cycle

with each vector of W_m being defined by

$$\begin{aligned} \boldsymbol{w}_j &= q_{d+1}^{\nu}(A^2) \cdot \Pi_A(P) \cdot E_{TG}(W_{j-1})\, \boldsymbol{e}^{(j-1)} \\ &= q_{d+1}^{\nu}(A^2) \cdot \Pi_A(P) \cdot \textstyle\prod_{k=0}^{j-1} E_{TG}(W_k)\, \boldsymbol{e}^{(0)} \end{aligned} \quad , \quad j = 1, \ldots, p\,, \tag{6.17}$$

and where $\boldsymbol{e}^{(j)}$ corresponds to the error between the solution $\boldsymbol{x}$ and the approximation $\boldsymbol{x}^{(j)}$ at the jth iteration ($\boldsymbol{x}^{(0)}$ is the initial guess). Incidentally, since Z_p is constructed by orthonormalizing the left multiplication of W_p with A, its range naturally satisfies

$$\text{range}(Z_p) = \text{range}(AW_p)\,. \tag{6.18}$$

6.2.3 Effect of the pollution on the alternative coarse correction

Let us discuss the effect of the pollution on the alternative coarse correction. The Section 6.1 demonstrated that the pollution block K_f impacts the classical coarse correction. Because the minimization space W_p is generated with the classical coarse correction by way of (6.12), the block of pollution still impacts the contraction of the small eigenvectors of V_c when using the alternative coarse correction. To better understand this phenomenon, let us first demonstrate that applying the alternative coarse correction is equivalent to solving a normal equations problem.

Theorem 4. *Let W_p be the minimization space of smoothed coarse correction vectors as prescribed by equations* (6.16) *and* (6.17)*. Accordingly, let Z_p and H_p be the corresponding Arnoldi operators. Hence, applying the alternative coarse correction is equivalent to solving the normal equations problem restricted to the minimization space W_p, such that*

$$W_pH_p^{-1}Z_p^TA = \Pi_{A^TA}(W_p)\,. \tag{6.19}$$

Proof. From the Arnoldi relation (6.11), we have

$$Z_p = AW_pH_p^{-1} \quad , \quad H_p = Z_p^TAW_p = H_p^{-T}W_p^TA^TAW_p. \tag{6.20}$$

Substituting (6.20) in (4), it naturally follows that solving the minimization problem (6.11) is equivalent to solving the normal equation system within the subspace spanned by W_p such that

$$\begin{aligned} W_p H_p^{-1} Z_p^T A &= W_p \left(W_p^T A^T A W_p\right)^{-1} H_p^T Z_p^T A \\ &= W_p \left(W_p^T A^T A W_p\right)^{-1} W_p^T A^T A \\ &= \Pi_{A^T A}\left(W_p\right). \end{aligned} \tag{6.21}$$

□

As stated by Theorem 4, applying the alternative coarse correction amounts to solving a normal equations problem. Resorting to the Euclidean norm in (6.11) prevents the divergence by squaring the eigenvalues of the initial problem as the next Theorem 5 demonstrates. The downside of this approach is that the gap between small and large magnitude eigenvalues increases. As a consequence, the contraction of the smallest eigenvectors is impacted due to the pollution arising from the large ones.

Theorem 5 formalizes this issue in the hypothetical case where the minimization space contains only one vector. In fact, this particular case happens after the first multigrid cycle. As we will later discuss on the effect of additional smoothing steps on the coarse correction vectors, we first address the case where W_1 is constructed without smoothing the coarse correction vector. For ease of notation, define $\boldsymbol{a}_i$ the ith column of the inverse of $K^T \Lambda K$ such that

$$\boldsymbol{a}_i := \left[\left(K^T \Lambda K\right)^{-1}\right]_{:,i} \quad , \quad \boldsymbol{a}_{i,i} = \left[\left(K^T \Lambda K\right)^{-1}\right]_{i,i}. \tag{6.22}$$

Theorem 5. *Let the minimization space W_1 of the alternative coarse correction be defined by a single coarse correction vector as in the first multigrid cycle, such that*

$$\boldsymbol{v}_i \in V_c \qquad , \qquad W_1 = \Pi_A(P)\boldsymbol{v}_i, \tag{6.23}$$

and Z_1 and H_1 the associated Arnoldi operators as prescribed by equations (6.13) *and* (6.14) *respectively. Therefore, the alternative coarse correction contracts $\boldsymbol{v}_i$ at a rate*

$$\boldsymbol{v}_i^T \left(I - W_1 H_1^{-1} Z_1^T A\right) \boldsymbol{v}_i = 1 - \frac{\lambda_i^2 \boldsymbol{a}_{i,i}^2}{\boldsymbol{a}_i^T \left(\Lambda_c^2 + K_f^T \Lambda_f^2 K_f\right) \boldsymbol{a}_i}. \tag{6.24}$$

Proof. First, recall from Theorem 4 that applying the alternative coarse correction amounts to solving a normal equation problem, such that

$$\boldsymbol{v}_i^T \left(I - W_1 H_1^{-1} Z_1^T A\right) \boldsymbol{v}_i = 1 - \boldsymbol{v}_i^T W_1 \left(W_1^T A^T A W_1\right)^{-1} W_1^T A^T A \boldsymbol{v}_i. \tag{6.25}$$

Secondly, let the Euclidean basis vectors $\boldsymbol{e}_i$ and $\boldsymbol{c}_i$ be defined as in the proof of Theorem 1. Since $\boldsymbol{v}_i$ belongs to the set of smallest eigenvectors V_c and that the classical coarse correction can be formulated with the K matrix as in (3.79), then

W_1 has the form

$$\begin{aligned} W_1 = \Pi_A(P)\,\boldsymbol{v}_i &= VK\left(K^T\Lambda K\right)^{-1}K^T\Lambda V^T\boldsymbol{v}_i \\ &= \lambda_i VK\left(K^T\Lambda K\right)^{-1}K^T\boldsymbol{e}_i \\ &= \lambda_i VK\left(K^T\Lambda K\right)^{-1}\boldsymbol{c}_i \\ &= \lambda_i VK\boldsymbol{a}_i. \end{aligned} \tag{6.26}$$

As a consequence of (6.26), we have

$$\begin{aligned} \boldsymbol{v}_i^T W_1 &= \boldsymbol{v}_i^T\Pi_A(P)\,\boldsymbol{v}_i \\ &= \lambda_i\boldsymbol{v}_i^T VK\boldsymbol{a}_i \\ &= \lambda_i\boldsymbol{a}_{i,i} \end{aligned} \quad , \tag{6.27}$$

and, in a similar way, we also obtain

$$\begin{aligned} W_1^T A^T A\boldsymbol{v}_i &= \boldsymbol{v}_i^T\Pi_A(P)\,A^T A\boldsymbol{v}_i \\ &= \lambda_i^3\boldsymbol{v}_i^T VK\boldsymbol{a}_i \\ &= \lambda_i^3\boldsymbol{a}_{i,i} \end{aligned} \quad . \tag{6.28}$$

Equation (6.26) also implies that left multiplying W_1 by the initial matrix A gives

$$AW_1 = A\Pi_A(P)\,\boldsymbol{v}_i = \lambda_i AVK\boldsymbol{a}_i = \lambda_i V\Lambda K\boldsymbol{a}_i, \tag{6.29}$$

such that the middle term of the right member in (6.25) is

$$W_1^T A^T AW_1 = \boldsymbol{v}_i^T\left(A\Pi_A(P)\right)^T A\Pi_A(P)\,\boldsymbol{v}_i = \lambda_i^2\boldsymbol{a}_i^T K^T\Lambda^2 K\boldsymbol{a}_i. \tag{6.30}$$

Injecting equations (6.27), (6.28), and (6.30) in the contraction rate formula (6.25) written in normal equation form finally leads to

$$\begin{aligned} \boldsymbol{v}_i^T\left(I - W_1H_1^{-1}Z_1^T A\right)\boldsymbol{v}_i &= 1 - \boldsymbol{v}_i^T W_1\left(W_1^T A^T AW_1\right)^{-1}W_1^T A^T A\boldsymbol{v}_i \\ &= 1 - \frac{\lambda_i^4\boldsymbol{a}_{i,i}^2}{\lambda_i^2\boldsymbol{a}_i^T\left(\Lambda_c^2 + K_f^T\Lambda_f^2 K_f\right)\boldsymbol{a}_i} \\ &= 1 - \frac{\lambda_i^2\boldsymbol{a}_{i,i}^2}{\boldsymbol{a}_i^T\left(\Lambda_c^2 + K_f^T\Lambda_f^2 K_f\right)\boldsymbol{a}_i}. \end{aligned} \tag{6.31}$$

□

In fact, Theorem 5 highlights that resorting to the alternative coarse correction squares the eigenvalues involved in the formula of the contraction rate. This property tends to silence the smallest eigenvalues over the largest, which decreases the contraction of the near-kernel space. Naturally, decreasing the entries of the pollution block K_f by improving the interpolation operator limits this issue. For instance, in the case where the range of P contains $\boldsymbol{v}_i$ exactly, then the ith column of K_f is zero such that the vector $\boldsymbol{a}_i$ only contains $a_{i,i}$ on its ith entry and zero elsewhere. As a consequence, the contraction is zero. However, in general, the trade-off between sparsity and approximation properties of P makes the pollution unavoidable. Even though it may be possible to reduce the entries of K_f enough to achieve good convergence, the pollution is likely to remain a strong limitation for the alternative coarse

correction as well, which can potentially slow the convergence down dramatically.

For this reason, smoothing the classical coarse correction vectors in W_1 by way of the polynomial $q_{d+1}^{\nu}(A^2)$ as prescribed by (6.12) compensates for this effect by reducing the prevalence of large eigenvectors in the minimization space. This idea of damping the large eigenvalues to reveal the smaller ones is also used to generate a relevant set of test vectors for the construction of the coarse variable operator introduced in Section 5.3.2.

The next theorem highlights the effect of the additional smoothing steps applied to the coarse correction vector.

Theorem 6. *Let the minimization space W_1 of the alternative coarse correction be defined by a single smoothed coarse correction vector as in the first multigrid cycle, such that*

$$\boldsymbol{v}_i \in V_c \qquad , \qquad W_1 = q_{d+1}^{\nu}\left(A^T A\right)\Pi_A\left(P\right)\boldsymbol{v}_i. \tag{6.32}$$

and Z_1 and H_1 the associated Arnoldi operators as prescribed by equations (6.13) *and* (6.14) *respectively. Therefore, the alternative coarse correction contracts $\boldsymbol{v}_i$ at a rate*

$$\boldsymbol{v}_i^T\left(I - W_1 H_1^{-1} Z_1^T A\right)\boldsymbol{v}_i = 1 - \frac{q_{d+1}^{2\nu}\left(\lambda_i^2\right)\lambda_i^2 \boldsymbol{a}_{i,i}^2}{\boldsymbol{a}_i^T\left(q_{d+1}^{2\nu}\left(\Lambda_c^2\right)\Lambda_c^2 + K_f^T q_{d+1}^{2\nu}\left(\Lambda_f^2\right)\Lambda_f^2 K_f\right)\boldsymbol{a}_i}. \tag{6.33}$$

Proof. This proof follows the same development than the proof of Theorem 5. In this setting, the alternative coarse correction is built on a single smoothed coarse correction vector such that

$$W_1 = q_{d+1}^{\nu}\left(A^T A\right)\Pi_A\left(P\right)\boldsymbol{v}_i = \lambda_i V q_{d+1}^{\nu}\left(\Lambda^2\right)K\boldsymbol{a}_i. \tag{6.34}$$

Injecting $q_{d+1}^{\nu}(A^2)$ in both equations (6.27) and (6.28), we obtain

$$\boldsymbol{v}_i^T W_1 = \lambda_i q_{d+1}^{\nu}\left(\lambda_i^2\right)\boldsymbol{a}_{i,i} \quad \text{and} \quad W_1^T A^T A\boldsymbol{v}_i = \lambda_i^3 q_{d+1}^{\nu}\left(\lambda_i^2\right)\boldsymbol{a}_{i,i}. \tag{6.35}$$

Similarly to (6.30), we have

$$W_1^T A^T A W_1 = \lambda_i^2 \boldsymbol{a}_i^T\left(q_{d+1}^{2\nu}\left(\Lambda_c^2\right)\Lambda_c^2 + K_f^T q_{d+1}^{2\nu}\left(\Lambda_f^2\right)\Lambda_f^2 K_f\right)\boldsymbol{a}_i. \tag{6.36}$$

For the same reason that the alternative coarse correction is equivalent to solving a normal equation problem of the form (6.25), one can inject the three terms of equations (6.35) and (6.30) such that

$$\begin{aligned}\boldsymbol{v}_i^T\left(I - W_1 H_1^{-1} Z_1^T A\right)\boldsymbol{v}_i &= 1 - \boldsymbol{v}_i^T W_1\left(W_1^T A^T A W_1\right)^{-1} W_1^T A^T A\boldsymbol{v}_i \\ &= 1 - \frac{q_{d+1}^{2\nu}\left(\lambda_i^2\right)\lambda_i^2 \boldsymbol{a}_{i,i}^2}{\boldsymbol{a}_i^T\left(q_{d+1}^{2\nu}\left(\Lambda_c^2\right)\Lambda_c^2 + K_f^T q_{d+1}^{2\nu}\left(\Lambda_f^2\right)\Lambda_f^2 K_f\right)\boldsymbol{a}_i}.\end{aligned} \tag{6.37}$$

□

This time, the eigenvalues involved in the contraction rate of Theorem 6 are not only squared, but also weighted by the damping factors of the Chebyshev polynomial

smoother. In particular, the polynomial smoother introduced in Chapter 4 has been especially designed to damp the largest eigenvalues in magnitude. This feature is of particular interest in this case because the entries K_f associated with the largest eigenvalues of Λ_f are those degrading the contraction rate the most. The following example illustrates the benefits of this feature on the simple 2×2 example of Section 3.2.

Example 3. *Consider again Example 2, where A is a 2×2 matrix, with $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ its eigenvectors respectively associated with eigenvalues $|\lambda_1| < |\lambda_2|$. Recall from* (3.84)*, that the interpolation operator P targets $\boldsymbol{v}_1$ as follows*

$$P = \boldsymbol{v}_1 + \epsilon \boldsymbol{v}_2 \quad \textit{and} \quad \boldsymbol{v}_1^T \Pi_A (P) \, \boldsymbol{v}_1 = \frac{\lambda_1}{\lambda_1 + \epsilon^2 \lambda_2}. \tag{6.38}$$

Let W_1 be the minimization space of dimension 1 *containing a single smoothed coarse correction vector, such that*

$$W_1 = q_{d+1}(A^2) \Pi_A(P) \boldsymbol{v}_1. \tag{6.39}$$

In addition, let Z_1 and H_1 be the associated Arnoldi operators as prescribed by equations (6.13) *and* (6.14) *respectively. From Theorem 6, the alternative coarse correction contracts $\boldsymbol{v}_1$ at a rate*

$$\boldsymbol{v}_1^T \left(I - W_1 H_1^{-1} Z_1^T A \right) \boldsymbol{v}_1 = 1 - \frac{q_{d+1}^2(\lambda_1^2) \lambda_1^2}{q_{d+1}^2(\lambda_1^2) \lambda_1^2 + q_{d+1}^2(\lambda_2^2) \epsilon^2 \lambda_2^2}. \tag{6.40}$$

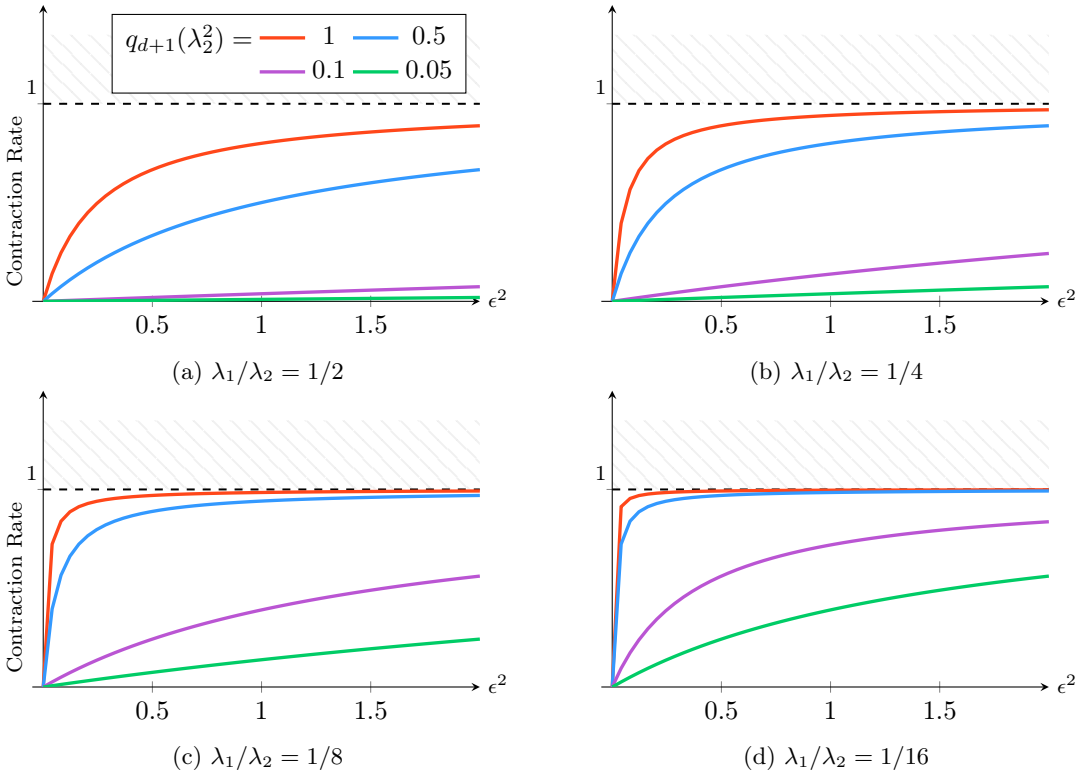
For ease of discussion, let us assume that the smallest eigenvector $\boldsymbol{v}_1$ is preserved by the smoother, such that $q_{d+1}(\lambda_1^2) = 1$ (recall that this assumption would be true for $\lambda_1 = 0$). The contraction rate can be written as a function of λ_1/λ_2, such that

$$\boldsymbol{v}_1^T \left(I - W_1 H_1^{-1} Z_1^T A \right) \boldsymbol{v}_1 = 1 - \frac{1}{1 + q_{d+1}^2(\lambda_2^2) \epsilon^2 \left(\lambda_2 / \lambda_1 \right)^2}. \tag{6.41}$$

Figure 6.4 illustrates the contraction of $\boldsymbol{v}_1$ after applying the alternative coarse correction with respect to the pollution and the polynomial smoother. The red curves are equivalent to the contraction rate prescribed by Theorem 5 where no additional smoothing step of the coarse correction vector is applied. Conversely, the three other curves represent the contraction rates for different ratios of the smallest eigenvalue over the largest, and when the coarse correction vector is smoothed by the Chebyshev polynomial.

As expected, the smoother improves the contraction by counterbalancing the squared large eigenvalue λ_2 that weights the pollution. Moreover, Figure 6.4 exhibits the importance of the additional smoothing step of the coarse correction vector as the gap between the smallest and the largest eigenvalues increases.

The filtering of large eigenvectors aimed at increasing the prevalence of the smallest ones is enabled because the Chebyshev polynomial smoother does not hit the near-zero eigenvalues. By contrast, applying GMRES to the coarse correction vectors of W_p may decrease the prevalence of small eigenvectors as it has no guarantee to preserve them due to the right-hand side dependency. This feature is another argument that motivated the development of Chebyshev polynomials smoother in Chapter 4.



(a) $\lambda_1/\lambda_2 = 1/2$ (b) $\lambda_1/\lambda_2 = 1/4$

(c) $\lambda_1/\lambda_2 = 1/8$ (d) $\lambda_1/\lambda_2 = 1/16$

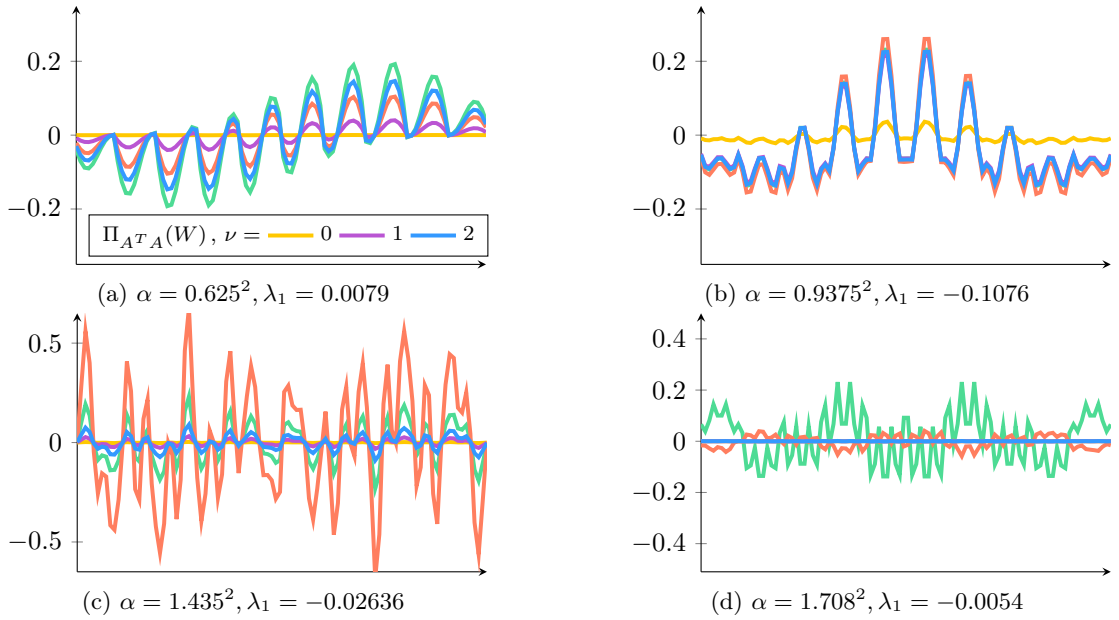
Figure 6.4: Contraction of the small eigenvector $\boldsymbol{v}_1$ with the alternative coarse correction with respect to the pollution and the polynomial smoother

6.2.4 Numerical experiments on the alternative coarse correction

Let us now apply one iteration of the alternative coarse correction to the two-dimensional shifted Laplacian problem with respect to the number ν of smoothing steps of the coarse correction vector. Here again, we target the smallest eigenvector $\boldsymbol{v}_1$ as it is the most critical component. After one cycle, the minimization space W_1 contains only one smoothed coarse correction vector, as prescribed by (6.32). Figure 6.5 plots the approximations returned by the alternative coarse correction depending on four different shifts α and several values of ν. As in the previous figures 6.1a and 6.1c, we use the ideal approximation operator $\hat{P}$ with the least-squares minimization strategy. For each figure of 6.5, the green curve corresponds to the target smallest eigenvector $\boldsymbol{v}_1$, whereas the red curve designates the approximation $\Pi_A(P)\boldsymbol{v}_1$ returned by the classical coarse correction. The same green and red curves corresponding to the shifts $\alpha = 0.625^2$ and $\alpha = 1.708^2$ also appear in figures 6.1a and 6.1c dedicated to the amplification phenomena of the classical coarse correction.

On the first hand, Figure 6.5 shows that the yellow curves are completely flat on three of the four figures. Such poor approximations of $\boldsymbol{v}_1$ are due to the lack of smoothing that counterbalances the squaring of eigenvalues in the contraction rate formula (6.24). On the other hand, adding more smoothing steps improves the approximation drastically, except on Figure 6.5d where the approximations remain near zero. Another important remark regards the shifted case $\alpha = 1.435^2$ illustrated in Figure 6.5c. In that case, the approximation returned by the classical coarse

correction in red is amplified; and also leads to divergence. By contrast the alternative coarse correction recovers the solution as ν increases. Additionally, it is worth noting that $\nu = 2$ smoothing steps of the coarse correction vector for $\alpha = 0.625^2$ returns a better approximation of $\boldsymbol{v}_1$ than the classical coarse correction.



(a) $\alpha = 0.625^2, \lambda_1 = 0.0079$ (b) $\alpha = 0.9375^2, \lambda_1 = -0.1076$

(c) $\alpha = 1.435^2, \lambda_1 = -0.02636$ (d) $\alpha = 1.708^2, \lambda_1 = -0.0054$

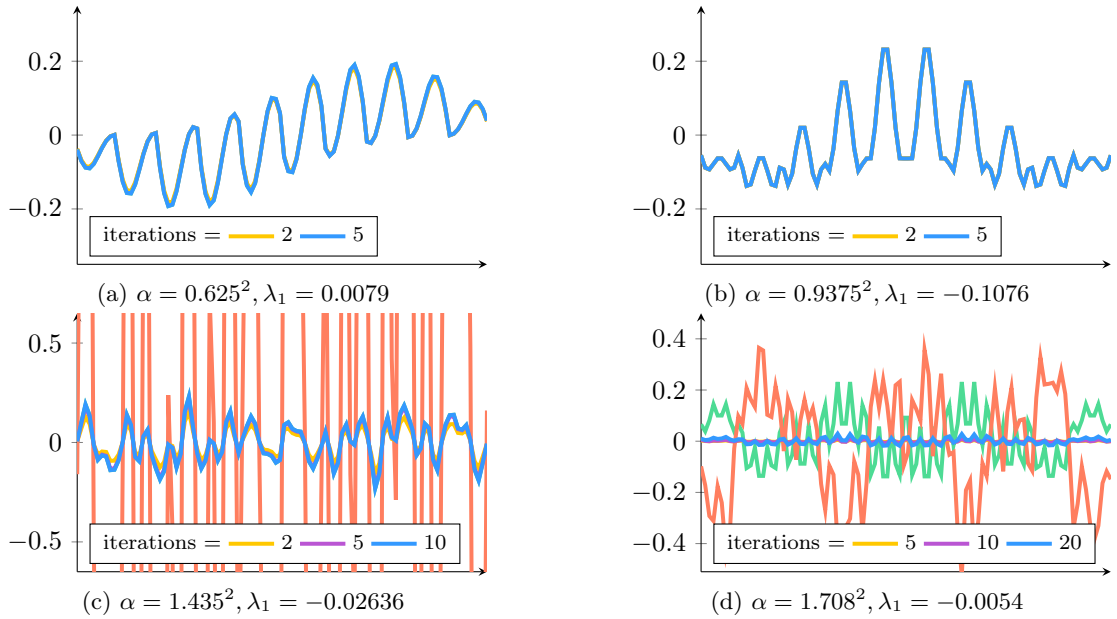
Figure 6.5: Smallest eigenvector $\boldsymbol{v}_1$ vs. approximations returned by the alternative coarse correction for $W_1 = q_{m+1}^{\nu}(A^2)\Pi_A(P)\boldsymbol{v}_1$ with respect to ν and α

More observations can be made by looking at the eigenvalue of each target smallest eigenvector. The eigenvalues λ_1 associated with each $\boldsymbol{v}_1$ are printed in their respective captions. In particular, we remark that the value of λ_1 for $\alpha = 0.9375^2$ of Figure 6.5b is the largest among the four. This explains the small oscillations of the yellow curve that are perfectly synchronized on those of $\boldsymbol{v}_1$ in green, even though no smoothing is applied to the coarse correction vector. Such an approximation is not really satisfying for good convergence in practice, but it exhibits the correlation between the approximation accuracy of the small eigenvectors and their associated eigenvalues in magnitude. Conversely, the smallest eigenvalues among the four are for $\alpha = 0.625^2$ and $\alpha = 1.708^2$. While additional smoothing steps help the contraction in the former case, all of the three approximations in the latter remain around zero due to larger entries of the pollution block K_f (one can compare the pollution for $\alpha = 0.625^2$ and $\alpha = 1.75^2$ in figures 5.17 and 5.18 respectively).

To summarize, the alternative coarse correction remedies the problem of divergence, and enables fast convergence even for certain cases where the classical coarse correction amplifies the error, such as the shifted case $\alpha = 1.435^2$. For certain other cases, the alternative coarse correction still suffers from the pollution effect, as for $\alpha = 1.708^2$. Even though the interpolation operator has good approximation properties for $\boldsymbol{v}_1$ as illustrated in Figure 6.1c, the pollution may remain too big for very small eigenvalues. In that case, the pollution affects both coarse correction methods

dramatically. Whereas the classical coarse correction eventually amplifies the error and leads to divergence, the alternative one remains slow at recovering the solution. Several options exist to accelerate the convergence in such a critical case, but at the cost of computational complexity. A first approach consists of increasing the exponent ν in (6.33) to decreases the damping factors $q_{d+1}^{2\nu}(\Lambda_f^2)$ since the smoother is convergent (see Section 2.1.2.2 of Chapter 4). A second approach consists of increasing the polynomial degree d. This technique helps decrease the oscillations of the error propagation function on large eigenvalues and consequently brings the diagonal entries of $q_{d+1}^{2\nu}(\Lambda_f^2)$ closer to zero as well. Both ideas help decrease the contraction rate in (6.33), and therefore speed-up the convergence of the method.

In practice, the multigrid cycle is iterated several times, such that the minimization space of the alternative coarse correction contains more than a single vector. While our discussion of the single iteration case enhances the understanding of the coarse correction, adding more vectors naturally improves the approximation of the solution. In fact, the general form of W_p as multigrid is iterated is given by (6.16) and (6.17). The following Figure 6.6 concludes this discussion by iterating the multigrid method with the alternative coarse correction in the same setting as Figure 6.5. The number of smoothing steps is set to $\nu = 2$, and the number of iterations goes from $p = 2$ to $p = 50$. Again, we let the solution be $\boldsymbol{x} = \boldsymbol{v}_i$ to better identify the behavior of the method on the most critical eigenvector. In addition, the red curve represents the approximation returned by the multigrid cycle with the classical coarse correction after 5 iterations.



(a) $\alpha = 0.625^2, \lambda_1 = 0.0079$ (b) $\alpha = 0.9375^2, \lambda_1 = -0.1076$

(c) $\alpha = 1.435^2, \lambda_1 = -0.02636$ (d) $\alpha = 1.708^2, \lambda_1 = -0.0054$

Figure 6.6: Smallest eigenvector $\boldsymbol{v}_1$ vs. approximations returned by multigrid with classical coarse correction vs. approximations returned by multigrid with the alternative coarse correction for W_p with respect to the number p of multigrid cycles and the shift α - $\nu = 2$

While multigrid with the classical coarse correction recovers the smallest eigenvector $\boldsymbol{v}_1$ for $\alpha = 0.625^2$ (see Figure 6.6a) and $\alpha = 0.9375^2$ (see Figure 6.6b), the approximation gets even worse as the number of iterations p increases for $\alpha = 1.435^2$ (see

Figure 6.6c) and $\alpha = 1.708^2$ (see Figure 6.6d). Regarding multigrid with the alternative coarse correction, the approximation curves fit with the solution after 5 iterations, except in the critical case $\alpha = 1.708^2$ where the method barely oscillates even after 20 iterations. More numerical experiments with random solutions $\boldsymbol{x}$ are discussed in the next chapter.

6.3 Intermediate conclusion

The first part of this chapter illustrated how the coarse correction can amplify the error and lead to divergence. The concept of pollution of interpolation introduced in Section 3.2 enables a better understanding of this issue. In particular, we demonstrated that small eigenvectors are the most impacted, and can get easily amplified by the coarse correction although the interpolation operator has good approximation properties and the pollution is small.

To address this issue, we introduced an alternative coarse correction based on an Euclidean norm minimization. Our solution is to augment a coarse correction subspace at each multigrid cycle with the vector returned by the classical coarse correction. At each iteration, the residual is therefore minimized in Euclidean norm within the coarse correction subspace. We also show that resorting to the Euclidean norm is equivalent to working with normal equations. Although it fixes the problem of divergence, it also affects the contraction of small eigenvalues. To address this issue, each coarse correction vector of the minimization subspace is smoothed with the polynomial smoother introduced in Chapter 4. More experiments on the overall method can be found in the next chapter.

Chapter 7

Numerical Experiments

The previous chapters introduced three standard multigrid components in the context of solving the Helmholtz equation : a smoother aimed at damping the large eigenvalues in magnitude, an interpolation operator whose range approximates the oscillatory space of small eigenvectors, and an alternative coarse correction that does not amplify the error. This chapter presents more numerical experiments, and is split into two parts. The first one applies our two-level method to the two-dimensional shifted Laplacian model problems of (1.2) and (1.3). The second part presents the multilevel case and several benchmarks on a two-dimensional Helmholtz problem with absorbing boundary conditions.

In this chapter, the smoother corresponds to the Chebyshev polynomial built on the normal equations, and its interval is determined by the spectral density approximation method presented in Section 4.1.1, such that the number n_μ of coefficients μ_j in the moment matching procedure in (4.6) is fixed to 15, and n_{vec} fixed to 5. The degree d of the polynomial is set to 3. Regarding the construction of the least-squares coarse variable operator $\hat{R}^T$, the number of smoothed test vectors in (5.17) is fixed to $\kappa = 15$, and the number of interpolation points in (5.24) never exceeds 4 (i.e., $\max_{i \in \mathcal{F}} \{\text{Card}(C_i)\} = 4$). For each following experiment, the given parameter ν also determines the number of smoothing steps of the test vectors and the number of smoothing steps of the coarse correction vectors contained in the minimization space of our alternative coarse correction.

7.1 The Two-level case

This section presents the results of different two-level methods on the two-dimensional shifted Laplacian matrices (1.2) and (1.3). The figures below show the number of iterations with respect to the shift α and depending on the type of coarse correction and the number of smoothing steps ν.

Since the number of iterations is presented with respect to the shift α, let us recall that the matrix is the most indefinite when $\alpha = 2^2$ for the 5-point stencil discretization matrix (1.2), in the sense that the spectrum reaches a balance between negative and positive eigenvalues. In the same way, the matrix is highly indefinite around $\alpha = 3^2$ for the 9-point stencil discretization matrix (1.3). Moreover, the near-kernel

space becomes more oscillatory as α increases. Lastly, we measure the cost of each method through the two-grid operator complexity ϕ_{TG} defined by

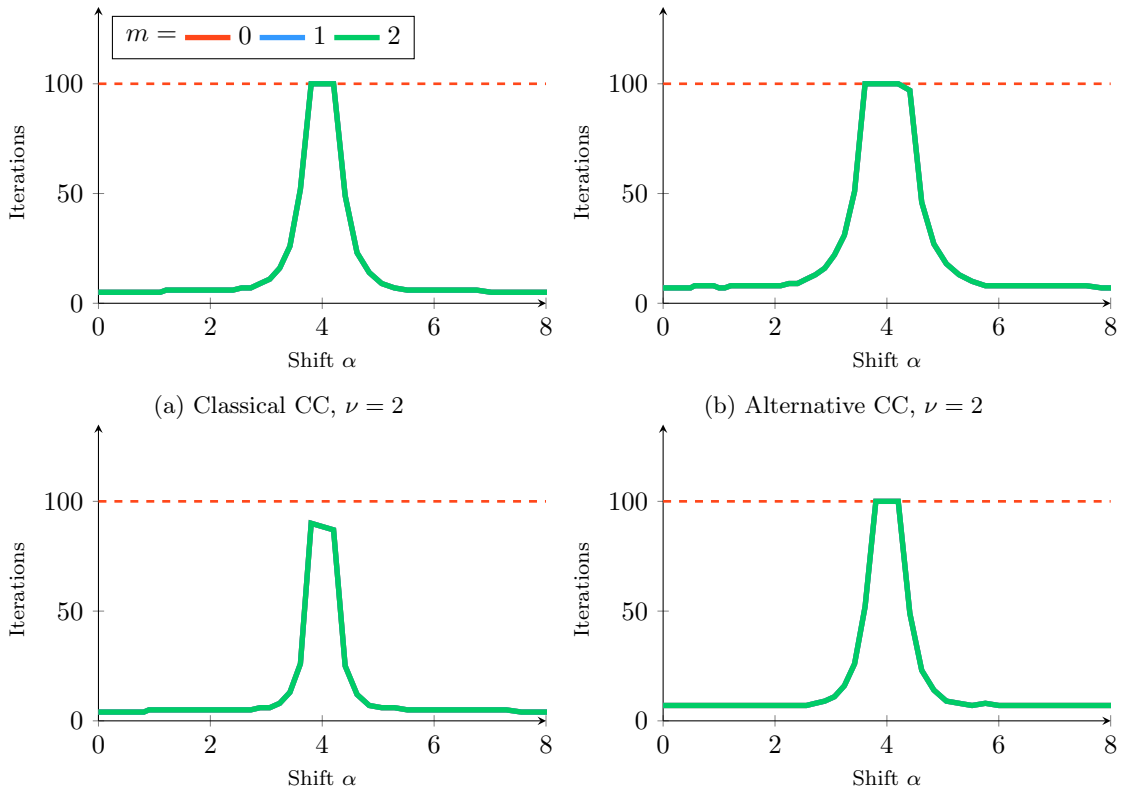
$$\phi_{\text{TG}} := \frac{\text{nnz}(A) + \text{nnz}(A_c)}{\text{nnz}(A)} \qquad \text{with} \qquad A_c = P^T AP. \tag{7.1}$$

Naturally, the method is cheaper when ϕ_{TG} is closer to 1. As multigrid works by iterating the cycle until convergence, we set the tolerance of the relative residual norm to 10^{-6}, and the maximal number of iterations to 100. While we use normal equations in the multilevel extension presented in the next section, we stick to the initial matrix in the approximation of ideal interpolation as no substantial difference in terms of convergence appeared when running the numerical experiments in the two-level case.

7.1.1 Benchmarks on the 5-point stencil shifted Laplacian matrix

7.1.1.1 Classical variable operators with SPAI

The Figure 7.1 plots the number of iterations of four two-level methods applied to the 5-point stencil discretization matrix (1.2). In this first experiment, the interpolation operator is an approximation of the classical ideal interpolation operator P_* of Section 5.2.2. The number of iterations and the operator complexity of Table 7.1 are given with respect to the pattern augmentation degree m.



(a) Classical CC, $\nu = 2$

(b) Alternative CC, $\nu = 2$

(c) Classical CC, $\nu = 4$

(d) Alternative CC, $\nu = 4$

Figure 7.1: Number of iterations of two-level methods using the SPAI to approximate the ideal interpolation from the classical variable operators R^T and S - The 5-point stencil case

In the 5-point stencil case, the classical fine variable block $A_{ff} = S^T AS$ is strictly diagonal. Therefore, the SPAI technique computes its inverse exactly and the interpolation operator is ideal regardless of the pattern augmentation degree m. This convenient structural property implies that all the three curves in Figure 7.1 fit perfectly, and appear blended in a single green curve. However, in a multilevel setting, the fine variable block does not remain diagonal throughout the grid hierarchy because the pattern of coarse matrices are unlikely to have the same structure, and probably contain more non-zero entries. For this reason, the next section applies the same two-level setting to a 9-point stencil discretization matrix to prevent our analysis from any bias relative to this convenient structural property.

None of the four experiments with the classical ideal interpolation operator in Figure 7.1 diverged, regardless of the indefiniteness of the 5-point stencil matrix. The peaks in Figure 7.1a, Figure 7.1b and Figure 7.2d reach the maximal number of iterations because the convergence is too slow but not because the coarse correction amplified the error. Alternatively, the third Figure 7.2c shows that increasing the number of smoothing steps to $\nu = 4$ enables the two-level method with the classical ideal interpolation operator to converge in less than 100 iterations for all shifts. As we will see further with experiments on the 9-point stencil matrix, this behavior is exceptional, and happens only because the initial matrix in this case enables a practical ideal interpolation operator that the SPAI approach can perfectly match. While the error in not amplified by the coarse correction in this case, we recall the method may diverge although P is ideal, as Figure 6.1d illustrates. Lastly, we observe that the values of ϕ_{TG} are all the same in Table 7.1, because the interpolation operator is ideal for all values of m in this particular case.

m	0	1	2
ϕ_{TG}	1.81	1.81	1.81

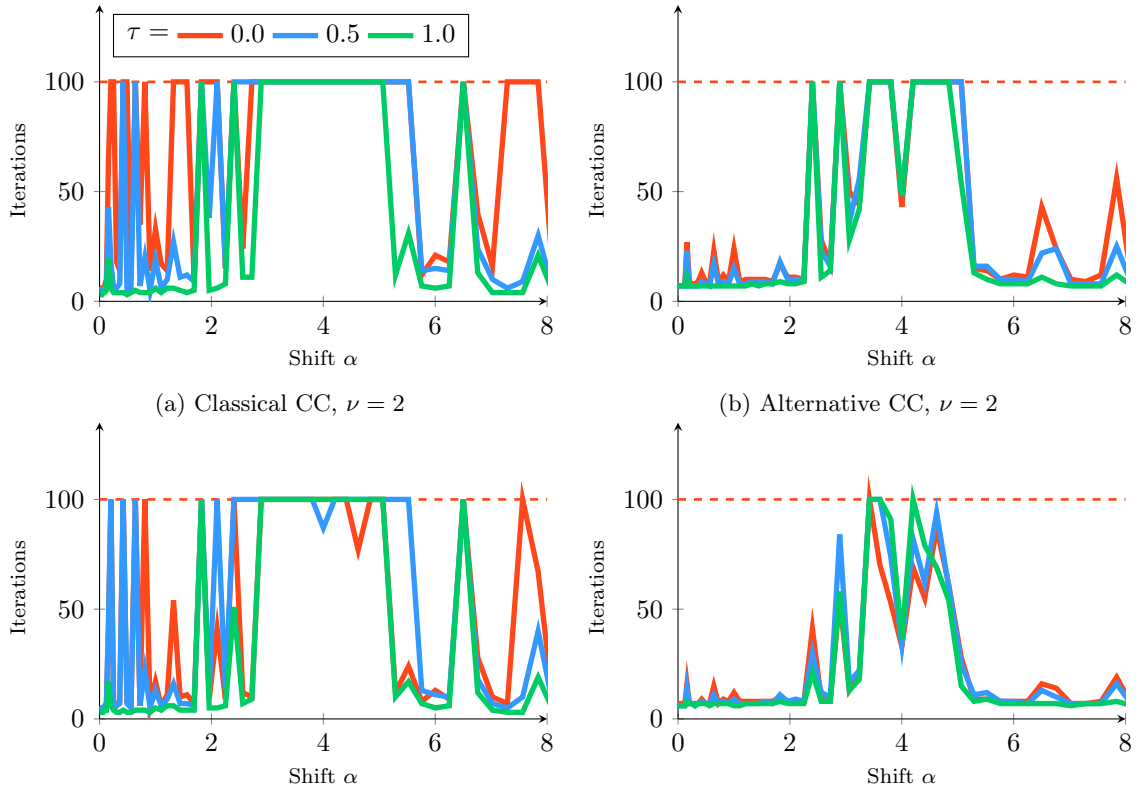
Table 7.1: Operator complexity of the two-level method using the SPAI approach to approximate the ideal interpolation from the classical variable operators R^T and S with respect to m - The 5-point stencil case

7.1.1.2 Least-squares variable operators with the subspace restriction approach

By contrast, Figure 7.2 plots the number of iterations of the two-level method using the least-squares variable operators (5.29). Here, the ideal approximation is computed by way of the subspace restriction approach, of in Section 5.3.3.

This setting does not benefit from the underlying structure of the initial matrix A, contrary to the classical setting illustrated in Section 7.1.1.1. This time, the peaks on the left figures 7.2a and 7.2c represent divergence.

While most failures arise around $\alpha = 2^2$, both red and blue curves reach the maximal number of iterations for intermediate shifts as well. Increasing the fine variable minimization subspace to $\tau = 1.0$ helps in certain cases, but it is not always sufficient. The most dramatic improvement provided by the alternative coarse correction



(a) Classical CC, $\nu = 2$

(b) Alternative CC, $\nu = 2$

(c) Classical CC, $\nu = 4$

(d) Alternative CC, $\nu = 4$

Figure 7.2: Number of iterations of two-level methods using the subspace restriction approach to approximate the ideal interpolation from the least-squares variable operators $\hat{R}^T$ and $\hat{S}$ - The 5-point stencil case

illustrated in Figure 7.2b and 7.2d appears to be for sudden peaks arising for intermediate values of τ. Moreover, increasing the number of smoothing iterations helps, as it also improves the minimization space of the alternative coarse correction. Although the alternative correction prevents divergence, it does not fix the slow convergence issue that we notice for highly indefinite problems. Lastly, the operator complexity given by Table 7.2 grows quickly with τ when coupling the least-squares variable operators with the subspace restriction approach.

τ	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
ϕ_{TG}	1.81	3.00	3.47	3.69	3.93	4.16	4.35	4.55	4.73	4.87	5.15

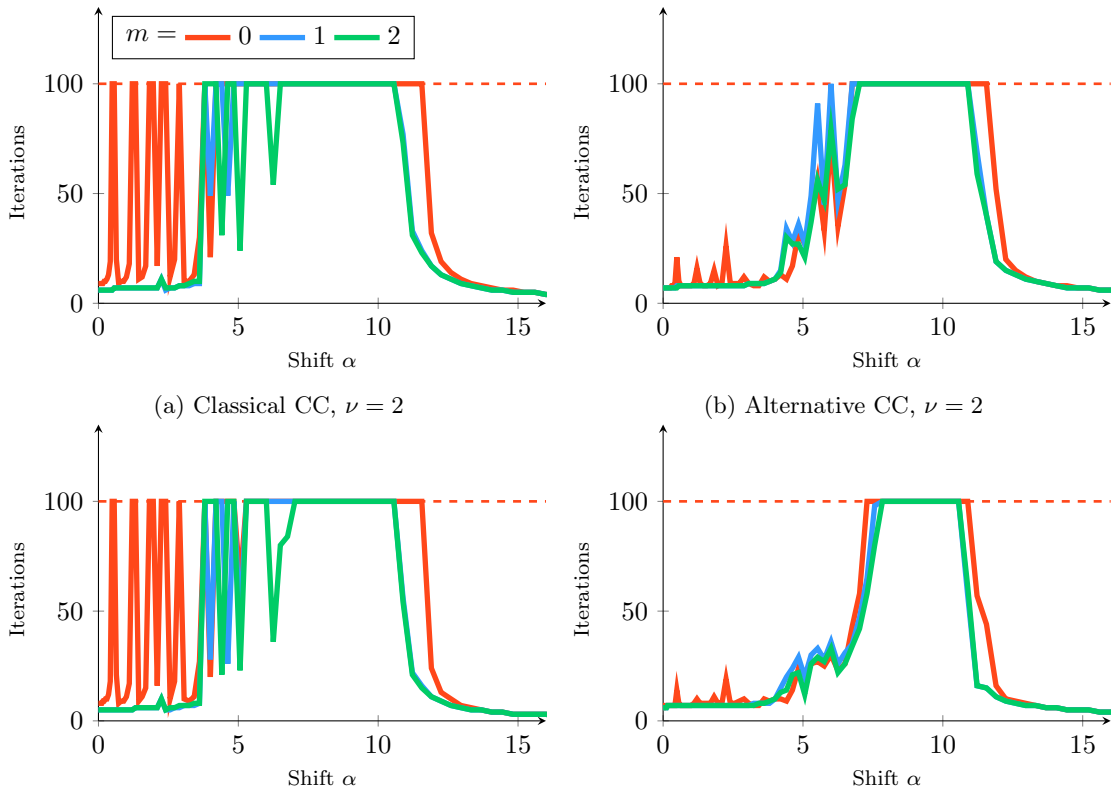
Table 7.2: Operator complexity of the two-level method using the subspace restriction approach to approximate the ideal interpolation from the least-squares variable operators $\hat{R}^T$ and $\hat{S}$ with respect to τ - The 5-point stencil case

7.1.2 Benchmarks on the 9-point stencil shifted Laplacian matrix

7.1.2.1 Classical variable operators with SPAI

With the aim of challenging our algebraic two-level methods, it is more appropriate to avoid the convenient structural properties from which benefits the SPAI approach

when resorting to the classical variable operators R^T and S. Hence, we also benchmark our methods on the 9-point discretization matrix whose stencil is defined in (1.3). In this case, the A_{ff} matrix is not diagonal. As a consequence, SPAI returns an inexact approximation of the ideal interpolation.



(a) Classical CC, $\nu = 2$

(b) Alternative CC, $\nu = 2$

(c) Classical CC, $\nu = 4$

(d) Alternative CC, $\nu = 4$

Figure 7.3: Number of iterations of two-level methods using the SPAI to approximate the ideal interpolation from the classical variable operators R^T and S - The 9-point stencil case

Contrary to the 5-point stencil matrix characterized by convenient structural properties, these red curves reveal that the two-level method diverges when applying the classical coarse correction, even for small values of α. Comparing figures 7.3a and 7.3c, it appears that adding more smoothing steps does not address the divergence. Increasing the pattern of non-zero entries helps for certain small values of the shift α as it brings P closer to the ideal interpolation operator and therefore decreases the pollution K_f. However, it is still not sufficient to fix the amplification of the coarse correction in all cases.

By contrast, figures 7.3b and 7.3d show that the divergence peaks arising for small values of α are fixed by the alternative coarse correction. Moreover, all three curves have approximately the same shape. The main difference is that the number of iterations associated with $m = 0$ slightly increases for values of α where the classical setting diverges. Increasing m flatten the curves, and enables solving the system in fewer iterations. Doubling the smoothing steps does not accelerate the convergence significantly in this case.

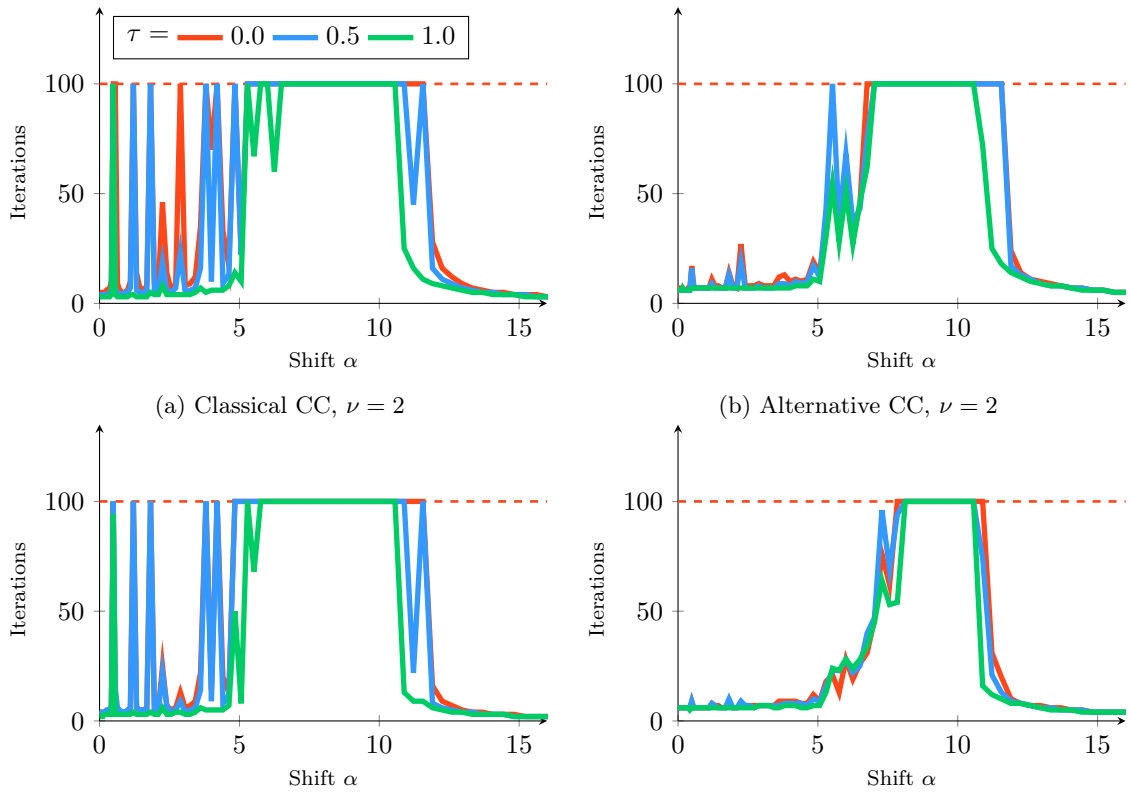
Finally, Table 7.4 presents the operator complexity of this two-level setting with respect to m. While increasing m from 0 to 1 reduces iterations but increases the number of non-zero entries in P, the structure remains the same for $m = 2$, probably because the ideal approximation P is already close to its ideal form P_*.

m	0	1	2
ϕ_{TG}	2.07	2.83	2.83

Table 7.3: Operator complexity of the two-level method using the SPAI approach to approximate the ideal interpolation from the classical variable operators R^T and S with respect to m - The 9-point stencil case

7.1.2.2 Least-squares variable operators with the subspace restriction approach

Lastly, Figure 7.4 plots the number of iterations with respect to α when using the least-squares variable operators with the subspace restriction approach in the 9-point stencil case.



(a) Classical CC, $\nu = 2$

(b) Alternative CC, $\nu = 2$

(c) Classical CC, $\nu = 4$

(d) Alternative CC, $\nu = 4$

Figure 7.4: Number of iterations of two-level methods using the subspace restriction approach to approximate the ideal interpolation from the least-squares variable operators $\hat{R}^T$ and $\hat{S}$ - The 9-point stencil case

Here again, figures 7.4a and 7.4c show that the classical coarse correction amplifies the error and leads to divergence even for small shifts. Increasing τ apparently helps

convergence, but the coarse correction remains likely to amplify the error. Certain divergence scenarios when $\nu = 2$ are fixed by doubling the number of smoothing iterations. Doing so improves the set of test vectors in approximating the near-kernel space, and therefore leads to a better least-squares minimization coarse variable operator and participates in decreasing the pollution K_f. Although the best setting for the classical coarse correction is $\tau = 1$ and $\nu = 4$, it remains difficult to derive a general setting that ensures the convergence of the standard method with the classical coarse correction in all cases. The alternative coarse correction is necessary, even though the convergence remains too slow for highly indefinite problems. Figures 7.4b and 7.4d represent the same experiment with the alternative coarse correction. The peaks around $\alpha = 3^2$ show a slow convergence because of near-zero eigenvalues. Except for these extremely indefinite cases, the method converges. Moreover, increasing τ or ν provides a better convergence rate, but at the cost of complexity. The numbers of the operator complexity are given in Table 7.4 with respect to τ. These numbers are smaller than in Table 7.1, because the initial 9-point stencil discretization matrix A is denser and yields a larger denominator in the definition of the operator complexity (7.1).

τ	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
ϕ_{TG}	1.86	2.35	2.51	2.75	2.93	3.15	3.34	3.43	3.69	3.95	4.16

Table 7.4: Operator complexity of the two-level method using the subspace restriction approach to approximate the ideal interpolation from the least-squares variable operators $\hat{R}^T$ and $\hat{S}$ with respect to τ - The 9-point stencil case

7.2 Extension to the Multilevel case

The following numerical experiments demonstrate the convergence of multilevel methods when applied to two-dimensional Helmholtz problems with absorbing boundary conditions. The continuous problem is

$$\text{(Helmholtz + A.B.C.)} \quad \Leftrightarrow \quad \left\{ \begin{array}{rll} -\Delta u - k^2 u & = f & \text{on } \Omega \\ \partial_n u - \iota k u & = 0 & \text{on } \partial\Omega \end{array} \right. . \tag{7.2}$$

In these experiments, the continuous model problem (7.2) is discretized with a finite difference scheme such that the resulting discretization matrix has a 5-point stencil pattern similar to (1.2), plus the coefficients resulting from the discretization of absorbing boundary conditions. The only differences with the shifted Laplacian matrices are the absorbing boundary conditions that prevent the discretization matrix from being singular. As a consequence, the discretization matrix is indefinite, complex and non-Hermitian. The restriction operation is made through the transpose conjugate P^* or $\hat{P}^*$, and the squared matrix in the Chebyshev polynomial smoother is replaced by A^*A. In the numerical experiments that follow, the discretization coefficient kh is set to 0.625 (i.e., 10 points per wavelength) but the wavenumber k varies. Since kh is constant, the mesh size h decreases as k is growing. As a consequence, the matrix size grows with the wavenumber. The size of each matrix level with respect to k is given in Figure 7.5.

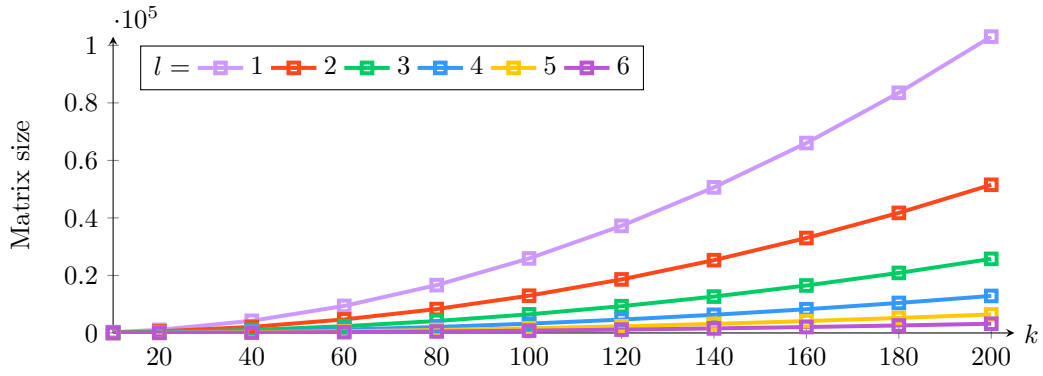


Figure 7.5: Matrix size of each level with respect to the wavenumber k

In this section, we resort to the alternative coarse correction only, where Z^T is replaced by its conjugate counterpart Z^* in (6.11).

In addition, let l be the level index in the hierarchy. Similarly to the two-level experiments of Section 7.1, the cost of the method is discussed with the operator complexity variable given by

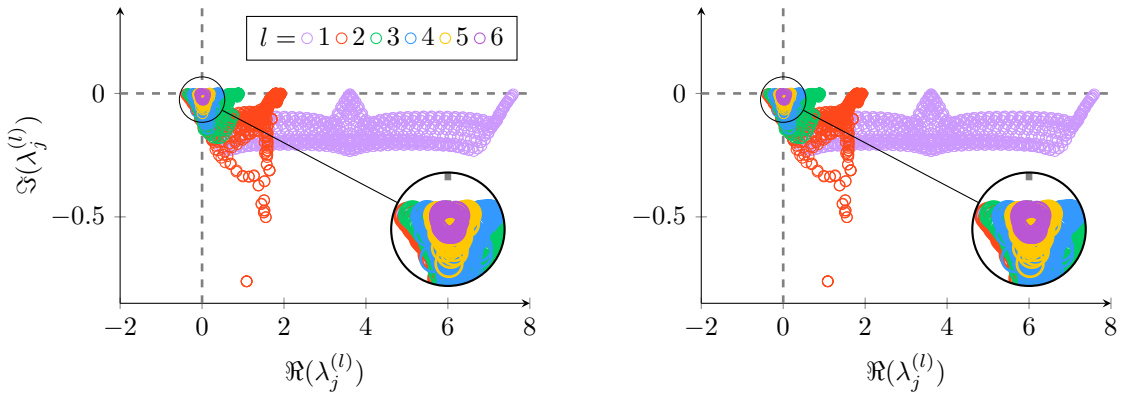
$$\phi_{\text{MG}} := \frac{\text{nnz}(A_1) + \sum_{l=2}^{l_{\max}} \text{nnz}(A_l)}{\text{nnz}(A_1)} \quad \text{with} \quad A_{l+1} = P_{l+1}^T A_l P_{l+1} \quad , \quad A_1 := A. \tag{7.3}$$

Intermediate matrices have more non-zero entries and more complex structures. Therefore, approximations of the ideal interpolation are much harder to generate than in the two-level case. Using normal equations in this context ensures that the ideal interpolation operator satisfies variational properties because A^*A generates a norm. Whereas we omitted normal equations for the approximation of the ideal interpolation operator in the two-level setting, we generally noticed better performance when using them into the ideal approximation phase of our multilevel method. We therefore use normal equations in the ideal approximation step.

7.2.1 Classical Ideal approximations

First, let us try the multigrid method with the interpolation operator resulting from the classical variable operators (5.8) and the sparse ideal approximate inverse (5.16).

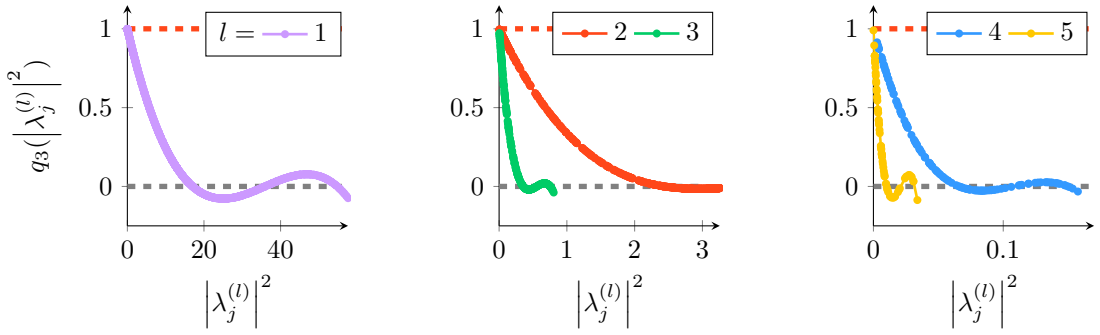
We begin with Figure 7.6 which represents the complex eigenvalues of each level for $k = 20$ and $m \in \{0, 1\}$. Despite the 5-point stencil structure of the initial matrix, the SPAI does not benefit from the diagonal form of A_{ff} because of the normal equations in the design of P. As a consequence, P is not the exact ideal interpolation but an approximation that depends on m. It explains why the eigenvalues of the second level matrices in red are different on both figures. That said, it clearly appears that, as the matrix gets coarser, the spectrum gets more contracted toward the origin. Hence, the proportion of positive and negative real parts reaches an equilibrium from the fourth level, and the imaginary parts decrease magnitude-wise with the level index. In that sense, we can say that each new coarse matrix is more indefinite than its upper parent in the hierarchy.



(a) $m = 0$ (b) $m = 1$

Figure 7.6: Eigenvalues for each level with respect to m for $k = 20$ and $kh = 0.625$

As explained in Section 4.1.2, designing a polynomial smoother with normal equations is especially relevant for highly indefinite matrix. Since the polynomial smoother of each level is constructed algebraically, let us study their damping factors with respect to the eigenvalues of each level. The polynomial of the finest level is illustrated in Figure 7.7a, whereas polynomials of intermediate levels are illustrated in Figure 7.7b and 7.7c. The y-axis corresponds to the damping factor with respect to the squared modulus of the eigenvalues represented by the x-axis.

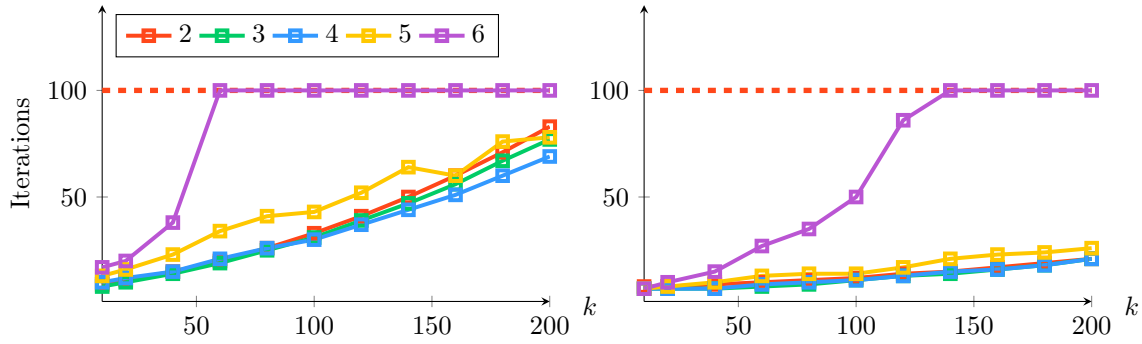


(a) Finest level (b) Levels 2 and 3 (c) Levels 4 and 5

Figure 7.7: Damping factors of the Chebyshev polynomials for each level with respect to m for $k = 20$ and $kh = 0.625$

The number of generated Chebyshev roots is set to $d + 1 = 3$ in these experiments. Therefore, the degree of the polynomial error propagation matrix is also 3. Except the polynomial represented by the red curve, we notice that all of the other polynomials have their smallest root located before half of the largest magnitude eigenvalue. This observation is due to the spectral density approximation phase that defines the appropriate interval of eigenvalues to be damped by the smoother. Setting the lower-bound of the interval under the assumption that the spectrum is uniformly distributed can either lead to a narrower interval where intermediately large eigenvalues are least damped, or lead to a slower capture of the largest eigenvalues due to more oscillations within a wider interval than necessary.

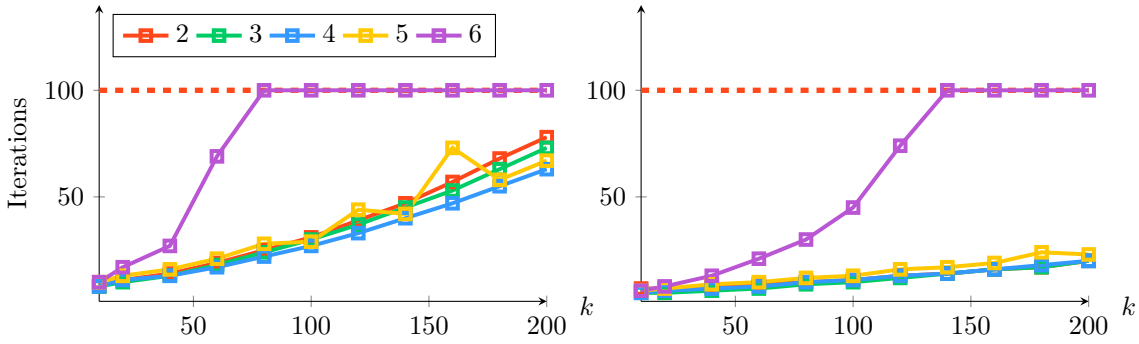
Finally, Figure 7.8 illustrates the number of iterations with respect to the wavenumber k and depending on the number of levels. Figure 7.8a corresponds to the case where no pattern extension is applied (i.e., $m = 0$), whereas Figure 7.8b shows the convergence when $m = 1$. We first observe that the number of iterations generally increases as more levels are added. This observation is not uncommon in classical multigrid methods in the SPD case, since increasing the number of levels is equivalent to recursively approximating the inverse of the coarsest matrix by an additional two-level method. However, this general trend has no guarantee to hold in every cases in our setting, for instance for $k = 140$ in Figure 7.8a where the five-level method converges faster than its three levels counterpart. In fact, the amplification of small eigenvectors by the classical coarse correction may be more important as more levels are added, which may benefit the alternative coarse correction by increasing their prevalence in the minimization space. Ideally, we want the iteration count to be bounded, but we notice that the number of iterations grows with the wavenumber k, although increasing m accelerates the convergence and leads to a slower growth. Despite the gain provided by augmenting the pattern to $m = 1$, the purple curve of the six level method reaches the dashed red line. In other words, the six-level methods failed to converge in less than 100 iterations for larger values of k in both cases. This issue is probably because the SPAI approach struggles finding a relevant sparse approximation of P_* as coarse matrices contain more non-zero entries. As we observe failure for 6 levels, we did not consider adding more levels in these experiments.¨



(a) $m = 0$ (b) $m = 1$

Figure 7.8: Number of iterations with respect to the wavenumber k and m, for $\nu = 2$

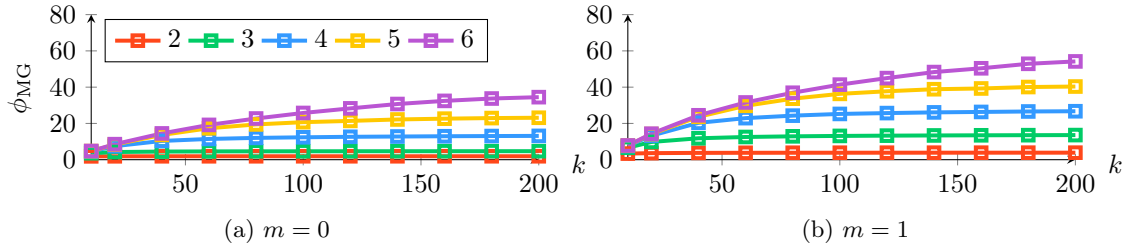
Figure 7.9 shows that doubling the number of smoothing steps slightly accelerates the convergence, but the general trends observed in Figure 7.8 remain the same. We remind that increasing ν not only damps large magnitude eigenvectors faster, but also improves the prevalence of small eigenvectors within the minimization space of the alternative coarse correction by refining the coarse correction vectors. We refer to Section 6.2 for more details on our alternative coarse correction. Lastly, we remark that the convergence rate slightly decreases with k when $m = 1$, but still reaches the maximal number of iterations when resorting to 6 levels.



(a) $m = 0$ (b) $m = 1$

Figure 7.9: Number of iterations with respect to the wavenumber k and m, for $\nu = 4$

Finally, Figure 7.10 plots the operator complexity ϕ_{MG} defined in (7.3). While setting $m = 1$ gives good convergence up to 5 levels, it is at the cost of more expensive operators. Improving the sparsity of the coarse levels is a topic of future research.



(a) $m = 0$ (b) $m = 1$

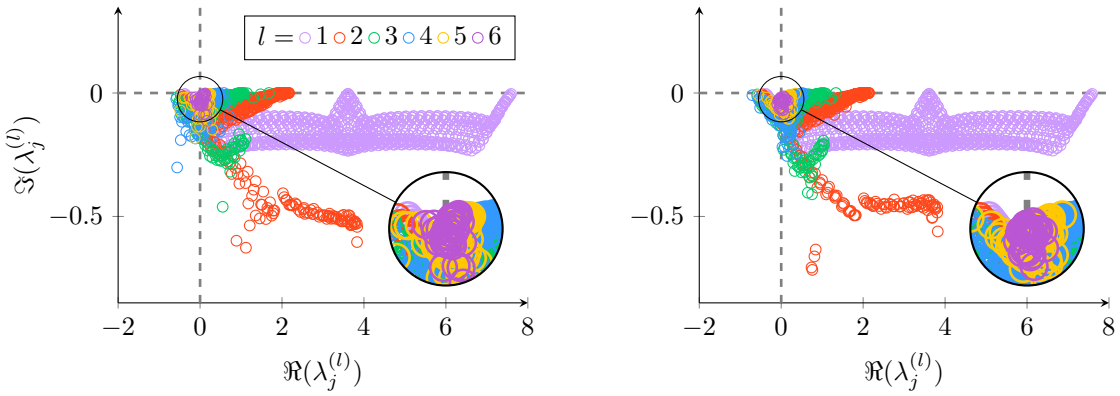
Figure 7.10: Operator complexity with respect to the wavenumber k and m

7.2.2 Least-squares variable operators plus Ideal Subspace Restriction

We end this chapter with the interpolation operator $\hat{P}$ that relies on the least-squares variable operators $\hat{R}^T$ and $\hat{S}$ introduced in Section 5.3.2. While the set of test vectors used in the least-squares minimization strategy is initialized randomly on the finest level, we choose each coarse level to proceed from the restriction of its fine level parent set in the hierarchy. Thereafter, these coarse test vectors are smoothed by the Chebyshev polynomial smoother as usual. Our motivation is that each generated set is an approximation of the near-kernel space, and each new interpolation is designed to capture it. Subsequently, restricting the fine level parent test vectors always provides a better initial guess than initializing the new coarse set randomly.

Secondly, the ideal approximation operator $\hat{P}$ results from the ideal subspace restriction approach of Section 5.3.3. Whereas the previous figures are plotted with respect to m, we now apply our method with respect to τ. Let us recall that τ controls the number of selected columns of $\hat{S}$ in the ideal approximation phase. In particular, $\tau = 0.0$ means no ideal approximation phase, such that no column of $\hat{S}$ is selected. In fact, $\hat{P} = \hat{R}^T$ in that case. Conversely, we recall that setting $\tau = 1.0$

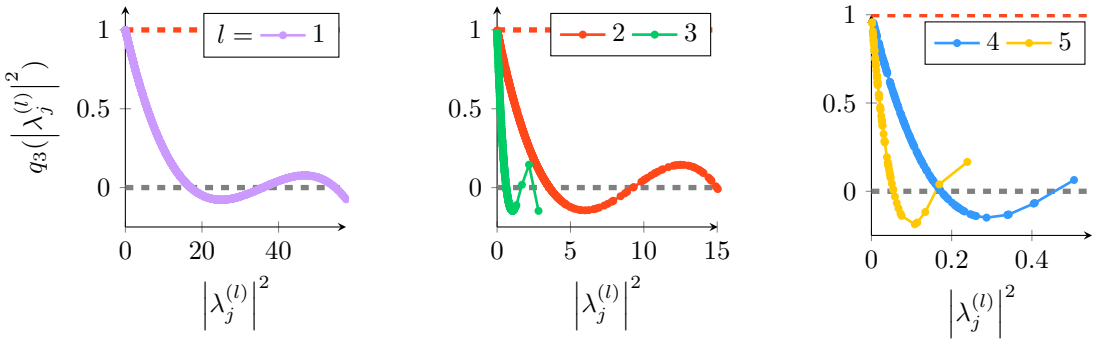
means selecting all the columns of $\hat{S}$ associated with non-zeros in the quantity (5.35). Although $\tau = 1.0$, $\hat{P}$ is not the exact ideal but an inexact approximation of it.



(a) $\tau = 0$ (b) $\tau = 1$

Figure 7.11: Eigenvalues if each level with respect to τ for $k = 20$ and $kh = 0.625$

Let us begin with Figure 7.11, which plots the complex eigenvalues of each matrix in the multigrid hierarchy, for $\tau = 0.0$ and $\tau = 1.0$. The same observations than for 7.6 can be made in this case. The matrix becomes more indefinite with the level index, and the concentration of eigenvalues around zero seems more important when increasing the value of τ. In addition, we also remark that the real part has a larger amplitude in this setting than in Figure 7.6.

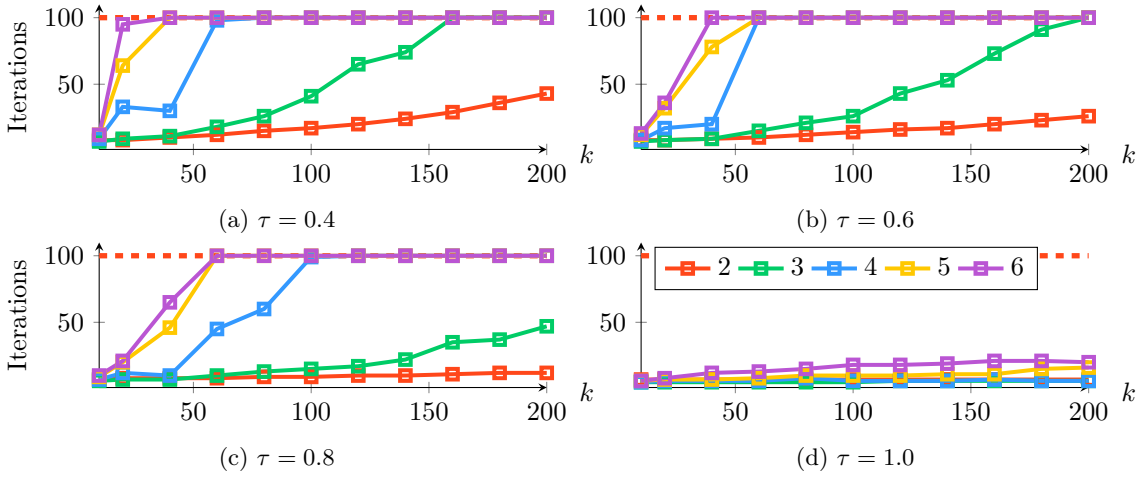


(a) Finest level (b) Levels 2 and 2 (c) Levels 4 and 5

Figure 7.12: Damping factors of the Chebyshev polynomials of each level with respect to τ for $k = 20$ and $kh = 0.625$

The latter observation has an incidence on the shape of the polynomial of Figure 7.12. Figure 7.11 reveals that the spectrum of the second matrix contains eigenvalues of larger magnitude compared to the previous setting. As a consequence, it naturally follows that the interval for generating the Chebyshev roots is larger as well. Increasing the size of the interval means more oscillations of the polynomial within the interval, and results in a slower damping rate for the largest magnitude eigenvalues. Another observation is that the largest magnitude eigenvalues of the last three

levels appear detached from the others. This phenomenon has the same incidence on the shape of the resulting polynomial. For the sake of capturing only a few of these eventually extreme eigenvalues, the polynomial becomes more oscillatory.

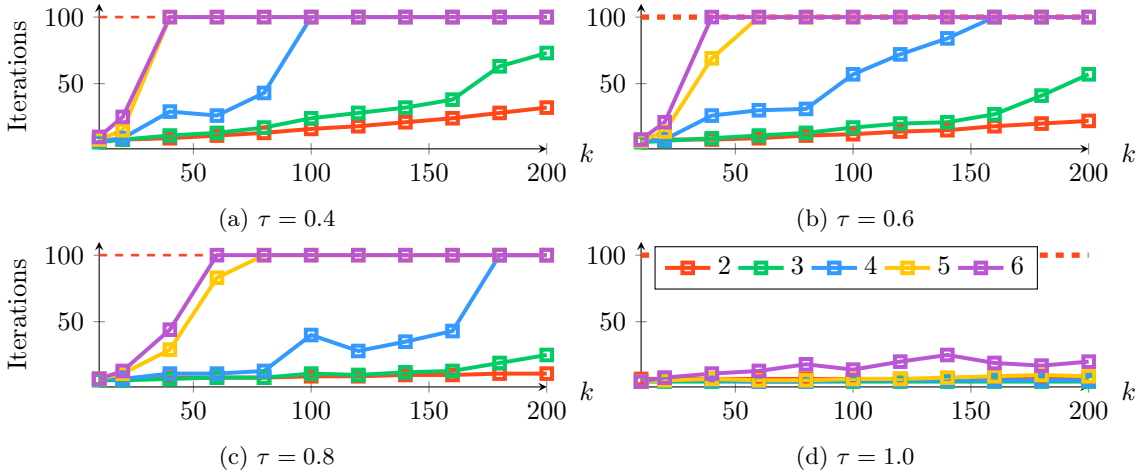


(a) $\tau = 0.4$ (b) $\tau = 0.6$

(c) $\tau = 0.8$ (d) $\tau = 1.0$

Figure 7.13: Number of iterations with respect to the wavenumber k and τ, for $\nu = 2$

Finally, Figure 7.13 shows that the number of iterations grows faster with the wavenumber k when fewer columns of $\hat{S}$ are selected to approximate the least-squares ideal interpolation operator. The number of iterations remains almost constant only when $\tau = 1.0$ as illustrated in Figure 7.13d. Hence, it appears that selecting all the columns associated with a non-zero entries in (5.37) has a strong incidence on the method. Coarse matrices get denser as more levels are added. Hence, the number of selected columns increases and each column may have an equivalent importance. Subsequently, omitting a few of these columns by decreasing τ has more impact on deeper levels.



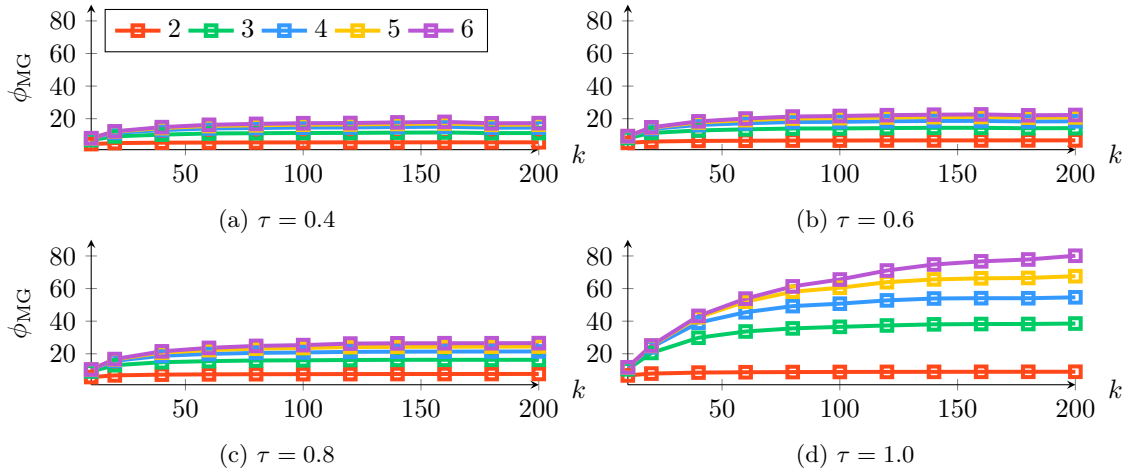
(a) $\tau = 0.4$ (b) $\tau = 0.6$

(c) $\tau = 0.8$ (d) $\tau = 1.0$

Figure 7.14: Number of iterations with respect to the wavenumber k and τ, for $\nu = 4$

Figure 7.14 illustrates the same parameters but with $\nu = 4$ smoothing steps instead of 2. While the general trends are the same, the number of iterations globally decreases when doubling the number of smoothing steps. Figure 7.15 portrays the operator complexities of each method.

Setting $\tau = 1$ enables the method to converge with nearly constant iteration counts up to five levels according to Figures 7.13 and 7.14, but the operator complexity shown in Figure 7.15 is too high for a practical implementation yet. Even though we kept cost considerations in mind, and therefore developed an approximation based on sparsity constraints, the question of finding more practical operators for indefinite problems is an important topic for the next research on multigrid for Helmholtz. Nevertheless, Figure 7.13 and Figure 7.14 show promising results of our alternative method in solving the indefinite Helmholtz equation with absorbing boundary conditions in a constant number of iterations independently of the matrix size and k.



(a) $\tau = 0.4$ (b) $\tau = 0.6$

(c) $\tau = 0.8$ (d) $\tau = 1.0$

Figure 7.15: Operator complexity with respect to the wavenumber k and τ

Chapter 8

Conclusion and Perspectives

The CEA studies the electromagnetic behavior of three-dimensional complex objects. In this context, numerical linear algebra comes into play when solving the discretized Maxwell's equations. Iterative algorithms are often preferred as they perform better on modern supercomputers. To this day, the linear system of equations is solved by way of a Domain Decomposition Method. At each iteration, independent sub-problems are solved and transmitted to the approximation of the global solution. This approach is robust and allows a better understanding of wave propagation phenomena. However, computing the solution to the local problems requires a direct solver that lacks scalability. To anticipate the release of the next exascale machines, the CEA started investigating alternative methods that scale better than the current domain decomposition method, and multigrid methods are among the most promising ones in that perspective.

However, they strongly rely on theoretical assumptions that do not hold when applied to the Maxwell's equations. More precisely, the discretization matrix is indefinite and has a large and oscillatory near-kernel space. To design a multigrid method for wave propagation problems, the smoother should be adapted to damp potentially large negative eigenvalues and the interpolation designed for approximating the span of the oscillatory near-kernel space correctly. Another important issue is that the keystone coarse correction loses its minimization properties because the matrix does not generate a norm in the indefinite case. With the final aim of solving Maxwell's equation, the CEA opened this thesis to investigate an algebraic multigrid method for solving the indefinite and oscillatory Helmholtz equation.

In Chapter 2, we recalled a few multigrid fundamentals to help the discussion throughout this manuscript. Then, we summarized the state of the art of multigrid for Helmholtz in Chapter 3 and introduced the concept of "pollution" to highlight the relation between the interpolation error and the coarse correction. In particular, we demonstrated that the traditional coarse correction can easily amplify the error associated with small eigenvectors, although the range of interpolation approximates the near-kernel space properly. This traditional coarse correction appears hopeless in the indefinite case. In the next three chapters, we tackled the three main components of multigrid methods that are : the smoother, the interpolation operator, and the coarse correction.

In Chapter 4, we presented a Chebyshev polynomial smoother built on normal equations that damps the large eigenvalues magnitude-wise, regardless of their signs. In the setup phase of our smoother, it is necessary to define an appropriate interval for the selection of the Chebyshev nodes. Hence, we introduced an algorithm based on spectral density approximation techniques to determine the interval algebraically, i.e., without prior geometrical or spectral knowledge on the problem. Then, Chapter 5 addresses the question of interpolation, where we introduced an approximation of the ideal interpolation operator based on least-squares variable operators. The underlying motivation of this new approach is to make the process of ideal approximation more predictable than its classical counterpart, because the least-squares variable operators are better initial approximations of the theoretical complementarity required. Moreover, the column restriction approach with normal equations provides minimization properties to the ideal approximation operator. Then, we opened Chapter 6 by illustrating a case where the traditional coarse correction amplifies the error, even if the interpolation operator approximates the near-kernel space correctly. To remedy the divergence scenarios, we finally introduced an alternative coarse correction based on Euclidean norm minimization. This alternative coarse correction now operates as a contraction of this error, contrary to the traditional one that may amplify it.

The last Chapter 7 benchmarks our two-level method on two different shifted Laplacian discretization matrices and various shifts. In addition, we presented a multilevel extension that we apply on the two-dimensional Helmholtz equation with absorbing boundary conditions. The two-dimensional shifted Laplacian experiments show that our two-level alternative method converges for most shifts where the traditional method diverges. However, the alternative setting remains slow for intermediate shifts that yield extremely indefinite matrices. These matrices are not only characterized by an even proportion of negative and positive eigenvalues, but also by a larger distribution of near-zero eigenvalues. Although our alternative two-level method fixes divergence of traditional methods, its convergence remains too slow in these extreme cases because the pollution still impacts the contraction of the near-zero eigenvalues dramatically. Accelerating the convergence for highly indefinite matrices is another important question that will allow us to add more levels in the multilevel extension as well. At this stage, we see two ways for accelerating the convergence of the method. The first consists of improving interpolation in order to reduce the pollution, while maintaining good sparsity to make the method practical. The second is to better filter the large eigenvectors that decrease the prevalence of the smallest eigenvectors in the minimization space of the alternative coarse correction. While our polynomial smoother already damps the large eigenvalues appropriately without hitting the smallest ones, better filtering approaches may exist as well. That said, we can always find even more demanding problems by playing with the shift. The pollution will always have a strong impact on near-zero eigenvalues. The Helmholtz problem with absorbing boundary conditions has closer connections with real-world applications, and has also been considered in our works. In particular, our multilevel extension solves the two-dimensional Helmholtz equation with absorbing boundary conditions in a constant number of iterations

independently of the matrix size, and with a number of levels that goes up to 6. Moreover, no prior knowledge of the problem is required by our method. While making the method more practical is a topic of future research, our numerical results in terms of convergence are promising for solving indefinite problems with multigrid. The setting of these successful multilevel results requires to select all the columns of the least-squares fine variable operator associated with the non-zero entries of (5.37) in the ideal approximation phase. This requirement is due to the fill-in of coarse matrices that grows with the level index. Naturally, decreasing the number of selected columns of the fine variable operator impacts the convergence more dramatically on coarse levels that have more non-zero entries.

Improving the sparsity of the coarse matrices is a topic of future research that will make the method more practical. One idea is to consider non-Galerkin matrices. Even in the SPD case, traditional multigrid methods may diverge if these non-Galerkin approximations are not spectrally equivalent to their Galerkin counterparts. However, in our setting, they would only help produce the coarse correction vectors of the minimization space. Subsequently, the divergence issue is already fixed by our framework since the alternative coarse correction contracts the error in Euclidean norm in all cases. Nevertheless, the impact on the convergence rate remains an open question. The design of our multigrid operators implicitly relies on normal equations (e.g., polynomial smoother on normal equations, ideal approximation with normal equations, coarse correction through Euclidean minimization). Therefore, developing an algebraic multigrid methods for normal equations matrices is a completely different research direction that we also considered throughout this thesis. In particular, the normal equations matrix of the 5-point stencil problem (1.2) decouples into two separate problems for $\alpha = 4$. This feature can benefit reduction-based algorithms. While this direction of research may give promising results in the future, one remaining difficulty is that squaring the matrix brings the small eigenvalues even closer to zero, which makes them even more sensitive to the pollution. Lastly, the research on multigrid for definite Maxwell problems is also progressing. One promising research direction is to consider block smoothers to treat the local near-kernel components [18]. In the future, it may be interesting to merge these new ideas with our work to finally design an algebraic multigrid method for indefinite Maxwell's equations.

Appendix A

Appendix

A.1 Cauchy's bound Theorem and Chebyshev roots as interpolation points

The following theorem, generally attributed to Cauchy, states that the error of the polynomial of interpolation is bounded by a function of interpolation points.

Theorem 7 (Cauchy's bound theorem)**.** *Let f be a $d+1$ times differentiable function, and p_d be a polynomial of degree d constructed from $d+1$ interpolation points x_i, as defined in* (2.24)*. The error between f and its polynomial interpolation p_d is bounded as follows*

$$|f(x) - p_d(x)| \leq \frac{M_{d+1}}{(d+1)!}\|w_{d+1}\|_\infty, \tag{A.1}$$

where

$$M_{d+1} := \sup_{a \leq x \leq b} \left\{ \left| f^{(d+1)}(x) \right| \right\} \text{ and } w_{d+1}(x) := \prod_{i=1}^{d+1}(x - x_i). \tag{A.2}$$

In our context, the f function is x^{-1}. What Cauchy's bound of Theorem 7 reveals is that the error of interpolation relies on the unitary polynomial w_{d+1} with roots x_i. Thus, the aim of finding a relevant set of $d+1$ interpolation points x_i is to minimize such a function. As mentioned in Section 2.1.2.1 of Chapter 2, the roots of the first kind Chebyshev polynomial (2.27) constitutes the best set of interpolation points. As recalled in [66], the fact that T_{d+1} in (2.27) is a polynomial can be induced from the trigonometric relation

$$\cos(\theta)\cos(d\theta) = \frac{1}{2}\left[\cos(\theta - d\theta) + \cos(\theta + d\theta)\right] \tag{A.3}$$

which is equivalent to

$$\cos((d+1)\theta) = 2\cos(\theta)\cos(d\theta) - \cos((d-1)\theta). \tag{A.4}$$

Hence, equations (2.27) and (A.4) lead to the three-term recurrence relation (2.28). From the three-term recurrence relation of (2.28), the leading coefficient of T_{d+1} is 2^d. As a consequence, the polynomial can be rewritten

$$T_{d+1}(t) = 2^d \prod_{i=1}^{d+1}(t - t_i). \tag{A.5}$$

Moreover, one can demonstrate that the superior bound of the unitary polynomial $2^{-d}T_{d+1}$ is the smallest among all the unitary polynomials of degree $d+1$. In particular, one can show that

$$\frac{1}{2^d}\left\|T_{d+1}\right\|_\infty = \frac{1}{2^d} \leq \left\|w_{d+1}\right\|_\infty, \tag{A.6}$$

where w_{d+1} can be any unitary polynomial as defined in (A.2) of Theorem 7. Then, designing the polynomial p_d by selecting the interpolation points x_i as the roots of a Chebyshev polynomial (i.e., $w_{d+1} = \frac{1}{2^d}T_{d+1}$) minimizes the Cauchy's bound (A.1).

A.2 Further developments on GMRES

For ease of discussion in what follows, let $\boldsymbol{q}_i$ be the i^{th} column of Q_d, and $h_{j,i}$ denote the entry (j,i) of $\bar{H}_d$. The orthonormalized set of Krylov vectors denoted Q_d can be written

$$Q_d = [\boldsymbol{q}_1, \ldots, \boldsymbol{q}_d] \quad \text{where} \quad q_i \perp q_{i+1} \quad i = 1, \ldots, d-1. \tag{A.7}$$

Moreover, the Hessenberg matrix of the Arnoldi relation (2.38) has the form

$$\bar{H}_d = \begin{pmatrix} h_{1,1} & \ldots & h_{1,i} & \ldots & h_{1,d} \\ h_{2,1} & \ddots & & & \vdots \\ 0 & \ddots & h_{i,i} & & h_{i,d} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & h_{d,d} \\ 0 & \ldots & \ldots & 0 & h_{d+1,d} \end{pmatrix} \tag{A.8}$$

Algorithm 7, called Arnoldi procedure, summarizes the Krylov basis construction process.

Algorithm 7 Arnoldi procedure

$\boldsymbol{q}_1 \leftarrow \boldsymbol{b}/\|\boldsymbol{b}\|_2$ ▷ The first Krylov vector is the normalized right-hand side $\boldsymbol{b}$
for $i = 1, d$ **do**
  $\boldsymbol{w} \leftarrow A\boldsymbol{q}_i$ ▷ Each new Krylov vector first results from matrix vector products
  for $j = 1, i$ **do**
    $h_{j,i} \leftarrow \langle \boldsymbol{q}_j, \boldsymbol{w} \rangle$ ▷ The Gram-Schmidt coefficients are stored in H_d
    $\boldsymbol{w} \leftarrow \boldsymbol{w} - h_{j,i}\boldsymbol{q}_j$ ▷ The new vector is orthonormalized against all the others
  end for
  if $\|\boldsymbol{w}\|_2 > 0$ **do** ▷ i.e., if $A\boldsymbol{q}_i \notin \text{vect}\{\boldsymbol{q}_1, \ldots, \boldsymbol{q}_i\}$
    $h_{i+1,i} \leftarrow \|\boldsymbol{w}\|_2$
    $\boldsymbol{q}_{i+1} \leftarrow \boldsymbol{w}/h_{i+1,i}$ ▷ The basis is augmented with the orthonormalized vector $A\boldsymbol{q}_i$
  end if
end for

Let $\bar{G}_i$ be the ith Given's rotation matrix such that

$$\bar{G}_i := \begin{pmatrix} 1 & 0 & \dots & \dots & \dots & 0 \\ 0 & \ddots & & & & \vdots \\ \vdots & & c_i & s_i & & \vdots \\ \vdots & & -s_i & c_i & & \vdots \\ \vdots & & & & \ddots & 0 \\ 0 & \dots & \dots & \dots & 0 & 1 \end{pmatrix} \quad , \quad c_i := \frac{h_{i,i}}{\sqrt{h_{i,i}^2 + h_{i+1,i}^2}} \text{ and } s_i := \frac{h_{i+1,i}}{\sqrt{h_{i,i}^2 + h_{i+1,i}^2}}. \tag{A.9}$$

Also, let $G_d := \bar{G}_d \dots \bar{G}_1$ be the unitary matrix resulting from the product of the d Given's rotations. Multiplying $\bar{H}_d$ with G_d returns the $(d+1 \times d)$ upper triangular matrix

$$\bar{U}_d := G_d \bar{H}_d = \begin{pmatrix} u_{1,1} & \dots & u_{1,i} & \dots & u_{1,d} \\ 0 & \ddots & & & \vdots \\ \vdots & \ddots & u_{i,i} & & u_{i,d} \\ \vdots & & \ddots & \ddots & \vdots \\ \vdots & & & \ddots & u_{d,d} \\ 0 & \dots & \dots & \dots & 0 \end{pmatrix} \tag{A.10}$$

The next development shows that minimizing the residual amounts to solve a triangular system of linear equations

$$\begin{aligned} \min \|\boldsymbol{r}^{(k+1)}\|_2 &= \min_{\tilde{\boldsymbol{\rho}}_d \in \mathbb{C}^d} \|\boldsymbol{r}^{(k)} - AQ_d \tilde{\boldsymbol{\rho}}_d\|_2 = \min_{\tilde{\boldsymbol{x}} \in \mathcal{K}_d} \|\boldsymbol{r}^{(k)} - A\tilde{\boldsymbol{x}}\|_2 \\ &= \min_{\tilde{\boldsymbol{\rho}}_d \in \mathbb{C}^d} \|\boldsymbol{r}^{(k)} - Q_{d+1} \bar{H}_d \boldsymbol{\rho}_d\|_2 = \min_{\tilde{\boldsymbol{\rho}}_d \in \mathbb{C}^d} \|Q_{d+1}(\beta \overrightarrow{\boldsymbol{e}}_1 - \bar{H}_d \boldsymbol{\rho}_d)\|_2 \end{aligned} \tag{A.11}$$

Define $\beta := \|\boldsymbol{r}^{(k)}\|_2$ with $\boldsymbol{q}_1 = \frac{\boldsymbol{r}^{(k)}}{\beta}$, and $\boldsymbol{g}_{d+1} := \beta G_d \overrightarrow{\boldsymbol{e}}_1$. Since both Q_{d+1} and G_d are unitary

$$\min \|\boldsymbol{r}^{(k+1)}\|_2 = \min_{\tilde{\boldsymbol{\rho}}_d \in \mathbb{C}^d} \|Q_{d+1}(\beta \overrightarrow{\boldsymbol{e}}_1 - \bar{H}_d \boldsymbol{\rho}_d)\|_2 = \min_{\tilde{\boldsymbol{\rho}}_d \in \mathbb{C}^d} \|\beta \overrightarrow{\boldsymbol{e}}_1 - \bar{H}_d \boldsymbol{\rho}_d\|_2 \tag{A.12}$$

$$= \min_{\tilde{\boldsymbol{\rho}}_d \in \mathbb{C}^d} \|\beta G_d \overrightarrow{\boldsymbol{e}}_1 - G_d \bar{H}_d \boldsymbol{\rho}_d\|_2 = \min_{\tilde{\boldsymbol{\rho}}_d \in \mathbb{C}^d} \|\boldsymbol{g}_{d+1} - \bar{U}_d \boldsymbol{\rho}_d\|_2 \tag{A.13}$$

Denote by U_d and $\boldsymbol{g}_d$ the $(m \times m)$ upper triangular matrix and the vector of size d obtained from $\bar{U}_d$ and $\boldsymbol{g}_{d+1}$ by deleting their last row and component g_{d+1} respectively. It follows

$$\min \|\boldsymbol{r}^{(k+1)}\|_2^2 = \min_{\tilde{\boldsymbol{\rho}}_d \in \mathbb{C}^d} \|\boldsymbol{g}_{d+1} - \bar{U}_d \tilde{\boldsymbol{\rho}}_d\|_2^2 = \min_{\tilde{\boldsymbol{\rho}}_d \in \mathbb{C}^d} |g_{d+1}|^2 + \|\boldsymbol{g}_d - U_d \tilde{\boldsymbol{\rho}}_d\|_2^2. \tag{A.14}$$

It follows that the residual is minimized for

$$\boldsymbol{\rho}_d = U_d^{-1} \boldsymbol{g}_d \;\Leftrightarrow\; \min \|\boldsymbol{r}^{(k+1)}\|_2^2 = |g_{d+1}|^2 . \tag{A.15}$$

Because the solution to the minimization problem lies in $\mathcal{K}_d$, the approximation is updated as follows

$$\boldsymbol{x}^{(i+1)} = \boldsymbol{x}^{(i)} + Q_d \boldsymbol{\rho}_d. \tag{A.16}$$

A.3 Idealistic example of ideal interpolation

Define coarse and fine spaces respectively in directions of small and large eigenvectors denoted by the sets V_c and V_f respectively such that

$$R^T := V_c \ , \ S := V_f. \tag{A.17}$$

Since eigenvectors are orthonormal, the necessary condition $RS = 0$ is satisfied. Injecting (A.17) in the definition of ideal interpolation (2.69) gives

$$P_* = R^T = V_c \text{ and } P_*^T A P_* = Diag(\lambda_1, \dots, \lambda_{n_c}). \tag{A.18}$$

This idealistic dichotomy enabled by P_* maximizes the complementarity principle. The near-kernel space contained in V_c is solved directly at the coarsest level while the large eigenvectors of V_f remains in the smoothing space.

A.4 Additional developments on the condition number of CSL preconditioned matrices

As shown in [31], the numerator of (3.59) can be approximated by

$$\lambda_{\max}^h \left(\left(H_{\gamma,\beta}^{-1} A_h \right)^* \left(H_{\gamma,\beta}^{-1} A_h \right) \right) \approx \max \left\{ 1, \frac{1}{\gamma^2 + \beta^2} \right\}. \tag{A.19}$$

Secondly, letting $\epsilon := \min_j \left(s_j^2 - \left(\frac{kh}{2} \right)^2 \right)$, the denominator is

$$\lambda_{\min}^h \left(\left(H_{\gamma,\beta}^{-1} A_h \right)^* \left(H_{\gamma,\beta}^{-1} A_h \right) \right) = \frac{\epsilon^2}{\left(\epsilon + (1+\gamma) \left(\frac{kh}{2} \right)^2 \right)^2 + \beta^2 \left(\frac{kh}{2} \right)^4}. \tag{A.20}$$

Assuming the smallest eigenvalue of the Helmholtz matrix is very small (i.e., $\epsilon \left(\frac{kh}{2} \right)^2 \ll \left(\frac{kh}{2} \right)^4$ and $\epsilon^2 \approx 0$), the minimal eigenvalue can be approximated by

$$\lambda_{\min}^h \left(\left(H_{\gamma,\beta}^{-1} A_h \right)^* \left(H_{\gamma,\beta}^{-1} A_h \right) \right) \approx \frac{4}{(1+\gamma)^2 + \beta^2} \left(\frac{2\epsilon}{(kh)^2} \right)^2. \tag{A.21}$$

As written in (3.60), the condition number of the squared left preconditioned matrix (3.59) is given by

$$\kappa^2 = \begin{cases} \frac{1}{4} \left(1 + \frac{2\gamma}{\gamma^2+\beta^2} \right) \left((kh)^2 / (2\epsilon) \right)^2 & \text{if } \gamma^2 + \beta^2 \leq 1, \\ \frac{1}{4} \left((1+\gamma)^2 + \beta^2 \right) \left((kh)^2 / (2\epsilon) \right)^2, & \text{if } \gamma^2 + \beta^2 \geq 1 \end{cases}. \tag{A.22}$$

Let us compare the action of each preconditioner in the particular case where $(0 < k^2 < \mu_1)$. The minimal and the maximal eigenvalues of each of the four matrices are respectively reached for indexes $j = 1$ and $j = n$. To compare the condition numbers of the plain and real shifted Laplacian preconditioned matrices, the ratio between both minimal eigenvalues is given by

$$\frac{\lambda_{\min}\left(\left(H_0^{-1}A\right)^*\left(H_0^{-1}A\right)\right)}{\lambda_{\min}\left(\left(H_1^{-1}A\right)^*\left(H_1^{-1}A\right)\right)} = \frac{\left(\mu_1+k^2\right)^2}{\mu_1^2} > 1. \tag{A.23}$$

Regarding maximal eigenvalues, one finds that

$$\lim_{\mu_n\to\infty}\lambda_{\max}\left(\left(H_0^{-1}A\right)^*\left(H_0^{-1}A\right)\right) = \lim_{\mu_n\to\infty}\lambda_{\max}\left(\left(H_1^{-1}A\right)^*\left(H_1^{-1}A\right)\right) = 1. \tag{A.24}$$

Consequently, the Laplacian preconditioner with no shift offers a better condition number than the real shifted preconditioner when the wavenumber is smaller than the minimal eigenvalues. In the same way, comparing minimal eigenvalues of the plain Laplacian preconditioner with the complex shifted preconditioner gives

$$\frac{\lambda_{\min}\left(\left(H_0^{-1}A\right)^*\left(H_0^{-1}A\right)\right)}{\lambda_{\min}\left(\left(H_\iota^{-1}A\right)^*\left(H_\iota^{-1}A\right)\right)} = \frac{\left(\mu_1+k^2\right)^2}{\mu_1^2+k^4} > 1. \tag{A.25}$$

As in (A.24), we have

$$\lim_{\mu_n\to\infty}\lambda_{\min}\left(\left(H_0^{-1}A\right)^*\left(H_0^{-1}A\right)\right) = \lim_{\mu_n\to\infty}\lambda_{\min}\left(\left(H_\iota^{-1}A\right)^*\left(H_\iota^{-1}A\right)\right) = 1. \tag{A.26}$$

Here again, the plain Laplacian preconditioner provides a smaller condition number than the complex shifted preconditioner in that first case where $0 < k^2 < \mu_1$.

In the second scenario where the wavenumber is greater than the minimal eigenvalue of the Laplacian (i.e., $\mu_1 < k^2 < \mu_n$), the maximal eigenvalue of the Laplacian preconditioner with no shift becomes

$$\lambda_{\max}\left(\left(H_0^{-1}A\right)^*\left(H_0^{-1}A\right)\right) = \max\left\{\left(\frac{\mu_n-k^2}{\mu_n}\right)^2, \left(\frac{\mu_1-k^2}{\mu_1}\right)^2\right\}. \tag{A.27}$$

Equation (A.27) shows that the maximal eigenvalue increases dramatically as μ_1 gets closer to zero. This feature is illustrated by the extreme left mark in Figure 3.6a. Let us now compare it with both real and complex shifted Laplacian preconditioned matrices. The maximal eigenvalue of the former is

$$\lambda_{\max}\left(\left(H_1^{-1}A\right)^*\left(H_1^{-1}A\right)\right) = \max\left\{\left(\frac{\mu_n-k^2}{\mu_n+k^2}\right)^2, \left(\frac{\mu_1-k^2}{\mu_1+k^2}\right)^2\right\}. \tag{A.28}$$

whereas the maximal eigenvalue of the latter is

$$\lambda_{\max}\left(\left(H_\iota^{-1}A\right)^*\left(H_\iota^{-1}A\right)\right) = \max\left\{\frac{\left(\mu_n-k^2\right)^2}{\mu_n^2+k^4}, \frac{\left(\mu_1-k^2\right)^2}{\mu_1^2+k^4}\right\}. \tag{A.29}$$

One can demonstrate that their spectra are upper bounded such that

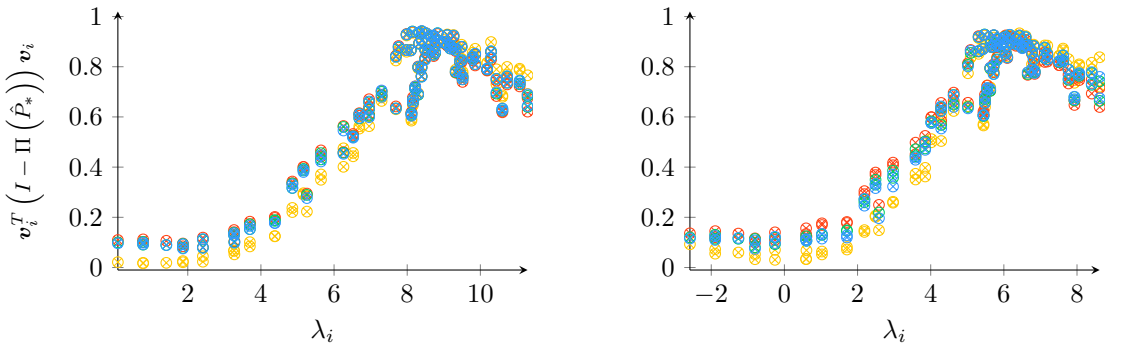
$$\begin{aligned}\lim_{\mu_n\to\infty}\lambda_{\max}\left(\left(H_1^{-1}A\right)^*\left(H_1^{-1}A\right)\right) &= \lim_{\mu_1\to 0}\lambda_{\max}\left(\left(H_1^{-1}A\right)^*\left(H_1^{-1}A\right)\right)\\ &= \lim_{\mu_1\to 0}\lambda_{\max}\left(\left(H_\iota^{-1}A\right)^*\left(H_\iota^{-1}A\right)\right)\\ &= \lim_{\mu_n\to\infty}\lambda_{\max}\left(\left(H_1^{-1}A\right)^*\left(H_1^{-1}A\right)\right) = 1.\end{aligned} \tag{A.30}$$

For a minimal eigenvalue of the same magnitude, shifted preconditioners provide a better condition number in the case where at least one eigenvalue of the initial Helmholtz problem is negative. Assuming the smallest eigenvalue of index d is near zero, then the associated Laplacian eigenvalue is $\mu_m \approx k^2 + \epsilon$. It naturally follows

$$\frac{\lambda_{\min}\left(\left(H_\iota^{-1}A\right)^*\left(H_\iota^{-1}A\right)\right)}{\lambda_{\min}\left(\left(H_1^{-1}A\right)^*\left(H_1^{-1}A\right)\right)} = \frac{\left(\mu_m + k^2\right)^2}{\mu_m^2 + k^4} = \frac{4k^4 + 4k^2\epsilon + \epsilon^2}{2k^4 + 2k^2\epsilon + \epsilon^2} \approx 2. \tag{A.31}$$

In theory, the complex shifted Laplacian offers the best condition number among the three preconditioners. In the first hand because it bounds the maximal eigenvalue, in the other because its minimal eigenvalue is twice larger than its real shifted counterpart.

A.5 Least-squares variable operators with SPAI



(a) Least-squares variable operators R^T and S - $\alpha = 0.625^2$

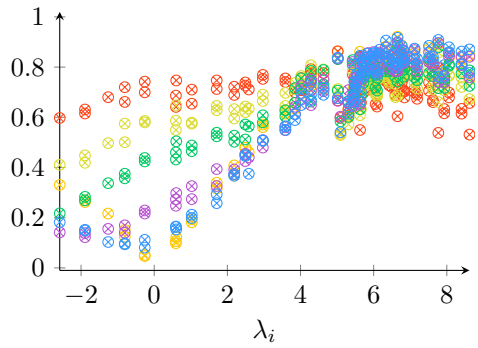
(b) Least-squares variable operators R^T and S - $\alpha = 1.75^2$

Figure A.1: Error of the l_2-projection onto the range of the classical and alternative ideal approximations using the SPAI approach for the model problem SL2D-9S with respect to m

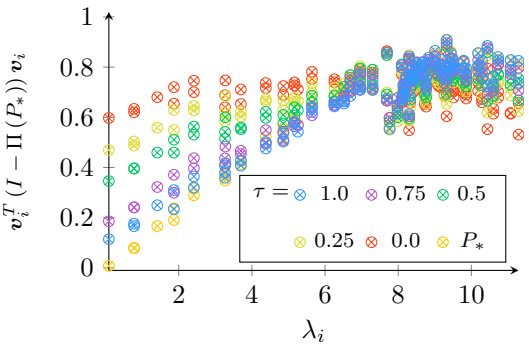
A.6 Classical variable operators with the subspace restriction approach

A.7 Alteration of the coarse correction with classical variable operators and SPAI

The Figure A.3 plots the same experiments as in Figure 6.1 but for a 9-point discretization stencil of the two-dimensional shifted Laplacian matrix. In this experiment, we use the classical variable operators R^T and S. Top figures correspond to $\alpha = 0.625^2$, whereas the shift associated with bottom figures is $\alpha = 2.1^2$. Both left figures A.3a and A.3b use the approximation P of the classical ideal interpolation P_* given by the SPAI approach and with normal equations, as described in Section 5.2.2. Conversely, the right figures correspond to the classical ideal interpolation operator P_*. Here again, blue and green curves fit well, but the coarse correction

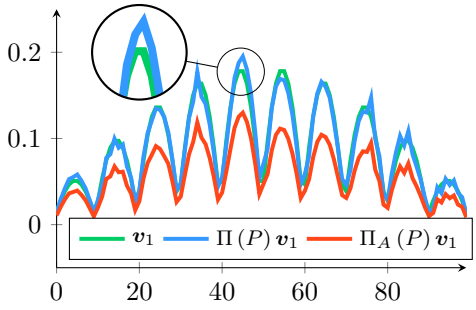


(a) Classical variable operators R^T and S - $\alpha = 0.625^2$

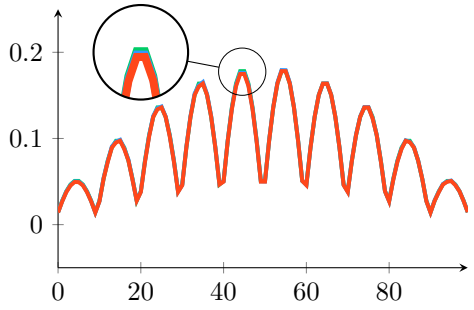


(b) Classical variable operators R^T and S - $\alpha = 1.75^2$

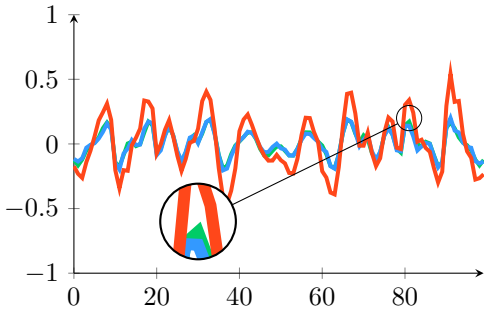
Figure A.2: Error of the l_2-projection onto the range of the classical ideal approximations with the subspace restriction approach for the model problem SL2D-9S with respect to τ



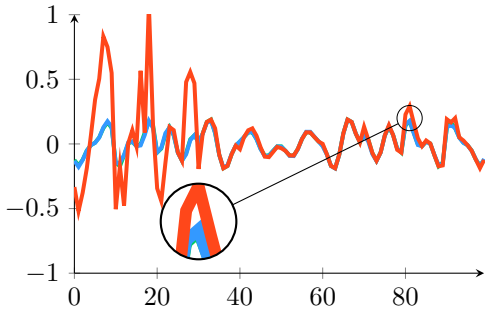
(a) 9-pts, $\alpha = 0.625^2$ - P , $m = 2$



(b) 9-pts, $\alpha = 0.625^2$ - P_*



(c) 9-pts, $\alpha = 2.1^2$ - P , $m = 2$



(d) 9-pts, $\alpha = 2.1^2$ - P_*

Figure A.3: Smallest eigenvector $\boldsymbol{v}_1$ vs. its l_2-projection $\Pi(P)\boldsymbol{v}_1$ vs. its coarse correction $\Pi_A(P)\boldsymbol{v}_1$, for two different shifts

vector is amplified for $\alpha = 2.1^2$. Moreover, the previous observation in the comparison between the ideal interpolation and its approximation holds in this case as well. Decreasing the interpolation error for $\alpha = 0.625^2$ improves the coarse correction vector, whereas it induces a stronger amplification when $\alpha = 2.1^2$.

Bibliography

[1] Mark Adams, Marian Brezina, Jonathan Hu, and Ray Tuminaro. Parallel multigrid smoothing: polynomial versus Gauss–Seidel. *Journal of Computational Physics*, 188(2):593–610, July 2003.

[2] Tuomas Airaksinen, Erkki Heikkola, Anssi Pennanen, and Jari Toivanen. An algebraic multigrid based shifted-Laplacian preconditioner for the Helmholtz equation. *Journal of Computational Physics*, 226(1):1196–1210, September 2007.

[3] Allison H. Baker, Robert D. Falgout, Tzanio V. Kolev, and Ulrike Meier Yang. Multigrid Smoothers for Ultraparallel Computing. *SIAM Journal on Scientific Computing*, 33(5):2864–2887, January 2011.

[4] N.S. Bakhvalov. On the convergence of a relaxation method with natural constraints on the elliptic operator. *USSR Computational Mathematics and Mathematical Physics*, 6(5):101–135, January 1966.

[5] Alvin Bayliss, Charles I Goldstein, and Eli Turkel. An iterative method for the Helmholtz equation. *Journal of Computational Physics*, 49(3):443–457, March 1983.

[6] Matthias Bollhöfer, Marcus J. Grote, and Olaf Schenk. Algebraic Multilevel Preconditioner for the Helmholtz Equation in Heterogeneous Media. *SIAM Journal on Scientific Computing*, 31(5):3781–3805, January 2009.

[7] A. Brandt, J. Brannick, K. Kahl, and I. Livshits. Bootstrap AMG. *SIAM Journal on Scientific Computing*, 33(2):612–632, January 2011.

[8] A. Brandt and I. Livshits. Wave-ray multigrid method for standing wave equations. *ETNA. Electronic Transactions on Numerical Analysis [electronic only]*, 6:162–181, 1997.

[9] A. Brandt and S. Taasan. Multigrid method for nearly singular and slightly indefinite problems. November 1985.

[10] James Brannick, Fei Cao, Karsten Kahl, Robert D. Falgout, and Xiaozhe Hu. Optimal Interpolation and Compatible Relaxation in Classical Algebraic Multigrid. *SIAM Journal on Scientific Computing*, 40(3):A1473–A1493, January 2018.

[11] M. Brezina, A. J. Cleary, R. D. Falgout, V. E. Henson, J. E. Jones, T. A. Manteuffel, S. F. McCormick, and J. W. Ruge. Algebraic Multigrid Based

on Element Interpolation (AMGe). *SIAM Journal on Scientific Computing*, 22(5):1570–1592, January 2001.

[12] M. Brezina, R. Falgout, S. MacLachlan, T. Manteuffel, S. McCormick, and J. Ruge. Adaptive Smoothed Aggregation (αSA). *SIAM Journal on Scientific Computing*, 25(6):1896–1920, January 2004.

[13] William L. Briggs, Van Emden Henson, and Steve F. McCormick. *A multigrid tutorial (2nd ed.)*. Society for Industrial and Applied Mathematics, USA, October 2000.

[14] Stéphanie Chaillat, Luca Desiderio, and Patrick Ciarlet. Theory and implementation of H-matrix based iterative and direct solvers for Helmholtz and elastodynamic oscillatory kernels. *Journal of Computational Physics*, 351:165–186, December 2017.

[15] T. Chartier, R. D. Falgout, V. E. Henson, J. Jones, T. Manteuffel, S. McCormick, J. Ruge, and P. S. Vassilevski. Spectral AMGe (ρAMGe). *SIAM Journal on Scientific Computing*, 25(1):1–26, January 2003.

[16] Edmond Chow. An unstructured multigrid method based on geometric smoothness. *Numerical Linear Algebra with Applications*, 10(5-6):401–421, July 2003.

[17] Edmond Chow, Robert D. Falgout, Jonathan J. Hu, Raymond S. Tuminaro, and Ulrike Meier Yang. 10. A Survey of Parallelization Techniques for Multigrid Solvers. In Michael A. Heroux, Padma Raghavan, and Horst D. Simon, editors, *Parallel Processing for Scientific Computing*, pages 179–201. Society for Industrial and Applied Mathematics, January 2006.

[18] L. Claus, R. D. Falgout, and M. Bolten. AMG Smoothers for Maxwell's Equations. Technical Report LLNL-CONF-744582, Lawrence Livermore National Lab. (LLNL), Livermore, CA (United States), January 2018.

[19] Lisa Claus. Multigrid smoothers for saddle point systems. 2019.

[20] Lisa Claus and Matthias Bolten. Non-overlapping block smoothers for the Stokes equations, August 2020. arXiv:2008.08719 [cs, math].

[21] Pierre-Henri Cocquet and Martin J. Gander. How Large a Shift is Needed in the Shifted Helmholtz Preconditioner for its Effective Inversion by Multigrid? *SIAM Journal on Scientific Computing*, 39(2):A438–A478, January 2017.

[22] Olivier Coulaud, Luc Giraud, Pierre Ramet, and Xavier Vasseur. Deflation and augmentation techniques in Krylov subspace methods for the solution of linear systems. 2013.

[23] Timothy A. Davis. *Direct Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics, January 2006.

[24] Hans De Sterck, Ulrike Meier Yang, and Jeffrey J. Heys. Reducing Complexity in Parallel Algebraic Multigrid Preconditioners. *SIAM Journal on Matrix Analysis and Applications*, 27(4):1019–1039, January 2006.

[25] James W. Demmel. *Applied Numerical Linear Algebra*. Society for Industrial and Applied Mathematics, January 1997.

[26] Vandana Dwarka and Cornelis Vuik. Scalable Convergence Using Two-Level Deflation Preconditioning for the Helmholtz Equation. *SIAM Journal on Scientific Computing*, 42(2):A901–A928, January 2020.

[27] Vandana Dwarka and Cornelis Vuik. Scalable multi-level deflation preconditioning for highly indefinite time-harmonic waves. *Journal of Computational Physics*, 469:111327, November 2022.

[28] Vandana Dwarka and Cornelis Vuik. Stand-alone Multigrid for Helmholtz Revisited: Towards Convergence Using Standard Components, August 2023. arXiv:2308.13476 [cs, math].

[29] Howard C. Elman, Oliver G. Ernst, and Dianne P. O'Leary. A Multigrid Method Enhanced by Krylov Subspace Iteration for Discrete Helmholtz Equations. *SIAM Journal on Scientific Computing*, 23(4):1291–1315, January 2001.

[30] Y. A. Erlangga, C. W. Oosterlee, and C. Vuik. A Novel Multigrid Based Preconditioner For Heterogeneous Helmholtz Problems. *SIAM Journal on Scientific Computing*, 27(4):1471–1492, January 2006.

[31] Y.A Erlangga, C Vuik, and C.W Oosterlee. On a class of preconditioners for solving the Helmholtz equation. *Applied Numerical Mathematics*, 50(3-4):409–425, September 2004.

[32] Y.A. Erlangga, C. Vuik, and C.W. Oosterlee. Comparison of multigrid and incomplete LU shifted-Laplace preconditioners for the inhomogeneous Helmholtz equation. *Applied Numerical Mathematics*, 56(5):648–666, May 2006.

[33] O. G. Ernst and M. J. Gander. Why it is Difficult to Solve Helmholtz Problems with Classical Iterative Methods. In Ivan G. Graham, Thomas Y. Hou, Omar Lakkis, and Robert Scheichl, editors, *Numerical Analysis of Multiscale Problems*, volume 83, pages 325–363. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[34] Oliver G. Ernst and Martin J. Gander. Multigrid methods for Helmholtz problems: A convergent scheme in 1D using standard components. In Ivan Graham, Ulrich Langer, Jens Melenk, and Mourad Sini, editors, *Direct and Inverse Problems in Wave Propagation and Applications*, pages 135–186. DE GRUYTER, September 2013.

[35] R. D. Falgout. An Introduction to Algebraic Multigrid. *Computing in Science and Engineering, vol. 8, no. 6, November 1, 2006, pp. 24-33*, April 2006.

[36] R. D. Falgout, S. Friedhoff, Tz. V. Kolev, S. P. MacLachlan, and J. B. Schroder. Parallel Time Integration with Multigrid. *SIAM Journal on Scientific Computing*, 36(6):C635–C661, January 2014.

[37] Robert D. Falgout and Jacob B. Schroder. Non-galerkin coarse grids for algebraic multigrid. *SIAM Journal on Scientific Computing*, 36(3):C309–C334, 2014.

[38] Robert D. Falgout and Panayot S. Vassilevski. On Generalizing the Algebraic Multigrid Framework. *SIAM Journal on Numerical Analysis*, 42(4):1669–1693, January 2004.

[39] Robert D. Falgout, Panayot S. Vassilevski, and Ludmil T. Zikatanov. On two-grid convergence estimates. *Numerical Linear Algebra with Applications*, 12(5-6):471–494, June 2005.

[40] R.P. Fedorenko. A relaxation method for solving elliptic difference equations. *USSR Computational Mathematics and Mathematical Physics*, 1(4):1092–1096, January 1962.

[41] Paul O Frederickson. *Fast approximate inversion of large sparse linear systems*. Lakehead University, Department of Mathematical Sciences, 1975.

[42] M. J. Gander, I. G. Graham, and E. A. Spence. Applying GMRES to the Helmholtz equation with shifted Laplacian preconditioning: what is the largest shift for which wavenumber-independent convergence is guaranteed? *Numerische Mathematik*, 131(3):567–614, November 2015.

[43] J. Gozani, A. Nachshon, and E. Turkel. Conjugate gradient coupled with multigrid for an indefinite problem. June 1984.

[44] Wolfgang Hackbusch. *Multi-Grid Methods and Applications*, volume 4 of *Springer Series in Computational Mathematics*. Springer Berlin Heidelberg, Berlin, Heidelberg, 1985.

[45] G. Ronald Hadley. A complex Jacobi iterative method for the indefinite Helmholtz equation. *Journal of Computational Physics*, 203(1):358–370, February 2005.

[46] Stuart C. Hawkins and Ke Chen. An Implicit Wavelet Sparse Approximate Inverse Preconditioner. *SIAM Journal on Scientific Computing*, 27(2):667–686, January 2005.

[47] P.W. Hemker. A note on defect correction processes with an approximate inverse of deficient rank. *Journal of Computational and Applied Mathematics*, 8(2):137–139, June 1982.

[48] Van Emden Henson and Panayot S. Vassilevski. Element-Free AMGe: General Algorithms for Computing Interpolation Weights in AMG. *SIAM Journal on Scientific Computing*, 23(2):629–650, January 2001.

[49] Van Emden Henson and Ulrike Meier Yang. BoomerAMG: A parallel algebraic multigrid solver and preconditioner. *Applied Numerical Mathematics*, 41(1):155–177, April 2002.

[50] Roger A. Horn and Charles R. Johnson. *Topics in Matrix Analysis*. Cambridge University Press, 1 edition, April 1991.

[51] Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 2 edition, October 2012.

[52] P. Hénon, P. Ramet, and J. Roman. PaStiX: a high-performance parallel direct solver for sparse symmetric positive definite systems. *Parallel Computing*, 28(2):301–321, February 2002.

[53] Johannes K. Kraus, Panayot S. Vassilevski, and Ludmil T. Zikatanov. Polynomial of best uniform approximation to x^{-1} and smoothing in two-level methods. 2010.

[54] A. Laird and M. Giles. Preconditioned iterative solution of the 2D Helmholtz equation. 2002.

[55] Matthieu Lecouvez. *Méthodes itératives de décomposition de domaine sans recouvrement avec convergence géométrique pour l'équation de Helmholtz*. These de doctorat, Palaiseau, Ecole polytechnique, January 2015.

[56] B. Lee, T. A. Manteuffel, S. F. McCormick, and J. Ruge. First-Order System Least-Squares for the Helmholtz Equation. *SIAM Journal on Scientific Computing*, 21(5):1927–1949, January 2000.

[57] Lin Lin, Yousef Saad, and Chao Yang. Approximating Spectral Densities of Large Matrices. *SIAM Review*, 58(1):34–65, January 2016.

[58] I. Livshits. Multiple Galerkin Adaptive Algebraic Multigrid Algorithm for the Helmholtz Equations. *SIAM Journal on Scientific Computing*, 37(5):S195–S215, January 2015.

[59] Ira Livshits. A scalable multigrid method for solving indefinite Helmholtz equations with constant wave numbers. *Numerical Linear Algebra with Applications*, 21(2):177–193, March 2014.

[60] Irene Livshits. An algebraic multigrid wave–ray algorithm to solve eigenvalue problems for the helmholtz operator. *Numerical Linear Algebra with Applications*, 11(2-3):229–239, March 2004.

[61] S. MacLachlan and Yousef Saad. A Greedy Strategy for Coarse-Grid Selection. *SIAM Journal on Scientific Computing*, 29(5):1825–1853, January 2007.

[62] Scott MacLachlan, Tom Manteuffel, and Steve McCormick. Adaptive reduction-based AMG. *Numerical Linear Algebra with Applications*, 13(8):599–620, October 2006.

[63] James M. Ortega and Robert J. Plemmons. Extensions of the Ostrowski-Reich theorem for SOR iterations. *Linear Algebra and its Applications*, 28:177–191, December 1979.

[64] Youcef Saad and Martin H. Schultz. GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems. *SIAM Journal on Scientific and Statistical Computing*, 7(3):856–869, July 1986.

[65] Yousef Saad. ILUT: A dual threshold incomplete LU factorization. *Numerical Linear Algebra with Applications*, 1(4):387–402, July 1994.

[66] Yousef Saad. *Iterative Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics, second edition, January 2003.

[67] A. H. Sheikh, D. Lahaye, and C. Vuik. A Scalable Helmholtz Solver combining the Shifted Laplace Preconditioner with Multigrid Deflation. *Reports of the Department of Applied Mathematical Analysis*, January 2011.

[68] A. H. Sheikh, D. Lahaye, and C. Vuik. On the convergence of shifted Laplace preconditioner combined with multilevel deflation. *Numerical Linear Algebra with Applications*, 20(4):645–662, August 2013.

[69] Gilbert Strang. Multigrid methods. Technical report, MIT, 2006.

[70] K. Stüben. A review of algebraic multigrid. *Journal of Computational and Applied Mathematics*, 128(1-2):281–309, March 2001.

[71] Lloyd N. Trefethen and David Bau. *Numerical Linear Algebra, Twenty-fifth Anniversary Edition*. Society for Industrial and Applied Mathematics, Philadelphia, PA, January 2022.

[72] U. Trottenberg, C.W. Oosterlee, and A. Schuller. *Multigrid*. Elsevier Science, 2000.

[73] Petr Vanek, Marian Brezina, and Radek Tezaur. Two-grid Method for Linear Elasticity on Unstructured Meshes. *SIAM Journal on Scientific Computing*, 21(3):900–923, January 1999.

[74] Petr Van\vek, Marian Brezina, and Jan Mandel. Convergence of algebraic multigrid based on smoothed aggregation. *Numerische Mathematik*, 88(3):559–579, May 2001.

[75] P. Vaněk, J. Mandel, and M. Brezina. Algebraic multigrid by smoothed aggregation for second and fourth order elliptic problems. *Computing*, 56(3):179–196, September 1996.

[76] Benson M. W. Iterative solution of large sparse linear systems arising in certain multidimensional approximation problems. *Util. Math.*, 22:127–140, 1982.

[77] Junping Wang. Convergence Analysis Without Regularity Assumptions for Multigrid Algorithms Based on SOR Smoothing. *SIAM Journal on Numerical Analysis*, 29(4):987–1001, 1992.

[78] P. Wesseling and C.W. Oosterlee. Geometric multigrid with applications to computational fluid dynamics. *Journal of Computational and Applied Mathematics*, 128(1-2):311–334, March 2001.

[79] Xuefeng Xu and Chen-Song Zhang. On the Ideal Interpolation Operator in Algebraic Multigrid Methods. *SIAM Journal on Numerical Analysis*, 56(3):1693–1710, January 2018.

[80] U. M. Yang. Parallel Algebraic Multigrid Methods - High Performance Preconditioners. Technical Report UCRL-BOOK-208032, Lawrence Livermore National Lab. (LLNL), Livermore, CA (United States), November 2004.

[81] I. Yavneh. Why Multigrid Methods Are So Efficient. *Computing in Science & Engineering*, 8(6):12–22, November 2006.

[82] David M. Young. On the accelerated SSOR method for solving large linear systems. *Advances in Mathematics*, 23(3):215–271, March 1977.

[83] Tareq Zaman, Nicolas Nytko, Ali Taghibakhshi, Scott MacLachlan, Luke Olson, and Matthew West. Generalizing Reduction-Based Algebraic Multigrid. 2022.

[84] Tareq Zaman, Nicolas Nytko, Ali Taghibakhshi, Scott MacLachlan, Luke Olson, and Matthew West. Generalizing reduction-based algebraic multigrid. *Numerical Linear Algebra with Applications*, 31(3):e2543, May 2024.

[85] Tareq Uz Zaman, Scott P. MacLachlan, Luke N. Olson, and Matthew West. Coarse-grid selection using simulated annealing. *Journal of Computational and Applied Mathematics*, 431:115263, October 2023.