PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA

FAST CALDERÓN PRECONDITIONING FOR HELMHOLTZ BOUNDARY INTEGRAL EQUATIONS

MARÍA IGNACIA FIERRO PICCARDO

Thesis submitted to the Office of Research and Graduate Studies
in partial fulfillment of the requirements for the degree of
Master of Science in Engineering

Advisor:
CARLOS JEREZ HANCKES

Santiago de Chile, April, 2019

PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA

FAST CALDERÓN PRECONDITIONING FOR HELMHOLTZ BOUNDARY INTEGRAL EQUATIONS

MARÍA IGNACIA FIERRO PICCARDO

Members of the Committee:
CARLOS JEREZ HANCKES
ELWIN VAN'T WOUT
MICHAEL KARKULIK
JOSÉ LUIS ALMAZÁN

Thesis submitted to the Office of Research and Graduate Studies
in partial fulfillment of the requirements for the degree of
Master of Science in Engineering

Santiago de Chile, April, 2019

ACKNOWLEDGEMENTS

I would like to thank to all people that helped me completing this thesis. First, I want to thank to the members of the committee for giving me an opportunity and acceding to read this document. I hope you enjoy it, as I enjoyed fabricating it. I must make a special mention to my advisor, Carlos Jerez, for seeing through my insecurities and trust me this project. I must rephrase the words of Consuelo and say that he always believed in me more than I do. I could never stop thanking him for that.

I cannot stop to mention who once were my office partners: Ignacio, who has been a great friend since years, Rubén who I admire for his ability of watching memes and still being a productive human being, Paul for supporting me as a researcher. At last is José, who always saw a shine in me, has a golden disposition to help, and was key to this investigation.

I want to thank my family, specially to my mother. Although it seems that we came from different worlds and although we do not keep a perfect relationship, you always let me fabricate and follow my own path, even if it was not the most classical one. I hope you realize how much I appreciate that you let me be who I want to be. Secondly, I want to thank my sister, Francisca, for becoming a friend this last decade. I would also like to thank to my grandmother, Adela, for being present in the most important moments of my life. I also thank to my aunts Marcela and Alejandra for always being caring, even if I am not present, and recognizing my cousins, Pablo, María José and Oscar for being like friends to me. I am so proud of them, as I know they are of me. Of course, I must mention my grandfather, who I know, if being present, could not be prouder and always believed in me.

Finally, I want to thank Gabriel for being a partner and helping me go through all the difficult times of this long process. I want you to know that I would have not come to this point without you. Thank you for all the patience and all your support.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS iii

LIST OF FIGURES vii

LIST OF TABLES ix

ABSTRACT xi

RESUMEN xii

1. CHAPTER 1 1
   1.1. Basic notation 1
   1.2. Motivation 2

2. CHAPTER 2 9
   2.1. Sobolev Spaces 9
   2.2. Trace operators 12
   2.3. Layer Potentials, BIEs and Calderón Identities 13
   2.4. Hierarchical matrices 16

3. CHAPTER 3 17
   3.1. Galerkin Boundary Element Discretization and Standard Calderón Preconditioning 17
   3.2. Modified Calderón Multiplicative Preconditioners (mCMPs) 19
      3.2.1. Construction of V_κ as mCMP 19
      3.2.2. Construction of W_κ as mCMP 22
   3.3. Hierarchical Modified Calderón Preconditioner ($\mathcal{H}$-mCMP) 25

4. CHAPTER 5 28
   4.1. Software 28
   4.2. Validation and early results 29

5. CHAPTER 6 35
5.1. Results for the V_κ as mCMP 35
5.1.1. Results for the submarine-like shape object 37
5.2. Results for the W_κ as mCMP 42
5.2.1. Results for the Fichera cube 42
6. CHAPTER 7 47
REFERENCES 48
APPENDIX 51
A. Theoretical results 52
B. Complementary content 55
B.1. Krylov Subspaces, a short insight 55
B.2. Matrix Preconditioning 58
B.3. Algebraic preconditioners 58
B.4. Some definitions on Hierrchical matrices 62

LIST OF FIGURES

1.1 Scattering problem. An incident wave u^i hits a scatterer, resulting in an outgoing wave, u^s 4

3.1 Primal (large triangles), barycentric (small triangles) and dual (dashed) meshes for piecewise-constant basis functions. 17

3.2 $\hat{P}_0 - P_1$ testing between two basis functions that intersect (gray area). Red and blue quadrilaterals are subdivisions of the dual basis function $\hat{P}_0$. 21

3.3 $\widehat{P}_1$ basis functions supports. Black dots represent dofs of type 1, 2 and 3 respectively. For the barycentric type 1 dofs, supports are given by three triangles conformed by the union of two barycentric triangles; barycentric type 2 dofs are built by four barycentric triangles; and, type 3 dofs by quadrilaterals made from two barycentric triangles. 25

4.1 Number of GMRES iterations to solve the Neumann problem over a sphere by unpreconditioned and preconditioned systems with a family of $\mathcal{H}$-mCMP preconditioners. Numbers in parenthesis represent the ratio ε/σ_{min} 32

4.2 Evolution of the number of iterations for the Dirichlet problem with an unpreconditioned and precondtioned system with a family of $\mathcal{H}$-mCMP. Numbers in parenthesis represent the ratio ε/σ_{min} 34

5.1 Fichera cube of length one, with 3,278 dofs. The mesh was obtained by using GMSH with a uniform triangulation refined at the reentrant corner. 35

5.2 Spectral behavior comparison between the unpreconditioned system and a family of $\mathcal{H}$-mCMP for the Neumann problem on a Fichera cube. 36

5.3 Assembly and solving times for preconditioned and unpreconditioned systems for the Neumann problem on a Fichera cube. 37

5.4 Latera and interior view of the submarine mesh of length 6 metres, height 2.5 metres and width 1 metre. The mesh has 3,740 vertices. 38

5.5 Spectral behavior comparison between the unpreconditioned system and a family of mCMP for the Neumann problem over a submarine grid. 39

5.6 Assembly and solving time for preconditioned and unpreconditioned systems over the submarine grid. 40

5.7 Condition number's evolution for systems over the submarine grid, preconditioned with $\mathcal{H}$-mCMP built with different leafsizes for the Neumann problem over a submarine like shaped object. Condition number grows at a rate similar to the stated in (3.9), remaining constant for $n = 400$. 41

5.8 Spectral behavior comparison between the unpreconditioned system and a family of mCMP for the Dirichlet problem, with $\kappa = 7$ over s Fichera cube. . 43

5.9 Assembly and solving time for preconditioned and unpreconditioned systems over the Fichera cube for the Dirirchlet problem. Solving times for the preconditioned systems vary between 11 and 60 seconds. 44

5.10 Spectral behavior comparison between the unpreconditioned system and a family of $\mathcal{H}$-mCMP for the Dirichlet problem, with $\kappa = 8$, over a submarine grid. 45

5.11 Assembly and solving time for preconditioned and unpreconditioned systems for the Dirichlet problem over the submarine grid (6,936 dofs). 46

LIST OF TABLES

3.1 Summary table for the mCMP developed for the hypersingular operator. *Complexity order* stands for the total number of evaluations required in the Gaussian quadrature method, which is reduced by a factor 4. 21

3.2 Summary table for the mCMP developed for the weakly singular operator. *Complexity order* stands for the total number of evaluations required in the Gaussian quadrature method, which is reduced by a factor of 2.4. 23

4.1 GMRES iteration comparison for the unpreconditioned system and the system preconditioned with different versions of the CMP, for the Neumann problem over a unit sphere of 1,442 dofs, and $\kappa = 5$. 30

4.2 $H^{\frac{1}{2}}$- and L^2-relative errors for different CMP methods applied to a Neumann problem 2.17 over a unit sphere meshed with triangles of different characteristic lengths (h). 31

4.3 Number of iterations of GMRES(20) solving the Dirichlet problem over a sphere of 2128 dofs with different versions of the $\mathcal{H}$-mCMP. 33

4.4 $H^{-\frac{1}{2}}$ and L^2 relative error evolution for different CMP methods applied to a Dirichlet problem over a unit sphere meshed with triangles of different characteristic lengths (h). 34

5.1 Number of GMRES iterations for the resolution of a system preconditioned with a family of $\mathcal{H}$-mCMP for a Neumann problem on a Fichera cube. . . . 36

5.2 GMRES iteration comparison between the unpreconditioned system and a family of $\mathcal{H}$-mCMP for the Neuman problem, with $\kappa = 8$, over a submarine grid. 38

5.3 Assembly times for families of $\mathcal{H}$-mCMP built with different leafsize values for the Neumann problem over a submarine like shaped object. Larger leafsize values take more time to be assembled. 41

5.4 GMRES comparison between the unpreconditioned system and a family of mCMP for the Dirichlet problem, with $\kappa = 7$ over s Fichera cube. 42

5.5 GMRES iteration comparison between the unpreconditioned system and a family of mCMP for the Dirichlet problem, with $\kappa = 8$, over a submarine grid. 45

ABSTRACT

Calderón multiplicative preconditioners are an effective way to improve the condition number of first kind boundary integral equations yielding provable mesh independent bounds. However, when discretizing by local low-order basis functions as in standard Galerkin boundary element methods, their computational performance worsens as meshes are refined. This stems from the barycentric mesh refinement used to construct dual basis functions that guarantee the discrete stability of L^2-pairings. Based on coarser quadrature rules over dual cells and $\mathcal{H}$-matrix compression, we propose a family of fast preconditioners that significantly reduce assembly and computation times when compared to standard versions of Calderón preconditioning for the three-dimensional Helmholtz weakly and hyper-singular boundary integral operators. Several numerical experiments validate our claims and point towards further enhancements.

Keywords: Operator Preconditioning, Calderón Precontitioning, Helmholtz equation, Hierarchical Matrices, Fast solvers, Boundary Elements Method.

RESUMEN

El uso de precondicionadores multiplicativos de Calderón es una manera efectiva de mejorar el número de condición de ecuaciones integrales de frontera de primer tipo, garantizando cotas para el número de condición, independientes de la discretización usada en el mallado. Sin embargo, al aplicar el Método de Elementos de Frontera de tipo Galerkin, el rendimiento en términos computacionales de estos precondicionadores empeora, a medida que la malla es refinada. Esto se debe al uso de un refinamiento baricéntrico de la malla, requerido en la construcción de funciones bases duales, que garantizan la estabilidad discreta de las matrices que surgen de los productos L^2 de las funciones base (matrices de Gram). Basándonos en reglas de cuadratura de menor precisión sobre celdas duales y el uso de compresiones mediante $\mathcal{H}$-mat (Matrices Jerárquicas) proponemos una familia de precondicionadores de Calderón *fast*, que significativamente reducen sus tiempos de ensamblaje, en comparación a las versiones estándar de precondicionadores de Calderón para los operadores integrales de frontera tridimensionales de Helmholtz, *weakly* y *hyper*-ingular. Una serie de experimentos numéricos validan nuestras afirmaciones y apuntan hacia mejoras futuras.

Palabras Claves: Precondicionamiento operacional, precondicionamiento de Calderón, ecuación de Helmholtz, Matrices Jerárquicas, *solvers* veloces, Método de Elementos de Frontera..

1. INTRODUCTION

1.1. Basic notation

$\mathbb{R}^d$: the d-dimensional real space with Euclidean norm.

$\mathbf{A}$: capital letters in bold font designates matrices.

$\mathbf{x}$: letters in bold font designates vectors.

$\mathbf{A}^*$: conjugate transpose of $\mathbf{A}$.

$\| \mathbf{x} \|_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p}$: p-norm of a vector.

$\| \mathbf{A} \|_p = \max_{\mathbf{x} \in \mathbb{C}^n, \mathbf{x} \neq 0} \frac{\|\mathbf{A}\mathbf{x}\|_p}{\|\mathbf{x}\|_p}$: p-norm of a matrix.

$|\mathbf{x}| = \sum_{i=1}^{d} x_i$: Norm $p = 1$ of a vector,$\mathbf{x}$.

$\rho(\mathbf{A})$: spectral radius of $\mathbf{A}$.

$\kappa_p(\mathbf{A})$: condition number of $\mathbf{A}$ (see appendix B.2)

$\| \mathbf{A} \|_2 = \rho(\mathbf{A}^*\mathbf{A})^{1/2}$: spectral norm of a matrix.

$(a, b)_X$: inner product defined on the space X between a and b.

$\overline{\Omega}$: closure of a set.

$(X)'$: topological dual of a vectorial space, X.

div$f$: stands for divergence of a scalar function, f.

$\Delta u, \nabla u$: are the standard definitions of laplacian and gradient of a function u.

$\lfloor \cdot \rfloor$: stands for the floor function (lower integer part of a function).

1.2. Motivation

Scattering problems model the behavior of waves that scatter from solid objects or propagate through non-uniform media. This phenomenon can be treated from two main approaches: direct scattering problems, where a known incoming wave hits an object, so the scattered wave is to be found, and inverse scattering problems, where both, incident and scattered waves are known and then the shape and properties of the material are unknowns.

To this research, direct scattering problems will be the main concern, and in particular, we are interested in scattering of acoustic waves. For this, we consider the propagation of sound waves in a homogeneous isotropic medium in $\mathbb{R}^3$ viewed as an inviscid fluid. We define $v = v(x,t)$, $p = p(x,t)$, $\rho = \rho(x,t)$ and $S = S(x,t)$ as the velocity, the pressure, density and specific entropy respectively. The wave motion is governed by Euler's equation, the equation of continuity, the state equation and the adiabatic hypothesis (Colton & Kress, 2012):

$$\begin{cases} \frac{\partial v}{\partial t} + (v \cdot \nabla)v + \frac{1}{\rho}\nabla p = 0 \\ \frac{\partial p}{\partial t} + \text{div}(\rho v) = 0 \\ p = f(\rho, S) \\ \frac{\partial S}{\partial t} + v \cdot \nabla S = 0 \end{cases} \tag{1.1}$$

where f is a function that depends on the nature of the fluid. By assuming that v, p, ρ and S are small perturbations of the static case ($v_0 = 0$, $p_0 = K_1$, $\rho_0 = K_2$, $S_0 = K_3$ –with K_1, K_2, K_3 constants–) one can linearize the aforementioned equations and thus obtain the wave equation:

$$\frac{1}{c^2}\frac{\partial^2 p}{\partial t^2} = \Delta p \tag{1.2}$$

Where c is the speed of sound defined by $c^2 = \frac{\partial f(\rho_0, S_0)}{\partial \rho}$.

However, all acoustic radiation problems can be boiled down to solving the wave equation, subject to certain initial and boundary conditions. For a constant frequency case, the problem reduces to solving the Helmholtz equation. From linearizing the Euler equation:

$$\frac{\partial v}{\partial t} + \frac{1}{\rho_0}\nabla p = 0 \tag{1.3}$$

we have that there exists a velocity potential $U = U(x,t)$, such that

$$v = \frac{1}{\rho_0}\nabla U$$

and

$$p = -\frac{\partial U}{\partial t}.$$

Then, the velocity potential also satisfies the wave equation, and for time harmonic waves of the form $U(x,t) = \text{Re}\{u(x)e^{-i\omega t}\}$ with frequency $\omega > 0$ we deduce that the complex valued dependent part, u satisfies the Helmholtz equation:

$$\Delta u + \kappa^2 u = 0 \tag{1.4}$$

where κ is the wave number given by $\kappa = \frac{\omega}{c}$.

To solve this problem, for simple geometries as a sphere or a cilinder, one resorts to spherical harmonics to compute the analytical solution for the Helmholtz equation. For complex geometries, in turn, the use of numerical tools is necessary to calculate good approximations for the solution u. Finite differences, Finite Elements Method, and collocation methods are widely used to this end, but in particular, the method that will concern to this investigation is the Boundary Elements Method (BEM), also called Method of Moments (MOM).

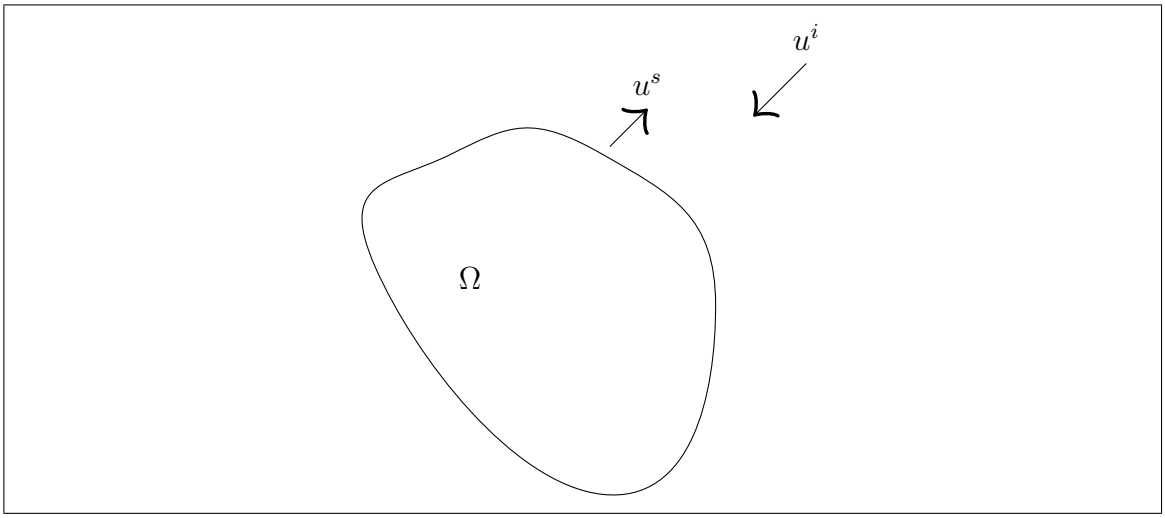

Figure 1.1. Scattering problem. An incident wave u^i hits a scatterer, resulting in an outgoing wave, u^s

BEM is based in the derivation of boundary integral equations (BIEs) from elliptic partial differential equations. In practice, the aplication of BEM is similar to Finite Elements Method, but we resort specifically to BEM as it has the characteristic of finding solutions for exterior and interior problems from the discretisation of the boundary data, instead of performing the discretisation of a whole volume. This is a suitable technique to solve 1.4 as many of its applications involve unbounded domains. As an example, we consider the classic scattering from a bounded obstacle, $\Omega \subset \mathbb{R}^3$. Given an incident field u^i that is a solution to 1.4 in $\mathbb{R}^3$, the problem is to find the scattered field u^s, such that the total field $u^t = u^i + u^s$ satisfies 1.4 in $\mathbb{R}^3 \setminus \overline{\Omega}$, as well as a boundary condition on $\partial\Omega$, which tipically could be a Neumann condition (*sound hard* obstacle):

$$\frac{\partial u^t}{\partial \hat{n}} = 0$$

or a Dirichlet condition (*sound soft* obstacle):

$$u^t|_{\partial\Omega} = 0.$$

Adding a radiation condition:

$$\lim_{R\to\infty}\int_{r=R}\left|\frac{du}{dr}-i\kappa u\right|^2 dS\to 0$$

the Neumann and Dirichlet problems can be transformed into Boundary Integral Equations (which will be presented in the next chapter) that can be solved with BEM. As we have already said, the use of this method requires only to mesh the boundary of the scattered object to find u^s , and in conclusion, BEM allows to easily solve problems in unbounded domains (Sauter & Schwab, 2010), saving computational space and assembly time of the matrices involved, in contrast to the aforementioned methods.

However, one of the main problems of applying BEM is that, in this particular research, we intend to solve a *first kind integral equation* whose discretisation leads to large systems that must be solved by iterative methods, which will not perform optimally, as the linear systems to be solved are *ill conditioned*(appendix B.2).

Given an integral operator, A on a Hilbert space $\mathcal{X}$, an integral equation is called of *first kind* if it has the form:

$$\mathsf{A}\phi = f$$

and of second kind, if it has the form:

$$(\alpha\mathsf{I}+\mathsf{A})\phi = f$$

Besides, if A is compact, then equations (1.2) and (1.2) are respectively called Fredholm integral equations of first kind and second kind. Second kind Fredholm integral equations, under suitable conditions give rise to well conditioned linear systems, because

the eigenvalues of a compact operator is composed by a sequence of eigenvalues accumulating at the origin, and then, the eigenvalues of $(\alpha \mathsf{I} + \mathsf{A})$ cluster at α.

Due to the characteristics of the integral equation to be solved, we resort to the GMRES (appendix B.1). As said before, this method has slow convergence issues when matrices are ill conditioned. Meurant and Tebbens (2015) proved that for diagonalizable matrices the norm of the residual vector depends on the eigenvalues, which if close to the origin, tend to hamper convergence. Also, proposition 4 formulated by Saad and Schultz (1986) reads:

> Supose that $\mathbf{A}$ is diagonalizable, so that $\mathbf{A} = \mathbf{X}\mathbf{D}\mathbf{X}^{-1}$ and let
>
> $$\epsilon^{(m)} = \min_{p \in \mathcal{P}_m, p(0)=1} \max_{\lambda_i} |p(\lambda_i)|$$
>
> where $\mathcal{P}_m$ is the space of polynomials of degree $\leq m$ and λ_i are the eigenvalues of the spectrum of $\mathbf{A}$
>
> Then the residual norm provided at the mth step of GMRES satisfies
>
> $$\|r_{m+1}\|_2 \leq \kappa_2(X)\epsilon^{(m)} \|r_0\|_2 . \tag{1.5}$$

Which shows a relation between the convergence of the GMRES method and the spectral condition number of $\mathbf{A}$.

Sustained by a rigorous mathematical background, *operator preconditioning* (OP) Hiptmair (2006) has shown remarkable results able to cope with ill-conditioned linear systems arising from the Galerkin discretization of boundary integral or partial differential equations. Fundamentally, OP seeks to create a well-defined endomorphism through: (i) linear operators whose induced sesquilinear forms satisfy continuous and discrete inf-sup conditions; (ii) same amounts of primal and dual degrees freedom; and, (iii) a non-singular Gram matrix linking dual and primal linear systems. Under these conditions, the preconditioned matrix is proven to be spectrally mesh independent (see (Hiptmair, 2006, Theorem 2.1)).

Among particular cases of OP for first-kind boundary integral equations, we mention *opposite order* (Steinbach & Wendland, 1998) and *Calderón* preconditioning. The former is based on canceling the symbol order of the operator whereas the latter is based on the so-called *Calderón identities*, setting specific choices for the preconditioners. With this, unlike algebraic preconditioners (appendix B.3), OP seeks to transform a first kind Fredholm integral equation into a second kind Fredholm integral equation.

Commonly, Calderón preconditioning is implemented via Calderón Multiplicative Preconditioners (CMP). For the Laplace case, examples on open curves and smooth screens can be found in Hiptmair, Jerez-Hanckes, and Urzúa-Torres (2013) and Hiptmair, Jerez-Hanckes, and Urzúa-Torres (2017); Hiptmair and Urzúa-Torres (2016), respectively. For the Electric Field Integral Equation (EFIE), prior to CMP techniques stabilization through the use of loop-star basis functions (Lee, Lee, & Burkholder, 2003; Stephanson & Lee, 2009; Yan, Jin, & Nie, 2010) as well as of the Helmholtz decomposition (Adams & Champagne, 2004; Christiansen & Nédélec, 2002) was a recurrent subject. The work of Andruilli *et al.* (Andriulli et al., 2008) introduced the CMP for the EFIE by employing dual Buffa-Christiansen (BC) basis functions (Buffa & Christiansen, 2007) after which it has seen further improvements and variations (*cf.* (Y. P. Chen, Sun, Jiang, & Chew, 2014; Cools, Andriulli, & Michielssen, 2011; Dobbelaere et al., 2015; Gossye, Huynen, Ginste, De Zutter, & Rogier, 2018)).

Despite the advantages of the CMP, in order to derive a dual-basis preconditioner, discretization on a barycentric refinement of the original mesh is performed. This entails a six-fold increase in computational complexity, rendering the method often impractical in realistic scenarios. Hence, the question arises as whether one can avoid barycentric refinement, or at least, how to improve its efficiency. Recent efforts have been devoted to tackle this issue. In Adrian, Andriulli, and Eibert (2018) a refinement free version of the CMP for the EFIE is constructed by leveraging on spectral equivalences between the weakly and hyper-singular operators and the Laplace-Beltrami operator. In parallel, a fast CMP for the EFIE was proposed in Escapil-Inchauspé and Jerez-Hanckes (2019) which

relies on a splitting of solution and preconditioner accuracies using hierarchical matrices ($\mathcal{H}$-mat) approximations with very promising results.

This research discusses two improvements over the original CMP for the Helmholtz case. Following the ideas of Escapil-Inchauspé and Jerez-Hanckes (2019), we preserve the most relevant information of the operator preconditioner by heavily compressing far-field interactions. This results in an almost equally effective preconditioner but much cheaper to compute when compared to the original one. Based on $\mathcal{H}$-mat approximations, we further accelerate assembly times by applying more efficient quadrature rules to integrate the dual basis functions. As we will show, joining these two techniques leads to a family of preconditioners whose efficiency depends on the accuracy of integration rules and on parameter choices used to construct their $\mathcal{H}$-mat approximation. As it will be shown, the $\mathcal{H}$-mat compression tolerance is a crucial parameter determining the performance of these preconditioners. Also, our modified CMPs inherit the shortcomings of the standard CMP with respect to growing wavenumbers as expected.

The following chapter states the mathematical background for the Helmholtz BIEs formulations. Chapter 3 proposes modified versions of the Calderón preconditioner, chapter 4 explains how the problems were solved computationally and chapter 5 shows the results obtained from the methods proposed. Finally, chapter 6 states the conclusions of this research.

2. CHAPTER 2: PRELIMINARIES

2.1. Sobolev Spaces

Let $\mathbb{R}^d$, $d = 1, 2, 3$ and let $\Omega \in \mathbb{R}^d$ For any integer $r \geq 0$. Following the definitions already stated by McLean (2000), we write:

$$\mathcal{C}^r(\Omega) = \{u : \partial^\alpha u \text{ exists and is continuous on } \Omega \text{ for } |\boldsymbol{\alpha}| \leq r\}$$

and

$$\mathcal{C}^\infty(\Omega) = \cap_{r\geq 0}\mathcal{C}^r(\Omega).$$

The support of u will be denoted by supp u.

Definition 2.1. *If $K \subset \Omega$ is compact, then we define:*

$$\mathcal{C}^r_K(\Omega) := \{u \in \mathcal{C}^r(\Omega) : \textit{supp } u \subset K\}, \;\; \mathcal{C}^\infty_K(\Omega) := \cap_{r\geq 0}\mathcal{C}^r_K(\Omega)$$

and

$$\mathcal{C}^\infty_0(\Omega) = \{u : u \in \mathcal{C}^\infty_K(\Omega) \textit{ for some } K \in \Omega\}.$$

We also define $L^2(\Omega) = \{u : \int_\Omega |u| d\mathbf{x} < \infty\}$.

Definition 2.2. *Let $f \in L^2(\Omega)$. For $i \in \{1 \dots d\}$. A class, $\partial_{x_i} f$ of measurable functions differing only in a zero measure set, is called a weak derivative of f respect to x_i, if*

$$(f, \partial_{x_i} g)_{L^2(\Omega)} = -(\partial_{x_i} f, g)_{L^2(\Omega)} \;\; \forall g \in \mathcal{C}^\infty_0(\Omega). \tag{2.1}$$

Having the weak derivative definition, and given a vector $\mathbf{m} \in \mathbb{R}^d$ we write the differential operator for $f \in L^2(\Omega)$ as:

$$\partial^{\boldsymbol{\alpha}} f = \begin{cases} f \text{ if } \boldsymbol{\alpha} = \mathbf{0} \\ \partial_{x_1}^{\alpha_1} \dots \partial_{x_d}^{\alpha_d} f \text{ if } \boldsymbol{\alpha} \neq \mathbf{0} \end{cases} .$$

Thus,

Definition 2.3. *Let $s \in \mathbb{N}$. The space*

$$H^s(\Omega) := \{f \in L^2 : \partial^{\boldsymbol{\alpha}} f \in L^2(\Omega) \forall \boldsymbol{\alpha} \in \mathbb{R}^d \ |\boldsymbol{\alpha}| \leq s\}$$

is called a Sobolev space in Ω.[1]

The completion of $L^2(\Omega)$ implies that $H^s(\Omega)$ becomes a Banach space with the norm:

$$\| f \|_{H^s(\Omega)} = \sum_{\boldsymbol{\alpha}:|\boldsymbol{\alpha}|\leq s} \|\partial^{\boldsymbol{\alpha}} f\|^2_{L^2(\Omega)} .$$

In the same way, $H_0^s(\Omega)$ stands for the completion of $\mathcal{C}_0^\infty$

For fractional Sobolev spaces, we introduce thei Sobolev-Slobodeckii semi-norm:

$$|\varphi|_l^2 := \int_\Omega \int_\Omega \frac{|\varphi(\mathbf{x}) - \varphi(\mathbf{y})|^2}{\|\mathbf{x} - \mathbf{y}\|_2^{d+2l}} d\mathbf{x} d\mathbf{y}$$

with $l \in (0, 1)$ and $\varphi \in L^2(\Omega)$. As s has a fractional order, we split $s = \lfloor s \rfloor + l$ to define the fractional Sobolev space:

$$H^s(\Omega) := \{f \in H^{\lfloor s \rfloor} : |\partial^{\boldsymbol{\alpha}} f|_l < \infty \text{ for } |\boldsymbol{\alpha}| = s\}$$

[1] This is a particular class of Sobolev spaces, which is originally defined with the L^p norm, but in this case we consider only define the L^2 case

with the norm:

$$\|\varphi\|^2_{H^s(\Omega)} = \|\varphi\|^2_{H^{\lfloor s \rfloor}(\Omega)} + |\varphi|^2_{H^s(\Omega)}\,,$$

where

$$|\varphi|^2_{H^s(\Omega)} = \sum_{|\boldsymbol{\alpha}|=\lfloor s \rfloor} |\partial^{\boldsymbol{\alpha}}\varphi|^2_l\,.$$

With this norm, arises the inner product:

$$(\varphi,\phi)_{H^s(\Omega)} := (\varphi,\phi)_{H^{\lfloor s \rfloor}(\Omega)} + \sum_{\boldsymbol{\alpha}=\lfloor s \rfloor} \int_\Omega \int_\Omega \frac{\overline{(\partial^{\boldsymbol{\alpha}}\varphi(\mathbf{x}) - \partial^{\boldsymbol{\alpha}}\varphi(\mathbf{y}))}(\partial^{\boldsymbol{\alpha}}\phi(\mathbf{x}) - \partial^{\boldsymbol{\alpha}}\phi(\mathbf{y}))}{\|\mathbf{x}-\mathbf{y}\|_2^{d+2l}} d\mathbf{x}d\mathbf{y}$$

For Sobolev spaces on boundaries, we can define a piecewise parametrization on $\Gamma := \partial\Omega$:

$$\Gamma = \sum_i \Gamma_i \quad \Gamma_i := \{\mathbf{x} \in \mathbb{R}^d : \mathbf{x} = \mathcal{X}(\zeta) \text{ for } \zeta \in \mathcal{T}_i \subset \mathbb{R}^{d-1}\}.$$

We also consider a partition of unity, $\{\varphi\}_{i=1}^p$ of non negative cut off functions $\phi_i \in \mathcal{C}_0^\infty(\mathbb{R}^d)$ satisfying:

$$\sum_i \varphi_i(\mathbf{x}) = 1 \text{ for } \mathbf{x} \in \Gamma \quad \varphi = 0 \text{ for } \mathbf{x} \in \Gamma \setminus \Gamma_i.$$

Then, for any function f defined on the boundary we can write:

$$f(\mathbf{x}) = \sum_i \varphi_i(\mathbf{x}) f(\mathbf{x}) = \varphi_i(\mathcal{X}(\zeta)) f(\mathcal{X}(\zeta)) \quad \text{for } \mathbf{x} \in \mathcal{T}_i \subset \mathbb{R}^{d-1}.$$

Thus, we can define the Sobolev norm:

$$\|f(\mathbf{x})\|^2_{H^s(\Gamma)} := \sum_i \|\varphi_i(\mathcal{X}(\zeta))f(\mathcal{X}(\zeta))\|^2_{H^s(\mathcal{T}_i)}\,.$$

For $s \in (0,1)$ we resort again to the Sobolev-Slobodeckii norm:

$$\|f\|^2_{H^s(\Gamma)} = \|f\|^2_{L^2(\Gamma)} + |\varphi|^2_{H^s(\Gamma)}\,.$$

Let X be either a domain or a surface. Then the following notation is used for the dual space:

$$H^{-s}(X) := (H^s_0(X))', \quad s \geq 0.$$

Note that in the case of closed surfaces ($X = \Gamma$) the boundary of X is the empty set and therefore $H^s_0(X) = H^s(X)$.

2.2. Trace operators

Definition 2.4. *(Sauter & Schwab, 2010, Definition 2.6.1) Let Ω be a (possibly unbounded) domain. The space $H^s_{loc}(\Omega)$ contains all continuous, linear functionals (distributions) on $\mathcal{C}^\infty_0$. Thus, $f \in (\mathcal{C}^\infty_0)'$ with the property that $\varphi f \in H^s(\Omega)$ for all $\varphi \in \mathcal{C}^\infty_0$.*

Theorem 2.1. *(Sauter & Schwab, 2010, Theorem 2.6.8) Let Ω^- be a Lipschitz domain, with boundary Γ and $\Omega^+ := \mathbb{R}^d \setminus \overline{\Omega^-}$.*

a) *for all $s \in (1/2, 3/2)$ there exists a continuous, linear trace operator:*

$$\gamma_D : H^s_{loc}(\mathbb{R}^d) \to H^{s-1/2}(\Gamma) \tag{2.2}$$

with

$$\gamma_D f := \lim_{\tilde{x}\in\Omega\to x\in\Gamma} f(x) \;\; \forall f \in \mathcal{C}^0(\mathbb{R}^d).$$

b) *For $l \in \{-,+\}$there exist one-sided, continuous, linear trace operators*

$$\gamma_D^l : H^s_{loc}(\Omega^l) \to H^{s-1/2}(\Gamma) \tag{2.3}$$

with

$$\gamma_D^l f := \lim_{\tilde{x}\in\Omega\to x\in\Gamma} f(x) \;\; \forall f \in \mathcal{C}^0(\overline{\Omega}^l)$$

and $\gamma_D^+ f = \gamma_D^- f = \gamma_D f$ almost everywhere for all $f \in H^s_{loc}(\mathbb{R}^d)$

Also, the Neumann trace id defined as:

$$\gamma_N f := (\mathbf{n}, \gamma_D \nabla f)_{\mathbb{R}^d} \tag{2.4}$$

and then:

$$\gamma := \begin{pmatrix} \gamma_D u \\ \gamma_N u \end{pmatrix}. \tag{2.5}$$

Finally, if γ^+ and γ^- denote the exterior and interior trace respectively, then jump and average operators are defined as:

$$[\gamma]_\Gamma := \gamma^+ - \gamma^- \tag{2.6}$$

$$\{\gamma\}_\Gamma := \frac{1}{2}(\gamma^+ + \gamma^-). \tag{2.7}$$

2.3. Layer Potentials, BIEs and Calderón Identities

Let $\mathbb{R}^3$ be split into a bounded Lipschitz domain Ω^- of boundary Γ and such that $\Omega^+ := \mathbb{R}^3 \setminus \Omega^-$. For a wavenumber $\kappa > 0$ and an incident plane wave, we aim at solving

the homogenous Helmholtz equation for the scattered wave

$$\Delta u + \kappa^2 u = 0, \quad \text{in } \Omega^- \cup \Omega^+, \tag{2.8}$$

together with suitable boundary conditions given below. For exterior problems, i.e. domains considering Ω^+, the Sommerfeld radiation condition

$$\lim_{R\to\infty} \int_{r=R} \left| \frac{du}{dr} - i\kappa u \right|^2 dS \to 0$$

is required for wellposedness McLean (2000).

Let $G_\kappa(z) = \frac{\exp(i\kappa|z|)}{4\pi|z|}$ denote the Helmholtz fundamental solution or kernel Sauter and Schwab (2010). With it, one defines the standard single and double layer potentials Sauter and Schwab (2010):

$$\mathcal{S}_\kappa(\varphi)(\mathbf{x}) := \int_\Gamma G_\kappa(\mathbf{x}-\mathbf{y})\varphi(\mathbf{y})dS(\mathbf{y}) : H^{-\frac{1}{2}}(\Gamma) \to H^1(\Omega^- \cup \Omega^+),$$
$$\mathcal{D}_\kappa(v)(\mathbf{x}) := \int_\Gamma \nabla G_\kappa(\mathbf{x}-\mathbf{y}) \cdot \hat{\mathbf{n}}(\mathbf{y})v(\mathbf{y})dS(\mathbf{y}) : H^{\frac{1}{2}}(\Gamma) \to H^1(\Omega^- \cup \Omega^+),$$

with extensions to standard trace and volume Sobolev spaces $H^{\pm\frac{1}{2}}(\Gamma)$ and $H^1(\Omega^- \cup \Omega^+)$.

The above defined potentials satisfy the *jump relations* (Sauter & Schwab, 2010, Theorem 3.3.1):

$$[\gamma_D \mathcal{S}_\kappa(\lambda_N)]_\Gamma = 0, \quad [\gamma_N \mathcal{S}_\kappa(\lambda_N)]_\Gamma = -\lambda_N,$$
$$[\gamma_D \mathcal{D}_\kappa(\lambda_D)]_\Gamma = \lambda_D, \quad [\gamma_N \mathcal{D}_\kappa(\lambda_D)]_\Gamma = 0.$$

Solutions of (2.8) can be given in terms of the *integral representation formula* (Sauter & Schwab, 2010, Theorem 3.1.13):

$$u = \mathcal{D}_\kappa([\gamma_D u]) - \mathcal{S}_\kappa([\gamma_N u]) \quad \text{in } \Omega^+ \cup \Omega^-. \tag{2.9}$$

By taking traces over (2.9) and using the above jump relations, one defines the following BIOs:

$$\mathsf{V}_\kappa := \{\gamma_D \mathcal{S}_\kappa\}_\Gamma : H^{-\frac{1}{2}}(\Gamma) \to H^{\frac{1}{2}}(\Gamma), \tag{2.10}$$

$$\mathsf{K}'_\kappa := \{\gamma_N \mathcal{S}_\kappa\}_\Gamma : H^{-\frac{1}{2}}(\Gamma) \to H^{-\frac{1}{2}}(\Gamma), \tag{2.11}$$

$$\mathsf{K}_\kappa := \{\gamma_D \mathcal{D}_\kappa\}_\Gamma : H^{\frac{1}{2}}(\Gamma) \to H^{\frac{1}{2}}(\Gamma), \tag{2.12}$$

$$\mathsf{W}_\kappa := -\{\gamma_N \mathcal{D}_\kappa\}_\Gamma : H^{\frac{1}{2}}(\Gamma) \to H^{-\frac{1}{2}}(\Gamma), \tag{2.13}$$

dubbed weakly singular, (adjoint) double layer and hypersingular operators, respectively, and whose continuity and coercivity properties can be found in (Sauter & Schwab, 2010, Chapter 3). With these, we can derive interior and exterior *Calderón projectors*:

$$\begin{bmatrix} \gamma_D^\pm u \\ \gamma_N^\pm u \end{bmatrix} = \mathbf{C}^\pm \gamma^\pm u := \begin{bmatrix} \frac{1}{2}\mathsf{I} \pm \mathsf{K}_\kappa & \mp \mathsf{V}_\kappa \\ \mp \mathsf{W}_\kappa & \frac{1}{2}\mathsf{I} \mp \mathsf{K}'_\kappa \end{bmatrix} \begin{bmatrix} \gamma_D^\pm u \\ \gamma_N^\pm u \end{bmatrix} \tag{2.14}$$

From the above identities, one deduces the first kind BIEs that we aim to solve:

$$\mp \mathsf{W}_\kappa \gamma_D^\pm u = \left(\frac{1}{2}\mathsf{I} \pm \mathsf{K}'_\kappa\right) \gamma_N^\pm u, \tag{2.15}$$

$$\pm \mathsf{V}_\kappa \gamma_N^\pm u = \left(\frac{1}{2}\mathsf{I} \pm \mathsf{K}_\kappa\right) \gamma_D^\pm u. \tag{2.16}$$

These BIEs are derived by the so-called *direct method*. Alternatively and indirectly, one can derive Ansatz representations by solely using either the single or double layer potentials. More importantly, for closed Lipschitz boundaries, (2.14) leads to the Calderón identities:

$$\mathsf{V}_\kappa \mathsf{W}_\kappa = \frac{1}{4}\mathsf{I}_{H^{\frac{1}{2}}(\Gamma)} - \mathsf{K}_\kappa^2, \tag{2.17}$$

$$\mathsf{W}_\kappa \mathsf{V}_\kappa = \frac{1}{4}\mathsf{I}_{H^{-\frac{1}{2}}(\Gamma)} - \mathsf{K}_\kappa'^2, \tag{2.18}$$

which can be interpreted as preconditioned versions of (2.15) and (2.16), i.e. the weakly singular operator preconditions the hyper-singular one and viceversa. Identity operators

are tagged to be understood in their respectively functional space. Clearly, W_κ is an operator of order +1 while V_κ is of order -1, thus explaining the opposite order preconditioning (Steinbach & Wendland, 1998). One should carefully handle the squared terms in (2.17) and (2.18), as only sufficiently smooth boundaries one can prove that these terms are compact (Sauter & Schwab, 2010, Section 3.9.3).

2.4. Hierarchical matrices

Hierarchical matrices are a method of matrix compression used for saving memory, space and accelerating the resolution of dense matrix systems. This is the reason why in conjunction with the Fast Multipole Methods, Hierarchical matrices are widely used for solving linear systems coming from BEM.

Some definitions on Hierarchical matrices can be found in different sources (Bebendorf, 2008; Hackbusch, 2015), but for the sake of completeness we offer a brief definition of Hierarchical matrices and complementary concepts that can be found in the appendix B.4.

Finally,

Definition 2.5. *A Hierarchical matrix is defined by:*

> *Let $L \in \mathbb{R}^{\mathcal{I}\times\mathcal{I}}$ be a matrix and T a block tree of $\mathcal{I} \times \mathcal{I}$ consisting of admissible and nonadmissible leaves. Let Let $k \in \mathbb{N}$L is called an $\mathcal{H}$-matrix of blockwise rank k, if for all admissible leaves $\tau \times \sigma \in I_{\mathcal{I}\times\mathcal{I}}$, rank$(L_{|\tau\times\sigma}) \leq k$ (Börm, Grasedyck, & Hackbusch, 2003)*

Thus, the use of $\mathcal{H}$-matrices should lead to a less expensive storage for matrices in terms of space and time. In the following, we take advantage of this fact to design a fast Calderón preconditioner.

3. CHAPTER 3: FAST CALDERÓN PRECONDITIONING

3.1. Galerkin Boundary Element Discretization and Standard Calderón Preconditioning

We now perform Galerkin low-order discretizations of the BIEs (2.15) and (2.16) as well as introduce their standard OP preconditioning based on barycentric mesh refinement to construct dual bases. Elements of the discrete Galerkin matrices arising from operators in (2.17) and (2.18) are built upon piecewise -constant or -linear basis functions as follows.

Let $\mathcal{T}_h$ refer to either primal (Γ_h), dual ($\widehat{\Gamma}_h$) or barycentrically refined primal ($\overline{\Gamma}_h$) mesh (see Figure 3.1). For $p = 0, 1$, we write $P_p \equiv P_p(\Gamma_h)$ for the space of piecewise constant or linear functions over the primal mesh Γ_h, respectively. Similarly for barycentric ($\overline{P}_p$) and dual mesh ($\widehat{P}_p$) basis functions. Given a primal mesh Γ_h, we also define n_T, n_V and n_E as the number of triangles, vertices and edges, respectively. With this, we must solve the discrete versions of (2.15) and (2.16):

$$\mathbf{W}_{\kappa,h}\mathbf{u} = \mathbf{b}, \tag{3.1}$$

$$\mathbf{V}_{\kappa,h}\mathbf{v} = \boldsymbol{\eta}, \tag{3.2}$$

where we have skipped the construction of the right-hand sides for the sake of brevity.

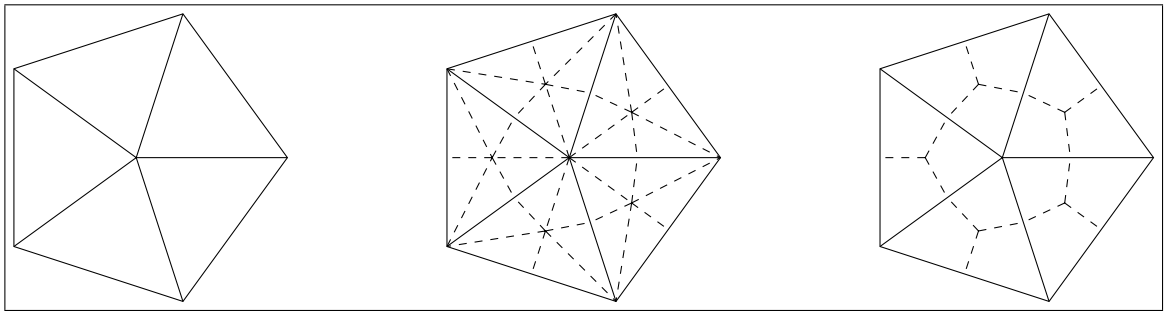

Figure 3.1. Primal (large triangles), barycentric (small triangles) and dual (dashed) meshes for piecewise-constant basis functions.

As aforementioned, the condition number of these systems deteriorates as $\mathcal{O}(h^{-1})$, thus requiring preconditioning. Hence, we will follow the OP strategy described in Theorem 0.1 (see appendix A). In the context of Calderón preconditioning, let $\mathsf{A} : X \to Y$ and $\mathsf{B} : Y \to X$ be two linear operators, namely the weakly singular integral operator and the hyper-singular integral operator and vice-versa, (*cf.* (2.17) and (2.18) wherein X and Y are alternatively $H^{\pm 1/2}(\Gamma)$). We use a pair of discrete spaces, $X_h \subset X$ and $Y_h \subset Y$ over meshes of characteristic length h, such that $\dim(X_h) = \dim(Y_h)$ to find a discretization for A, and B ($\mathbf{A}_h$ and $\mathbf{B}_h$ respectively). We also assume that X and Y are dual to each other and use a stable L^2-pairing, $\mathbf{G}_h^{X_h,Y_h} = (\phi_h, \varphi_h)_{L^2}$ for $\phi_h \in X_h$ and $\varphi_h \in Y_h$). With these elements, the preconditioned matrix takes the form:

$$(\mathbf{G}_h^{X_h,Y_h})^{-1}\mathbf{B}_h(\mathbf{G}_h^{X_h,Y_h})^{-T}\mathbf{A}_h$$

Specific standard CMP for the BIEs considered read:

- V_κ *as CMP* (2.17). The induced bilinear form of the operator W_κ is discretized over the primal mesh using $X_h = P_1 \subset H^{\frac{1}{2}}(\Gamma)$ whereas V_κ is approximated in $Y_h = \widehat{P}_0 \subset H^{-\frac{1}{2}}(\Gamma)$. However, $\widehat{P}_0$ basis functions over the dual mesh are built by assembling the discrete version of V_κ over $\overline{P}_0$ to then apply an averaging matrix $\mathbf{\Sigma}$ to obtain a dual basis function (see below).
- W_κ *as CMP* (2.18). We use as primal boundary element space $X_h = P_0 \subset H^{-\frac{1}{2}}(\Gamma)$ to discretize V_κ and $Y_h = \widehat{P}_1 \subset H^{\frac{1}{2}}(\Gamma)$ for W_κ.

For $\widehat{P}_1$ basis functions, the degrees of freedom are determined by the triangles in the primal mesh Buffa and Christiansen (2007). These are obtained as the weighted sum of piecewise linear basis functions defined over the barycentric mesh corresponding to each triangle. Based on their geometric relation, the degrees of freedom (dofs) for piecewise linear barycentric basis functions are assigned weights as follows:

(i) Barycentric nodes at the center of each primal triangle (weight 1.0);

(ii) Barycentric nodes that lie between two primal nodes (weight 0.5); and,

(iii) Primal mesh nodes, whose weights are given by $\frac{1}{n_{t_i}}$, where n_{t_i} is the number of primal triangles adjacent to the node i.

Thus, the standard CMP strategy reads:

$$(\mathbf{G}_h^{X_h,Y_h})^{-1}\mathbf{\Sigma}\mathbf{B}_h^{\overline{Y}_h,\overline{Y}_h}\mathbf{\Sigma}^T(\mathbf{G}_h^{X_h,Y_h})^{-T}\mathbf{A}_h^{X_h,X_h}\mathbf{u} = (\mathbf{G}_h^{X_h,Y_h})^{-1}\mathbf{\Sigma}\mathbf{B}_h^{\overline{Y}_h,\overline{Y}_h}\mathbf{\Sigma}^T(\mathbf{G}_h^{X_h,Y_h})^{-T}\mathbf{b}, \tag{3.3}$$

where $\mathbf{\Sigma}$ is an averaging matrix, which for $\mathbf{W}_{\kappa,h}^{\widehat{P}_1,\widehat{P}_1}$ is obtained as described in (Hiptmair & Urzúa-Torres, 2016, Section 3.3). To compute $\mathbf{\Sigma}$ for $\mathbf{V}_{\kappa,h}^{\widehat{P}_0,\widehat{P}_0}$, we resort to the strategy followed in BEMPP, which is different to the one used in (Hiptmair & Urzúa-Torres, 2016, Section 3.3). Specifically, the entries of $\mathbf{\Sigma}$ are given by:

$$\Sigma[i][j] = \begin{cases} 1 & \text{if the barycentric triangle } j \text{ belongs to the dual cell } i, \\ 0, & \text{otherwise} \end{cases}$$

In the case of V_κ as preconditioner (2.17), the matrix $\mathbf{B}_h^{Y_h,Y_h} = \mathbf{V}_{\kappa,h}^{\overline{P}_0,\overline{P}_0}$ is of size $6n_T \times 6n_T$ over the barycentric mesh. Opposingly, when the hypersingular operator W_κ acts as preconditioner, standard CMP reads $\mathbf{B}_h^{Y_h,Y_h} = \mathbf{W}_{\kappa,h}^{\overline{P}_1,\overline{P}_1}$ and is a $(n_T + n_V + n_E) \times (n_T + n_V + n_E)$ matrix built over the barycentric mesh.

3.2. Modified Calderón Multiplicative Preconditioners (mCMPs)

In what follows, we present the first building block for the construction of efficient CMPs, referred to as modified CMPs (mCMPs). Specifically, we aim at avoiding computing the averaging matrix $\mathbf{\Sigma}$ by directly building preconditioners on approximations of the dual mesh employing low accuracy numerical integration routines.

3.2.1. Construction of V_κ as mCMP

Consider the preconditioning strategy summarized in (2.17) and matrix $\mathbf{V}_{\kappa,h}^{\widehat{P}_0,\widehat{P}_0}$. As shown in Figure 3.1, the supports of $\widehat{P}_0$ basis functions are disjoint and obtained as the sum

of barycentric piecewise-constant basis functions, $\overline{P}_0$, that occupy the dual basis function support. This is the cornerstone of the mCMP as instead of integrating over each barycentric triangle support, other quadrature rules are directly applied over the dual basis support. This immediately translates into considering less quadrature points. However, the polygonal shape of the dual basis support depends on the amount of barycentric triangles therein. Hence, instead of integrating over the whole support, a quadrilateral mesh is considered: the dual support will be divided into coarser partitions than the barycentric one.

More succinctly, we exchange the standard CMP preconditioner (3.3) for the following form:

$$\mathbf{G}_h^{-1}\mathbf{V}_{\kappa,h}^{\widehat{P}_0,\widehat{P}_0}\mathbf{G}_h^{-T}\mathbf{W}_{\kappa,h}^{P_1,P_1}\mathbf{u} = \mathbf{G}_h^{-1}\mathbf{V}_{\kappa,h}^{\widehat{P}_0,\widehat{P}_0}\mathbf{G}_h^{-T}\mathbf{b}, \tag{3.4}$$

where $\mathbf{V}_{\kappa,h}^{\widehat{P}_0,\widehat{P}_0}$ is the Galerkin $(n_V \times n_V)$-matrix for the weakly singular operator over the dual mesh and $\mathbf{W}_{\kappa,h}^{P_1,P_1}$ is the discrete version of the hypersingular opetator over the primal mesh.

Computationally, self-interaction matrix entries of $\mathbf{V}_{\kappa,h}^{\widehat{P}_0,\widehat{P}_0}$ are obtained as in the original method: barycentric triangles are used which are then added up to deliver the matrix entry for the dual cell. For cross-interaction terms, quadrilateral elements are used by merging couples of barycentric triangles instead of working on the individual the barycentric triangles. This entails modifying quadrature rules: for piecewise constant basis functions, one point per triangular element is used in the original method whereas one point per quadrilateral element is computed for the modified method. Hence, one can expect at least an improvement of factor four in matrix entries computations when compared to the barycentric case, i.e. evaluations on the dual mesh are $9n_T^2 + 6n_T$ versus $36n_T^2$ directly on the barycentric mesh (table 3.1 summarizes this fact). For Gram matrices $\mathbf{G}_h$, dual pairings over $\widehat{P}_0$-P_1 are approximated via quadrature rules over the dual cell support (*cf.* Figure 3.2). Algorithm 1 summarizes cross-interaction computations for the weakly

singular mCMP[1].

Table 3.1. Summary table for the mCMP developed for the hypersingular operator. *Complexity order* stands for the total number of evaluations required in the Gaussian quadrature method, which is reduced by a factor 4.

Preconditioner	Standard CMP	mCMP
Preconditioning strategy	$\mathbf{G}_h^{-1}\Sigma\mathbf{V}_{\kappa,h}^{\overline{P}_0,\overline{P}_0}\Sigma^T\mathbf{G}_h^{-T}\mathbf{W}_{\kappa,h}^{P_1,P_1}$	$\mathbf{G}_h^{-1}\mathbf{V}_{\kappa,h}^{\widehat{P}_0,\widehat{P}_0}\mathbf{G}_h^{-T}\mathbf{W}_{\kappa,h}^{P_1,P_1}$
Complexity order	$36n_T^2$	$9n_T^2+6n_T$

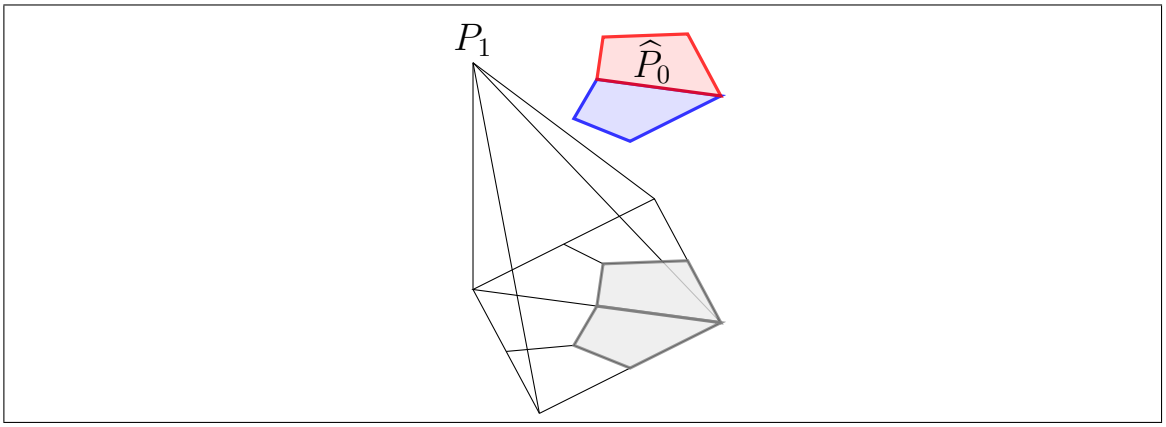

Figure 3.2. $\hat{P}_0 - P_1$ testing between two basis functions that intersect (gray area). Red and blue quadrilaterals are subdivisions of the dual basis function $\hat{P}_0$.

By Theorem 0.2 (see Appendix), the spectral condition number, denoted by κ_2, of the new preconditioned system is bounded as follows

$$\kappa_2(\mathbf{G}_h^{-1}\mathbf{V}_{\kappa,h}^{\widehat{P}_0,\widehat{P}_0}\mathbf{G}_h^{-T}\mathbf{W}_{\kappa,h}^{P_1,P_1}) \leq K\kappa_2\left(\mathbf{G}_h^{-1}\mathbf{V}_{\kappa,h}^{\widehat{P}_0,\widehat{P}_0}(\mathbf{\Sigma}\mathbf{V}_{\kappa,h}^{\overline{P}_0,\overline{P}_0}\mathbf{\Sigma}^T)^{-1}\mathbf{G}_h\right), \tag{3.5}$$

[1] IE stands for integration elements: Jacobian, quadrature weights, evaluation points.

Algorithm 1 Cross interaction calculations of V_κ mCMP.

input: Grid $\mathcal{T}_h$ (vertices n_{t_i}–number of quadrilaterals surrounding the dof i–, barycentric vertices, quadrilaterals obtained from merging barycentric triangles), κ.
for $(i,j) \in \{1,\dots,n_V\}$ **do**
  for each quadrilateral $k \in \{1,\dots,n_{t_i}\}$ surrounding the dof i **do**
    • compute IE of the quadrilateral k.
    for each quadrilateral $l \in \{1,\dots,n_{t_j}\}$ surrounding the dof j **do**
      • compute IE of the quadrilateral l.
      • compute cross-interaction from IE coming from quadrilaterals k and l and save in v.
    end for
  end for
  $\mathbf{V}_h^{\widehat{P}_0,\widehat{P}_0}[i][j] =$ sum of the entries of v.
end for
output: $\mathbf{V}_h^{\widehat{P}_0,\widehat{P}_0}$

wherein K is a bound for $\kappa_2(\mathbf{P}_h\mathbf{W}_{\kappa,h})$ given in Theorem 0.1. Thus, the new condition number depends on the factor

$$\kappa_2(\mathbf{G}_h^{-1}\mathbf{V}_{\kappa,h}^{\widehat{P}_0,\widehat{P}_0}(\mathbf{\Sigma}\mathbf{V}_{\kappa,h}^{\overline{P}_0,\overline{P}_0}\mathbf{\Sigma}^T)^{-1}\mathbf{G}_h) \tag{3.6}$$

so that if $\mathbf{V}_{\kappa,h}^{\widehat{P}_0,\widehat{P}_0} \approx \mathbf{\Sigma}\mathbf{V}_{\kappa,h}^{\overline{P}_0}\mathbf{\Sigma}^T$, then $\mathbf{G}_h^{-1}\mathbf{V}_{\kappa,h}^{\widehat{P}_0,\widehat{P}_0}(\mathbf{\Sigma}\mathbf{V}_{\kappa,h}^{\overline{P}_0,\overline{P}_0}\mathbf{\Sigma}^T)^{-1}\mathbf{G}_h \approx \mathbf{I}$. Thus, (3.6) will remain close to one.

3.2.2. Construction of W_κ as mCMP

We now solve (2.18) following previous ideas: avoiding the use of the averaging matrix $\mathbf{\Sigma}$ and building $\mathbf{W}_{\kappa,h}^{\widehat{P}_1,\widehat{P}_1}$ directly over the dual mesh.

For cross-interactions, barycentric supports corresponding to type 1 dofs can be merged into coarser triangles. At the same time, barycentric supports of type 3 dofs can be merged into quadrilaterals as in the case of $\widehat{P}_0$ basis functions. Figure 3.3 illustrates this idea and shows examples of adjacent dual basis supports for dofs of type 1, 2 and 3. Merging barycentric cells into coarser ones allows the use, in the case of dofs of type 1, of three quadrature points every two merged barycentric triangles instead of three points per barycentric triangle. In the case of type 3 dofs, two merged barycentric triangles lead to

quadrilaterals for which four quadrature points are employed instead of three points per barycentric triangle.

Interactions among overlapping dual basis functions occur when two basis functions have a common node, a common edge or are the same. Here, we again use the standard CMP algorithm: interactions are calculated over the barycentric mesh and then added up to a dual cell instead of merging the barycentric triangles.

As expected, the proposed hypersingular mCMP has lower computational complexity than the standard version for the integration algorithms due to the use of coarser meshes. For type 1 dofs, quadrature points are reduced from $18n_T$ to $9n_T$ whereas for type 3 reduce from $18n_T$ –three points per barycentric triangle– to $12n_T$ –4 points every two barycentric triangles. For type 2 dofs, the number of computations remains the same ($12n_E$). Thus, we reduce a total number of evaluations from $(36n_T + 12n_E) \times (36n_T + 12n_E)$ to $(21n_T + 12n_E) \times (21n_T + 12n_E)$. Assuming that $n_T \approx 2n_E$, we obtain a global reduction from $42n_T \times 42n_T$ to $27n_T \times 27n_T + 42n_T$ –coming from the calculation of overlaping interactions– which is reduction of a factor ≈ 2.4 (table 3.2 summarizes this fact). Algorithm 2 summarizes the cross interaction calculations for the hypersingular mCMP.

Table 3.2. Summary table for the mCMP developed for the weakly singular operator. *Complexity order* stands for the total number of evaluations required in the Gaussian quadrature method, which is reduced by a factor of 2.4.

Preconditioner	Standard CMP	mCMP
Preconditioning strategy	$\mathbf{G}_h^{-1}\Sigma\mathbf{W}_{\kappa,h}^{\overline{P}_1,\overline{P}_1}\Sigma^T\mathbf{G}_h^{-T}\mathbf{V}_{\kappa,h}^{P_0,P_0}$	$\mathbf{G}_h^{-1}\mathbf{W}_{\kappa,h}^{\widehat{P}_1,\widehat{P}_1}\mathbf{G}_h^{-T}\mathbf{V}_{\kappa,h}^{P_0,P_0}$
Complexity order	$42n_T \times 42n_T$	$27n_T \times 27n_T + 42n_T$

Algorithm 2 Cross interaction calculations of W_κ mCMP

input: Grid $\mathcal{T}_h$ (information of triangles , $n_{t_{jl}}$–number of primal triangles surrounding the dof jl–, barycentric vertices, etc.), κ, coefficient arrays (v_{c1}, v_{c2}), cross interaction arrays (v_{ci1}, v_{ci2})
for $(i, j) \in \{1, \ldots, n_T\}$ **do**
  for each barycentric vertex , k, in the triangle i **do**
    if vertex k is type 1 **then** merge each couple of barycentric triangles surrounding vertex k into triangles and compute the IE. Also $v_{c1}[k] = 1$
    else if vertex k is type 2 **then** compute the IE for each barycentric triangle. Also, $v_{c1}[k] = 0.5$
    else merge each couple of barycentric triangles surrounding vertex k into quadrilaterals and compute the IE. Also $v_{c1}[k] = n_{t_{ik}}^{-1}$
    for each barycentric vertex , l, in the triangle j **do**
      if vertex l is type 1 **then** merge each couple of barycentric triangles surrounding vertex l into triangles and compute the IE . Also $v_{c2}[l] = 1$
      else if vertex l is type 2 **then** compute the IE for each barycentric triangle. Also, $v_{c2}[l] = 0.5$
      else merge each couple of barycentric triangles surrounding vertex l into quadrilaterals and compute the IE. Also $v_{c2}[l] = n_{t_{jl}}^{-1}$
      • compute cross interactions from the integration elements coming from vertex k and vertex l and save them in an array, $v_{ci2}[l]$.
    end for
    • calculate $v_{ci1}[k]$ as the dot product $v_{ci2} \cdot v_{c2}$
  end for
  • calculate $\mathbf{W}_h^{\widehat{P}_1,\widehat{P}_1}[i][j]$ as the dot product $v_{ci1} \cdot v_{c1}$
end for
output: $\mathbf{W}_h^{\widehat{P}_1,\widehat{P}_1}$

As for the weakly singular mCMP, the preconditioning strategy (2.18) becomes

$$\mathbf{G}_h^{-1}\mathbf{W}_{\kappa,h}^{\widehat{P}_1,\widehat{P}_1}\mathbf{G}_h^{-T}\mathbf{V}_{\kappa,h}^{P_0,P_0}\mathbf{u} = \mathbf{G}_h^{-1}\mathbf{W}_{\kappa,h}^{\widehat{P}_1,\widehat{P}_1}\mathbf{G}_h^{-T}\mathbf{b}, \tag{3.7}$$

where $\mathbf{W}_{\kappa,h}^{\widehat{P}_1,\widehat{P}_1}$ is the Galerkin $(n_T \times n_T)$-matrix for the hypersingular operator built over the dual mesh and $\mathbf{V}_{\kappa,h}^{P_0,P_0}$ is the discrete version of the weakly singular operator over the primal mesh. An auxiliary matrix of size $(n_T + n_V + n_E) \times (n_T + n_V + n_E)$ is used to build $\mathbf{W}_{\kappa,h}^{\widehat{P}_1,\widehat{P}_1}$ in order to avoid computing barycentric basis functions that conform the dual basis functions more than once, as these last ones have overlapping supports. Then, although assembly time is reduced, storage not necessarily does. Still, this could

be improved by using only an auxiliary matrix of size $(n_V + n_E) \times (n_V + n_E)$ to store barycentric interactions that are part of more than one dual basis function.

By Theorem 0.2, we can derive a similar bound to (3.5):

$$\kappa_2(\mathbf{G}_h^{-1}\mathbf{W}_{\kappa,h}^{\widehat{P}_1,\widehat{P}_1}\mathbf{G}_h^{-T}\mathbf{V}_{\kappa,h}^{P_0,P_0}) \leq K\kappa_2(\mathbf{G}_h^{-1}\mathbf{W}_{\kappa,h}^{\widehat{P}_1,\widehat{P}_1}(\boldsymbol{\Sigma}\mathbf{W}_{\kappa,h}^{\overline{P}_1,\overline{P}_1}\boldsymbol{\Sigma}^T)^{-1}\mathbf{G}_h), \tag{3.8}$$

where κ_2 is the spectral condition number of a matrix and K is a bound for $\kappa_2(\mathbf{P}_h\mathbf{W}_{\kappa,h})$. Again, the boundedness of the condition number depends on the right-hand side factor, which shall remain close to one as $\mathbf{W}_{\kappa,h}^{\widehat{P}_1,\widehat{P}_1} \approx \boldsymbol{\Sigma}\mathbf{W}_{\kappa,h}^{\overline{P}_1,\overline{P}_1}\boldsymbol{\Sigma}^T$.

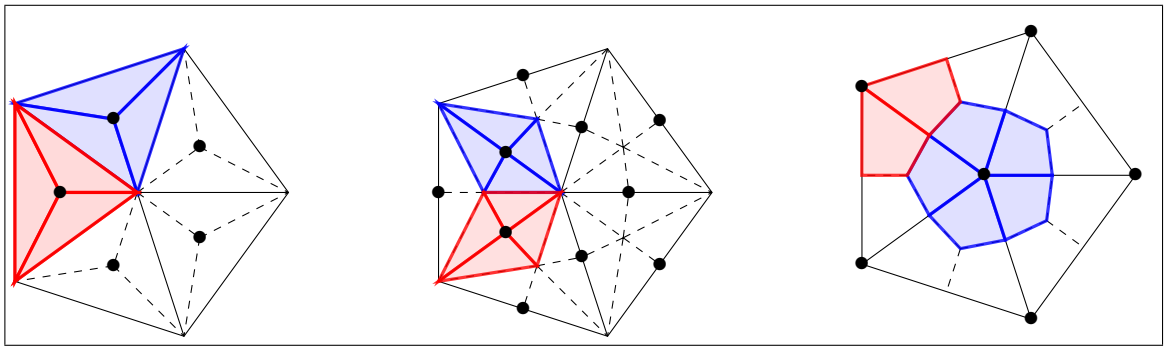

Figure 3.3. $\widehat{P}_1$ basis functions supports. Black dots represent dofs of type 1, 2 and 3 respectively. For the barycentric type 1 dofs, supports are given by three triangles conformed by the union of two barycentric triangles; barycentric type 2 dofs are built by four barycentric triangles; and, type 3 dofs by quadrilaterals made from two barycentric triangles.

3.3. Hierarchical Modified Calderón Preconditioner ($\mathcal{H}$-mCMP)

Based on work by Escapil-Inchauspé and Jerez-Hanckes Escapil-Inchauspé and Jerez-Hanckes (2019), we further improve our mCMP by using $\mathcal{H}$-mat approximations, denoting these versions $\mathcal{H}$-mCMP. For the sake of brevity, we skip definitions that can be found in Hackbusch (2015).

Far-field interactions are defined by block cluster trees, decompositions based on index trees $T(I)$ and $T(J)$, and assembled by means of low rank approximations of the original matrix, with subsequent savings in space and assembly time. Following Bebendorf (2008), a matrix $\mathbf{A} \in C_k^{m\times n}$ is said to be of low rank if $k(m+n) \ll m \cdot n$. Moreover, its representation is given by

$$\mathbf{A} = \sum_{i=1}^{k} u_i v_i^*, \quad u_i \in \mathbb{C}^m, \quad v_i \in \mathbb{C}^n,$$

where k is the rank of the approximation and $u_i v_i^*$ is an outer product. Ideally, the low rank approximation could be derived by a singular value decomposition (SVD), but this is prohibitively expensive Hackbusch (2015). Alternatively, we use the Adaptive Cross Approximation (ACA) which defines the rank of the approximation in an adaptive way, given an accuracy ε. According to Bebendorf, Maday, and Stamm (2014), the approximation is constructed incrementally by Algorithm 3, wherein `col` and `row` are routines that yield the j_q-th column and i_q-th row respectively of a submatrix of $\mathbf{A}$. Therefore, the algorithm computes a sequence of vectors u_q and v_q such that

$$\tilde{\mathbf{A}} = \sum_{q=1}^{k} \frac{u_q v_q^T}{(v_q)_{j_q}}.$$

In principle, the algorithm stops when $\left\|\tilde{\mathbf{A}} - \mathbf{A}\right\|_2 < \varepsilon$, which is equivalent Bebendorf et al. (2014) to say

$$\frac{\|u_{q+1}\|_2 \, \|v_{q+1}\|_2}{|v_{q+1}|} < \varepsilon.$$

For our $\mathcal{H}$-mCMPs, the ACA is computed directly on the dual mesh and so $\left\|\tilde{\mathbf{A}} - \mathbf{A}\right\|_2 \leq \varepsilon$ for each far-field block. Accordingly, by Theorem 0.3, it can be deduced that

$$\kappa_2(\mathbf{P}^{\mathcal{H}}(\mathbf{W}_{\kappa,h}/\mathbf{V}_{\kappa,h})) \leq \kappa_2(\mathbf{P}_h(\mathbf{W}_{\kappa,h}/\mathbf{V}_{\kappa,h}))\frac{1+\varepsilon\alpha}{1-\varepsilon\alpha}, \tag{3.9}$$

Algorithm 3 Low rank approximation calculation.

1: **input:** ε, κ, $\mathcal{T}_h$(grid)
2: set $q := 1$
3: **while** err$< \varepsilon$ **do**
4: Calculate $\mathbf{A}_{i_q,:} = \texttt{row}(i_q, \kappa, \mathcal{T}_h)$, choosing i_q such that

$$v_q := \mathbf{A}^T_{i_q,:} - \sum_{l=1}^{q-1} \frac{(u_l)_{i_q}}{(v_l)_{j_l}} v_l$$

is non-zero and compute $\mathbf{A}_{:,j_q} = \texttt{col}(j_q, \kappa, \mathcal{T}_h)$, for a j_q such that $\left|(v_q)_{j_q}\right| = \max_{j=1\ldots N} \left|(v_q)_j\right|$

5: Compute the vector $u_q := \mathbf{A}_{:,j_q} - \sum_{l=1}^{q-1} \frac{(v_l)_{j_q}}{(v_l)_{j_l}} u_l$
6: Compute the error indicator
err $= \left|(v_q)_{j_q}\right|^{-1} \|v_q\|_2 \|u_q\|_2$
7: **end while**

where $\alpha := \max\{1, C_{sp}\text{depth}(T)\} 1/\sigma_{\min}\kappa_2(\mathbf{G}_h)$, $\sigma_{\min}$ is the smallest singular value of $\mathbf{V}_{\kappa,h}$ or $\mathbf{W}_{\kappa,h}$, and depth(T) stands for depth of the tree T

$$\text{depth}(T) = \text{depth}(T(I \times J)) = \min\{\text{depth}(T(I)), \text{depth}(T(J))\}.$$

Consequently, the lower the tolerance used for ACA, the better condition number of the preconditioned system. Furthermore, for larger matrices, a lower tolerance should be used in order to keep the system well conditioned as factors depth(T) and $1/\sigma_{min}$ grow –the last one, due to the preconditioner singular values ($\mathbf{V}_{\kappa,h}$ or $\mathbf{W}_{\kappa,h}$), which tend to zero as the mesh shrinks. $\kappa_2(\mathbf{G}_h)$ should not affect the factor α as it is bounded (Sauter & Schwab, 2010, Remark 4.5.3). Considering a geometrically balanced clustering to construct $T(I)$, $T(J)$, the stopping criterion: size$(b) \leq m$, $m =$ leafsize with b in P a partition of T, will determine the length of the deepest branch of the tree and also depth$(T(I))$, depth$(T(J))$ for a level conserving block cluster tree (Hackbusch, 2015, Theorem 5.27). In other words, the clustering algorithm will require more levels to achieve a smaller leafsize and thus, bigger leafsizes should lead to a better conditioned system.

4. CHAPTER 5: METHODOLOGY

4.1. Software

To test the effectiveness of the mCMPs and $\mathcal{H}$-mCMPs for the BIEs resulting from exterior Helmholtz problems with Dirichlet and Neumann boundary conditions, respectively, we solve different cases for the unit sphere, the Fichera cube and a submarine-like shaped object.

Given the original method to handle numerical integration on the dual mesh, test cases for the mCMP and $\mathcal{H}$-mCMP were programed in C, for which the library GSL was used to store and perform matrix operations. This program emulates the BEMPP structure, which means that it returns PYTHON objects, such as NUMPY arrays or CTYPES, in the case of the $\mathcal{H}$-mat. Routines `matvec` and `matmat` were implemented to apply matrix-vector and matrix-matrix products between $\mathcal{H}$-matrices stored in CTYPES and PYTHON vectors or matrices. In order to validate the code, results were compared with those obtained with the library BEMPP (Betcke, Arridge, Phillips, Smigaj, & Schweiger, 2015). This means that we programmed the standard version of the CMP that will be called *Standard* and compared it with the standadrs version of the CMP fabricated with BEMPP that will be also called *Bempp*. Discrete operators were computed with numerical quadratures of order one for $\mathbf{V}_{\kappa,h}$ –order three for its singular part– and of order two for $\mathbf{W}_{\kappa,h}$ –order three for its singular part–. Tests were executed on a 32 core, 4 GB RAM per core, 64-bit Linux server.

A geometrically balanced clustering algorithm was used to create $\mathcal{H}$-mat along with ACA. The preconditioned BIO Galerkin matrix and right-hand sides for each case were computed using BEMPP and solved with restarted `GMRES(m)` routine of SCIPY, where the preconditioned version of the `GMRES(m)` with preconditioners defined as `LinearOperator` objects of SCIPY was implemented. Finally, for each mesh, $\mathcal{H}$-mat were calculated using different tolerance values ε each signaled by $\mathcal{H}$-mCMPε.

4.2. Validation and early results

In order to validate the code, a comparison was made between results coming from the use of Calderón preconditioners built in BEMPP and the preconditioners coming from the library developed for this work. To build the preconditioners, numerical quadratures of order one and order three for the singular part of the matrices were used in the case of $\hat{P}_0$ basis functions, and of order two and order three for the singular part of the matrices in the case of $\hat{P}_1$ basis functions.

For V_κ mCMP and V_κ $\mathcal{H}$-M$^2\varepsilon$CP, spherical meshes were created in GMSH, by using different refinement factors: $n = 4, 6, 8, 10, 12$ points per wavelength, with a constant wavenumber of $\kappa = 5$, resulting in systems of 181, 425, 653, 1,069 and 1,442 degrees of freedom (dofs). $\mathcal{H}$-mCMPε were calculated with a leafsize of 10, resulting in trees of depth 5, 6, 7, 8 and 9 respectively.

For a vibrating sphere of radius a, with a normal surface given by $\hat{v}_n(a, \theta, \phi, \omega)$, the Neumann boundary condition is given by $\gamma_N u = i\omega\rho_0 \hat{v}_n(a, \theta, \phi, \omega)$, where ρ_0 is the ambient density of the medium surrounding the sphere and ω is the angular frequency with $\kappa = \omega c^{-1}$, with c the speed of sound. General solutions for this problem are of the form (Wu, 2015):

$$u(r, \theta, \phi, \omega) = i\omega\rho_0 \sum_{n=0}^{\infty} \frac{h_n^{(1)}(\kappa r)}{\frac{dh_n^{(1)}(\kappa r)}{dr}|_{r=a}} \sum_{m=-n}^{n} Y_n^m(\cos\theta) \int_0^{2\pi} d\phi' \int_0^{\pi} \hat{v}_n(a, \theta', \phi', \omega) \\ Y_n^{m*}(\theta', \phi') \sin\theta' d\theta', \tag{4.1}$$

where $h_n^{(1)}(\kappa r)$ denote first kind spherical Hankel functions and $Y_n^m(\theta, \phi)$ are Legendre polynomials. For simplicity, we considered a constant Neumann boundary condition, $\gamma_N u = (3i - 5)\kappa$, for which $\hat{v}_n(a, \theta, \phi, \omega) = \frac{(3i-5)\kappa}{\omega\rho_0} = v_0$ is constant. For $n = m = 0$ in (4.1), the solution of the exterior problem is given by

$$u(r, \theta, \phi, \omega) = \rho_0 c v_0 \left(\frac{\kappa a}{\kappa a + i}\right) \left(\frac{a}{r}\right) e^{i\kappa(r-a)} \text{ for } a = 1.$$

The solution for this particular problem is

$$u(r,\theta,\phi,\omega) = (3i-5)\left(\frac{\kappa}{\kappa+i}\right)\left(\frac{1}{r}\right)e^{i\kappa(r-1)},$$

which was used to compute the $H^{\frac{1}{2}}$-relative error between numerical and analytical solutions.

The $H^{\frac{1}{2}}$-norm was calculated as (Langer & Steinbach, 2007)

$$\|u\|_{H^{\frac{1}{2}}(\Gamma)} = \sqrt{\sum_n \bar{u}_n \langle \phi_n, \mathsf{W}_0 u \rangle}$$

Table 4.1 shows the total number of iterations for GMRES(20) for unpreconditioned and preconditioned systems with different versions of the CMP. Results show that the preconditioner built using BEMPP, the standard preconditioner built for this work and the mCMP developed for this work perform similarly, thereby validating our code. Also, results in Table 4.1 for the hierarchical versions of the mCMP support the idea that a lower tolerance in the ACA algorithm will result in a more effective method, which coincides with the behavior described in (3.9). Table 4.2 shows $H^{\frac{1}{2}}$- and L^2-relative errors for all the methods considered, yielding same convergence rates, thereby validating our code.

Table 4.1. GMRES iteration comparison for the unpreconditioned system and the system preconditioned with different versions of the CMP, for the Neumann problem over a unit sphere of 1,442 dofs, and $\kappa = 5$.

Method	None	mCMP	BEMPP	Standard	$\mathcal{H}$-mCMP01E-6	$\mathcal{H}$-mCMP1E-5	$\mathcal{H}$-mCMP5E-5	$\mathcal{H}$-mCMP1E-4
Iterations	66	17	18	18	17	20	36	57

Table 4.2. $H^{\frac{1}{2}}$- and L^2-relative errors for different CMP methods applied to a Neumann problem 2.17 over a unit sphere meshed with triangles of different characteristic lengths (h).

h	L^2-relative error	$H^{\frac{1}{2}}$-relative error
3.14E-01	5.12E-02	5.12E-02
2.09E-01	2.25E-02	2.25E-02
1.57E-01	1.61E-02	1.56E-02
1.26E-01	1.21E-02	1.11E-02
1.05E-01	1.17E-02	9.88E-03

Finally, results in Figure 4.1 evidences that, for increasing numbers of dofs, smaller tolerances in the ACA algorithm are required in order to keep the effectiveness of the method. This can is explained by (3.9) as an increase factor of $\alpha = \max\{1, C_{sp}\text{depth}(T)\}/\sigma_{min}\kappa_2(\mathbf{G}_h)$ would require ε to decrease in order to keep the boundedness of $\kappa_2(\mathbf{P}^{\mathcal{H}}\mathbf{W}_{\kappa,h})$, as increasing the number of dofs also increases depth(T) and $1/\sigma_{min}$. It can also be seen that although the ratio $\varepsilon/\sigma_{min} > 1$, the preconditioner is still effective, which does not necessarily contradict (3.9).

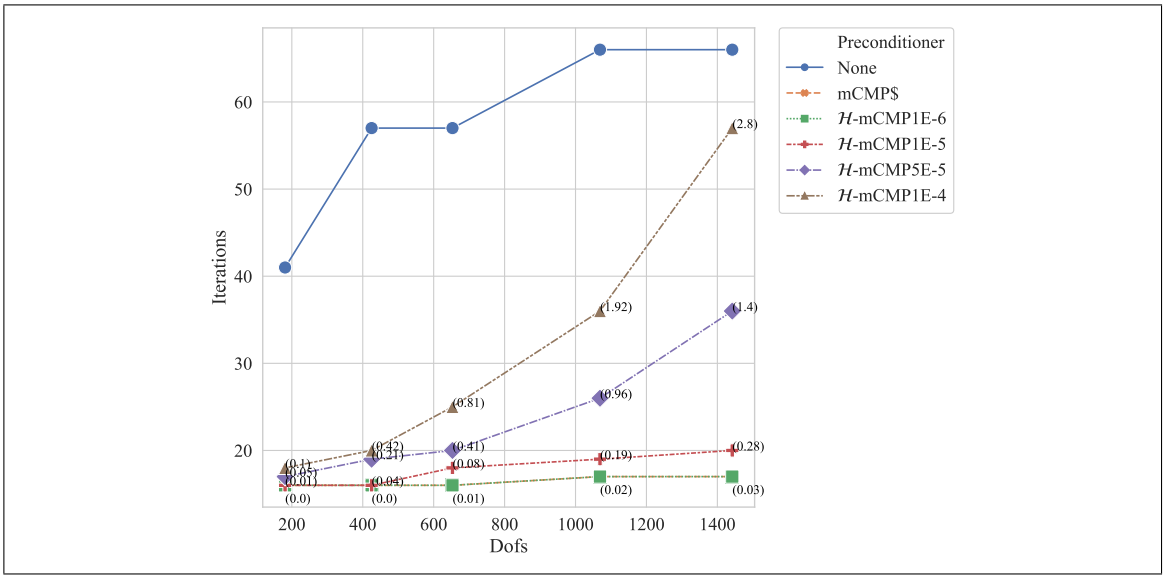

Figure 4.1. Number of GMRES iterations to solve the Neumann problem over a sphere by unpreconditioned and preconditioned systems with a family of $\mathcal{H}$-mCMP preconditioners. Numbers in parenthesis represent the ratio ε/σ_{min}

To validate the W_κ mCMP and W_κ $\mathcal{H}$-mCMPε, again, spherical meshes were created in BEMPP, with the same sequence of refinement factors: $n = 4, 6, 8, 10$ points per wavelength, with a constant wavenumber of $\kappa = 5$. The $\mathcal{H}$-mCMPε were calculated with a leafsize of 10, resulting in trees of depth 8, 9, 10 and 11 respectively.

For a vibrating sphere of radius a, with a superficial acoustic pressure on $r = a$ given by $u(a, \theta, \phi, \omega)$, we have a Dirichlet boundary condition given by $\gamma_D u = u(a, \theta, \phi, \omega)$. The general solution for this problem is (Wu, 2015):

$$u(r, \theta, \phi, \omega) = i\omega\rho_0 \sum_{n=0}^{\infty} \frac{h_n^{(1)}(\kappa r)}{h_n^{(1)}(\kappa r)|_{r=a}} \sum_{m=-n}^{n} Y_n^m(\cos\theta) \int_0^{2\pi} d\phi' \int_0^{\pi} \gamma_D u Y_n^{m*}(\theta', \phi') \sin\theta' d\theta' \tag{4.2}$$

In this case, we considered a constant Dirichlet boundary condition, $\gamma_D u = (3i - 5)\kappa$. Consequently, with $n = m = 0$ in (4.2), the solution for the exterior problem on the unit

sphere is

$$u(r,\theta,\phi,\omega) = \gamma_D u\left(\frac{a}{r}\right)e^{i\kappa(r-a)} \quad \text{for } a = 1.$$

with

$$\frac{du(r,\theta,\phi,\omega)}{dr} = \gamma_D u\left(i\kappa\left(\frac{a}{r}\right)e^{i\kappa(r-a)} - \left(\frac{a}{r^2}\right)e^{i\kappa(r-a)}\right) \quad \text{for } a = 1.$$

which was used to compute the $H^{\frac{1}{2}}$-relative error between numerical and analytical solutions. The $H^{-\frac{1}{2}}$-norm was calculated as (Langer & Steinbach, 2007):

$$\|u\|_{H^{-\frac{1}{2}}(\Gamma)} = \sqrt{\sum_n \bar{u}_n \langle \phi_n, \mathsf{V}_0 u\rangle}$$

Table 4.3 shows the number of GMRES iterations to solve for a sphere of 2,128 dofs. Results prove the effectiveness of the $\mathcal{H}$-mCMP and its hierarchical versions and as for the Neumann case, for a lower tolerance used in the ACA algorithm, the method performed better.

Table 4.3. Number of iterations of GMRES(20) solving the Dirichlet problem over a sphere of 2128 dofs with different versions of the $\mathcal{H}$-mCMP.

Method	None	BEMPP	Standard	mCMP	$\mathcal{H}$-mCMP1E-3	$\mathcal{H}$-mCMP1E-2	$\mathcal{H}$-mCMP1E-1
Iterations	150	20	30	20	40	60	80

Figure 4.2 shows that the effectiveness of $\mathcal{H}$-mCMPε decreases with the increment of dofs for a fixed ε and that the mCMP is not affected by smaller mesh sizes. Again, the preconditioners with higher values of ε are more sensitive to the reduction of the mesh size, but are more robust than the preconditioners built for the Neumann case, even if $\varepsilon/\sigma_{min} > 1$.

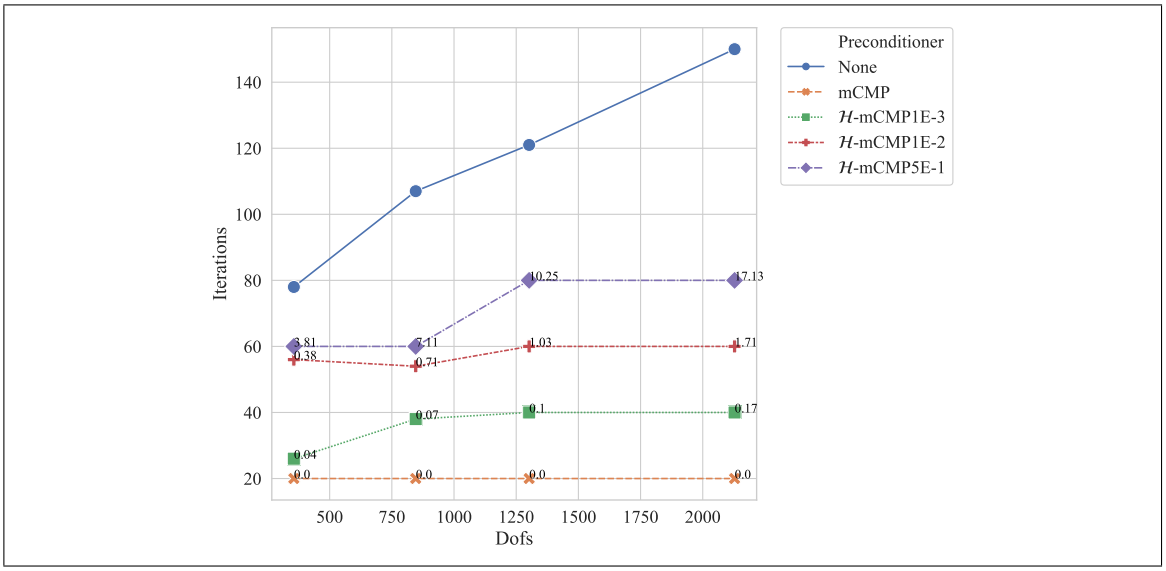

Figure 4.2. Evolution of the number of iterations for the Dirichlet problem with an unpreconditioned and precondtioned system with a family of $\mathcal{H}$-mCMP. Numbers in parenthesis represent the ratio ε/σ_{min} .

Table 4.4. $H^{-\frac{1}{2}}$ and L^2 relative error evolution for different CMP methods applied to a Dirichlet problem over a unit sphere meshed with triangles of different characteristic lengths (h).

h	L^2 -relative error	$H^{-\frac{1}{2}}$ -relative error
3.14.E-01	1.73E-02	3.75E-03
2.09.E-01	1.21E-02	1.93E-03
1.57.E-01	7.78E-03	1.20E-03
1.26.E-01	6.94E-03	7.97E-04

Table 4.4 shows the evolution of the $H^{-\frac{1}{2}}$ and L^2-relative error for a fixed $\kappa = 5$, with different mesh refinements h. This fact also validates the effectiveness of the code, as the preconditioners developed for this work converge the same as the preconditioner developed by BEMPP.

5. CHAPTER 6: RESULTS

5.1. Results for the V_κ as mCMP

5.1.0.1. Fichera cube

To prove the effectiveness of the mCMP for domains with corners, similar tests were carried out on a Fichera cube. A surface mesh of 3,278 nodes was used along with a different boundary condition: $\gamma_N u = \hat{n}_z e^{i\kappa z}$, with $\kappa = 15$. The $\mathcal{H}$-mCMPε were calculated with a leafsize of 100, resulting in a cluster tree of depth 10.

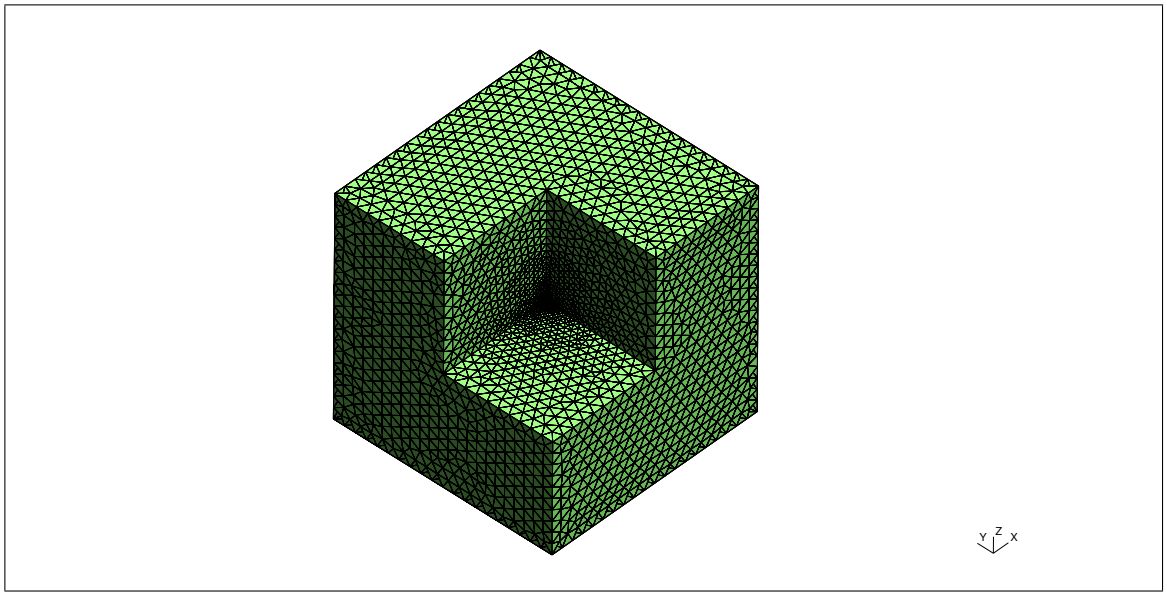

Figure 5.1. Fichera cube of length one, with 3,278 dofs. The mesh was obtained by using GMSH with a uniform triangulation refined at the reentrant corner.

Table 5.1 confirms that the preconditioner quality decreases when the ε parameter of the ACA algorithm is increased. In this case, after a tolerance of $\varepsilon = 5$E-5, the preconditioner is no longer effective, as expected from (3.9).

Table 5.1. Number of GMRES iterations for the resolution of a system preconditioned with a family of $\mathcal{H}$-mCMP for a Neumann problem on a Fichera cube.

None	mCMP	Bempp	Standard	$\mathcal{H}$-mCMP1E-7	$\mathcal{H}$-mCMP1E-6	$\mathcal{H}$-mCMP1E-5	$\mathcal{H}$-mCMP5E-5
940	300	300	300	300	380	300	1160

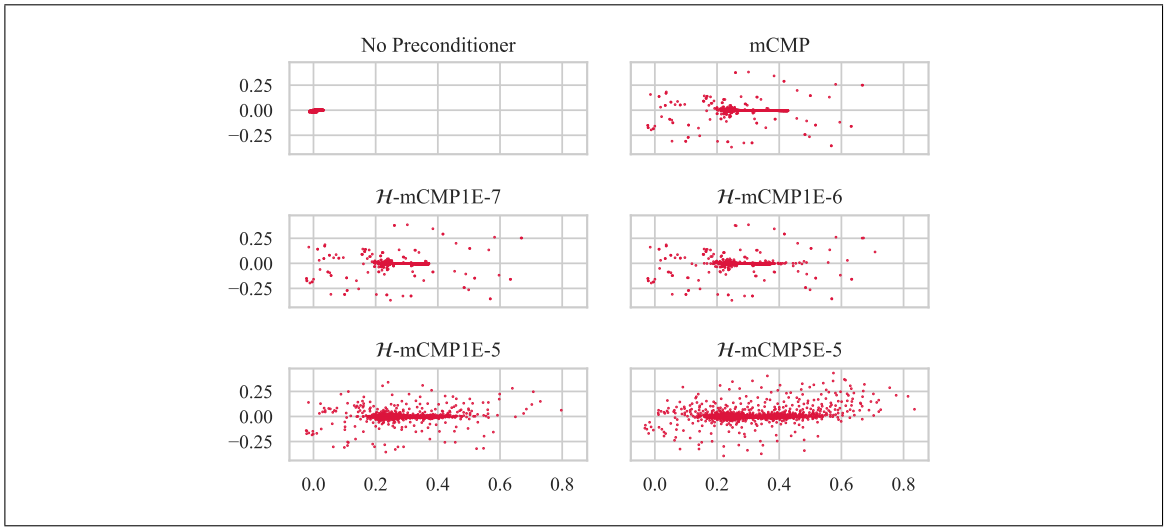

Figure 5.2. Spectral behavior comparison between the unpreconditioned system and a family of $\mathcal{H}$-mCMP for the Neumann problem on a Fichera cube.

Figure 5.2 shows the spectrum of each preconditioned system. The eigenvalues of the unpreconditioned system cluster near zero and move away when preconditioned to a larger value. Notice that the spectrum spreads when increasing ε, which explains why the preconditioner performance worsens. Also, Figure 5.3 shows that, for this particular example, the $\mathcal{H}$-mat versions of the mCMP have a total assembly time slightly higher than the unpreconditioned system and that the assembly time of the mCMP is significantly better than the BEMPP standard Calderón. It can also be noticed that although shorter, the

timing of the BEMPP Calderón is close to the magnitude of the standard Calderón developed for this work, which supports the efficiency of the modified preconditioner.

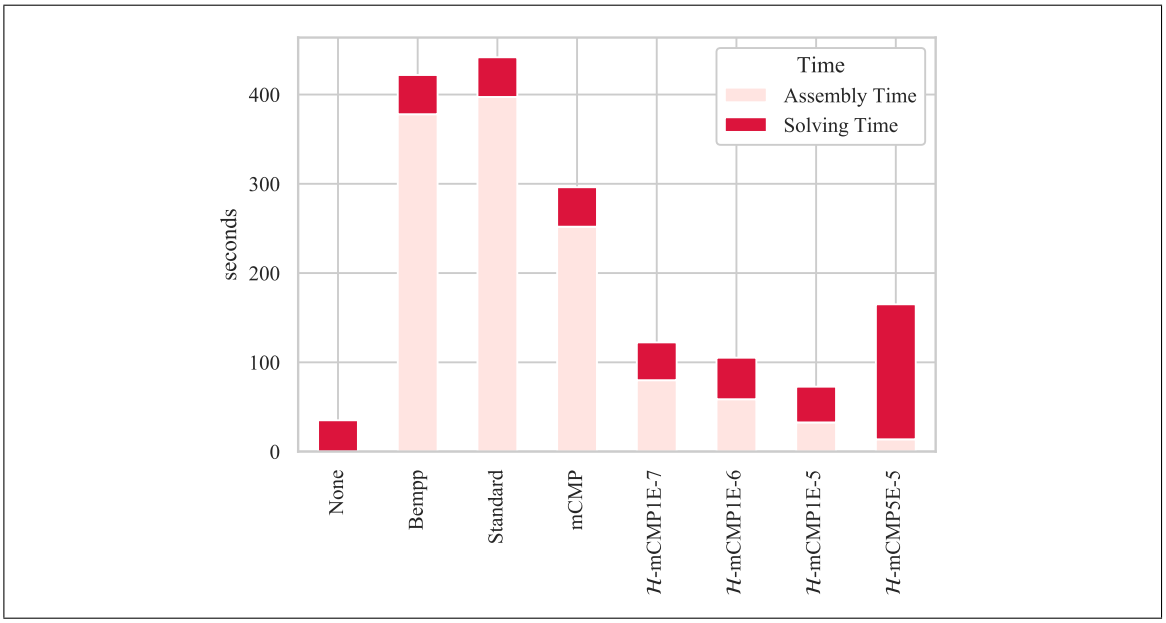

Figure 5.3. Assembly and solving times for preconditioned and unpreconditioned systems for the Neumann problem on a Fichera cube.

5.1.1. Results for the submarine-like shape object

A submarine-like object was built from a sphere, a cone and a wedge, whose mesh contains 3,470 nodes and 6,936 triangles. The boundary condition used for this case was $\gamma_N u = \hat{n}_z e^{i\kappa z}$, with $\kappa = 8$ and the $\mathcal{H}$-mCMPε were calculated with a leafsize of 50.

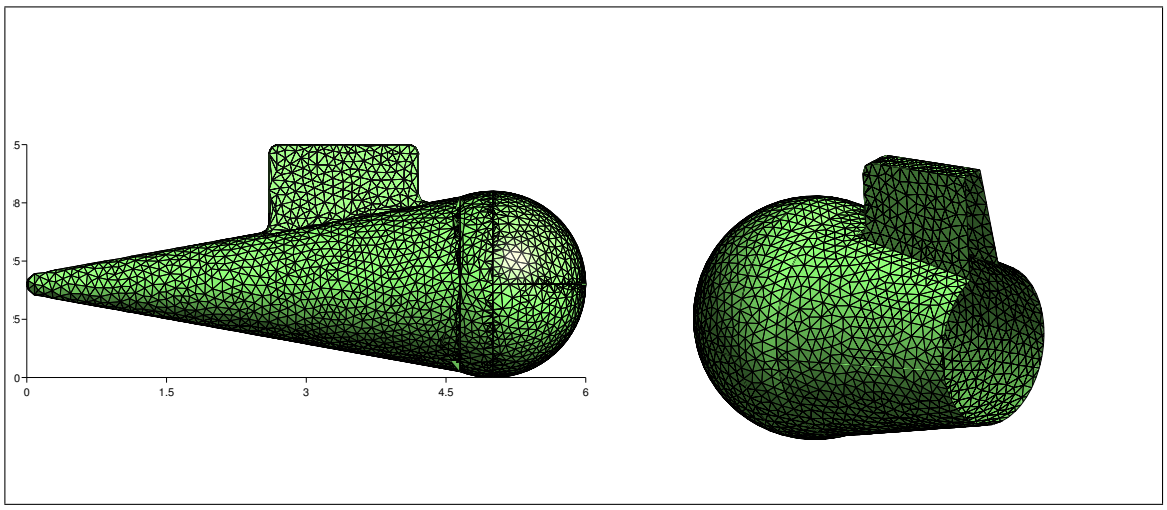

Figure 5.4. Latera and interior view of the submarine mesh of length 6 metres, height 2.5 metres and width 1 metre. The mesh has 3,740 vertices.

Table 5.2 shows that, in this case, the accuracy used in the ACA does not affect the number of iterations needed to solve the preconditioned system. This is confirmed by the eigenvalue distributions displayed in Figure 5.5, as all preconditioned systems have a similar distribution.

Table 5.2. GMRES iteration comparison between the unpreconditioned system and a family of $\mathcal{H}$-mCMP for the Neuman problem, with $\kappa = 8$, over a submarine grid.

None	mCMP	Bempp	Standard	$\mathcal{H}$-mCMP1E-6	$\mathcal{H}$-mCMP1E-5	$\mathcal{H}$-mCMP5E-5	$\mathcal{H}$-mCMP1E-4
7,399	840	900	920	820	860	680	740

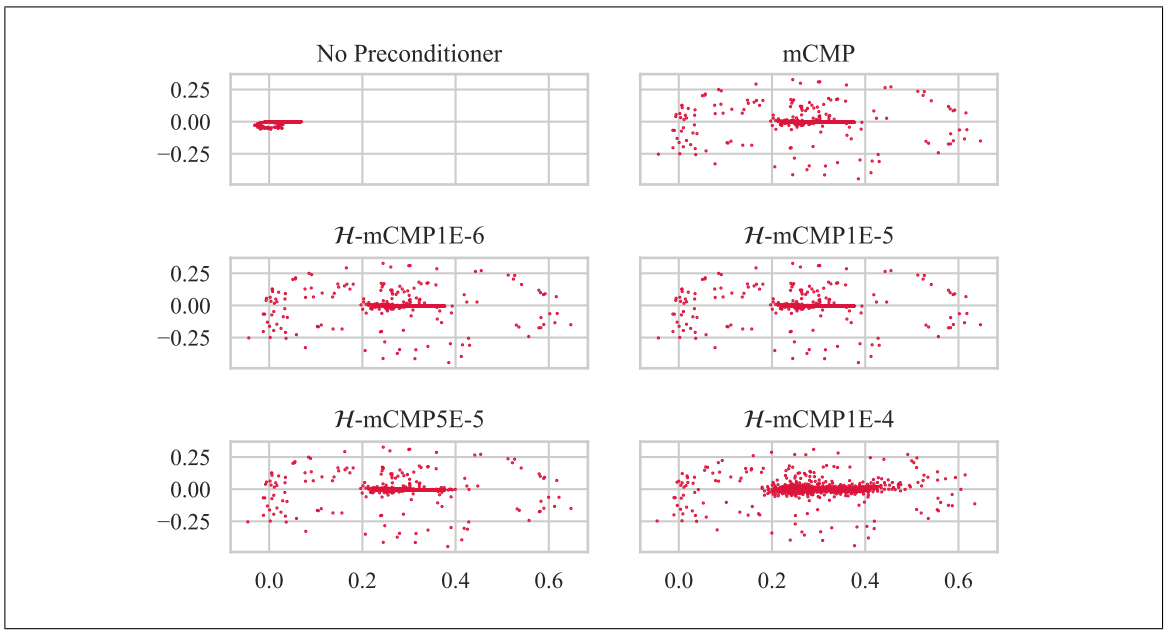

Figure 5.5. Spectral behavior comparison between the unpreconditioned system and a family of mCMP for the Neumann problem over a submarine grid.

Assembly and solving times for each preconditioner (Figure 5.6) confirm the standard CMP poor performance. Total execution time of the mCMP does not exceed that of the unpreconditioned system, however, the $\mathcal{H}$-mat versions perform better.

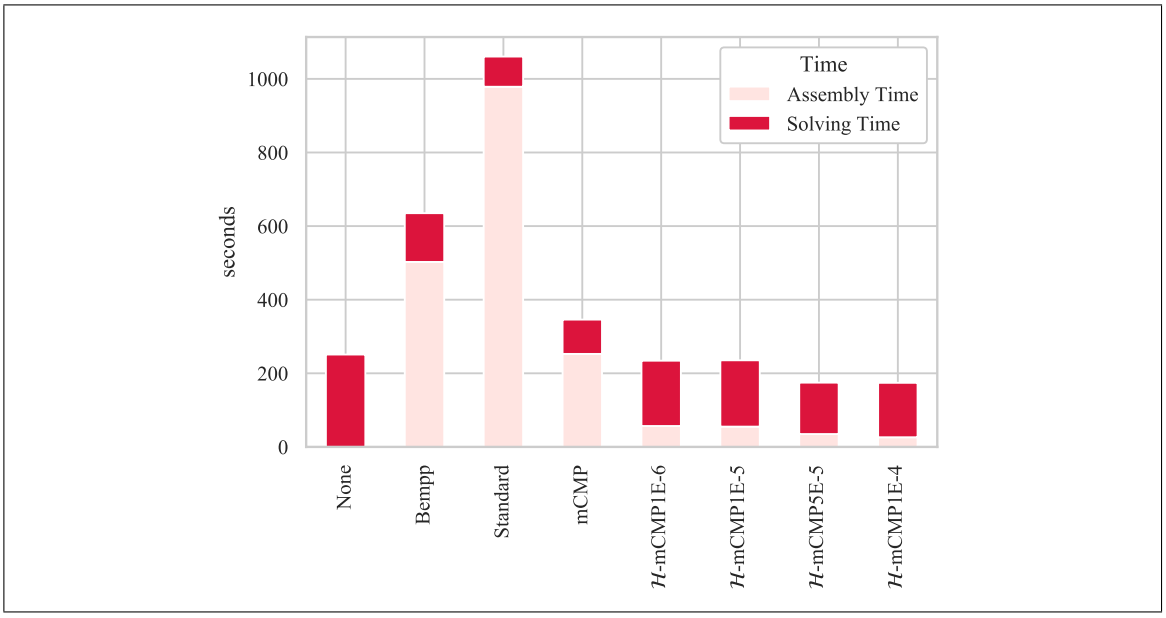

Figure 5.6. Assembly and solving time for preconditioned and unpreconditioned systems over the submarine grid.

Further analysis of the spectrum for the $\mathcal{H}$-mCMP shows that the resulting condition number has a behavior respect to ε similar to the one stated in (3.9). Figure 5.7 show the evolution of the condition number with ε for $\mathcal{H}$-mat built with different leafsizes. It can be seen that the increase of the leafsize results in a decreasing factor α from equation (3.9), as the depth(T) factor and the number of far-field blocks decrease. Thus, the preconditioner becomes more robust in terms of the ε parameter. Despite of this, larger lefsizes lead to higher assembly times as Table 5.3 shows.

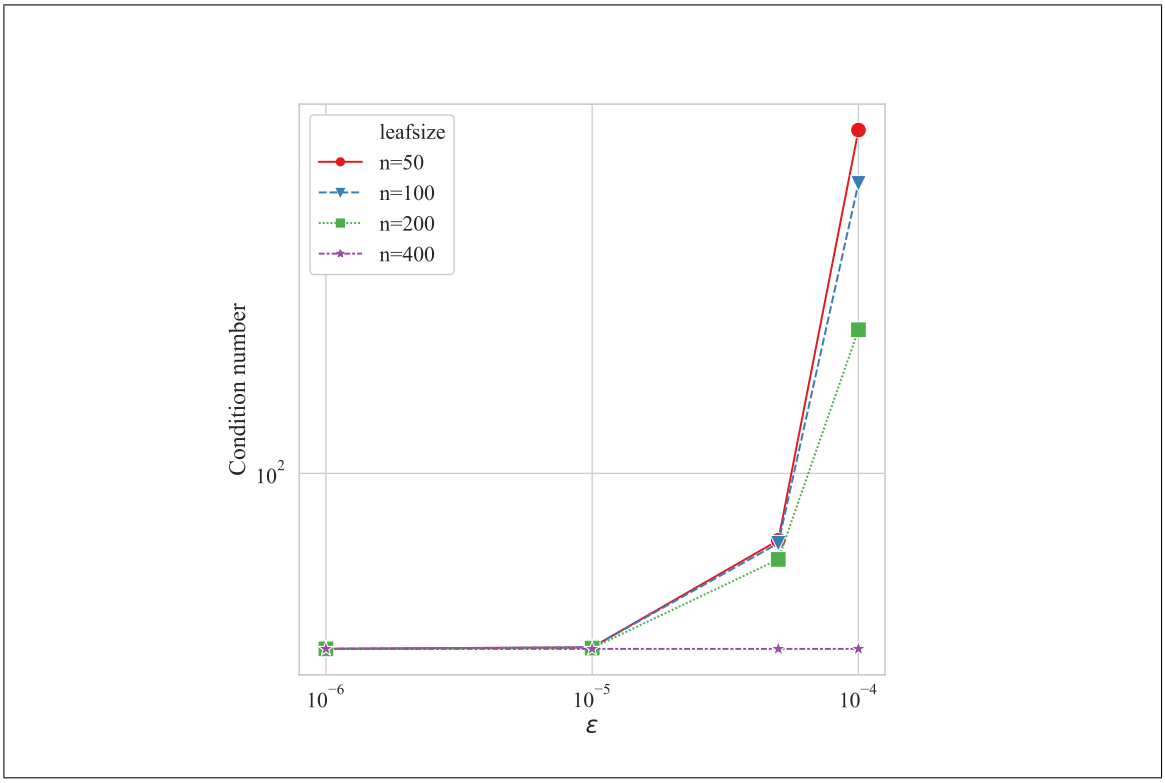

Figure 5.7. Condition number's evolution for systems over the submarine grid, preconditioned with $\mathcal{H}$-mCMP built with different leafsizes for the Neumann problem over a submarine like shaped object. Condition number grows at a rate similar to the stated in (3.9), remaining constant for $n = 400$.

Table 5.3. Assembly times for families of $\mathcal{H}$-mCMP built with different leafsize values for the Neumann problem over a submarine like shaped object. Larger leafsize values take more time to be assembled.

Leafsize	1E-6	1E-5	5E-5	1E-4
50	36.74	43.96	26.36	18.69
100	49.07	36.18	56.16	36.21
200	65.11	128.09	50.21	92.94
400	152.73	307.74	140.01	137.80

5.2. Results for the W_κ as mCMP

5.2.1. Results for the Fichera cube

In this case, a surface mesh of 5,284 triangles was used along with the boundary condition: $\gamma_D u = \hat{n}_z e^{i\kappa z}$, with $\kappa = 7$ and the $\mathcal{H}$-mCMPε were calculated with a leafsize of 50.

Figure 5.4 shows that modified preconditioners built with lower tolerances do not have necessarily a better performance. Figure 5.8 shows the spectral properties of each preconditioned system for a family of mCMP, where the spectrum of the unpreconditioned system is clustered near zero and moves away from this point by the preconditioner. For larger values of ε in the ACA approximation applied in the construction of the $\mathcal{H}$-mCMP the eigenvalues spread a bit, but remain clustered near a point far from zero, which explains the behavior exhibited in 5.4.

Table 5.4. GMRES comparison between the unpreconditioned system and a family of mCMP for the Dirichlet problem, with $\kappa = 7$ over s Fichera cube.

None	mCMP	Bempp	Standard	$\mathcal{H}$-mCMP1E-5	$\mathcal{H}$-mCMP1E-4	$\mathcal{H}$-mCMP5E-3	$\mathcal{H}$-mCMP1E-2
5,881	120	700	40	120	60	80	80

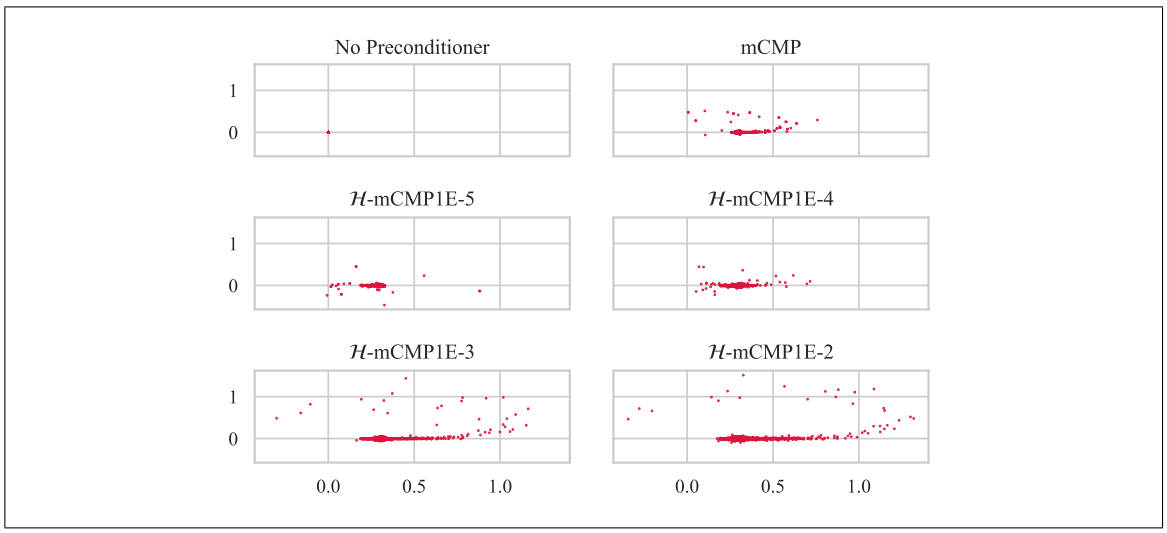

Figure 5.8. Spectral behavior comparison between the unpreconditioned system and a family of mCMP for the Dirichlet problem, with $\kappa = 7$ over s Fichera cube.

Figure 5.9 shows a comparison of assembly and solving times between the unpreconditioned and preconditioned systems with a family of $\mathcal{H}$-mCMP and mCMP. Results show that the use of the mCMP by itself is not enough to get an efficient preconditioner, but one of its hierarchical versions does take less time to be assembled and solved than the unpreconditioned method.

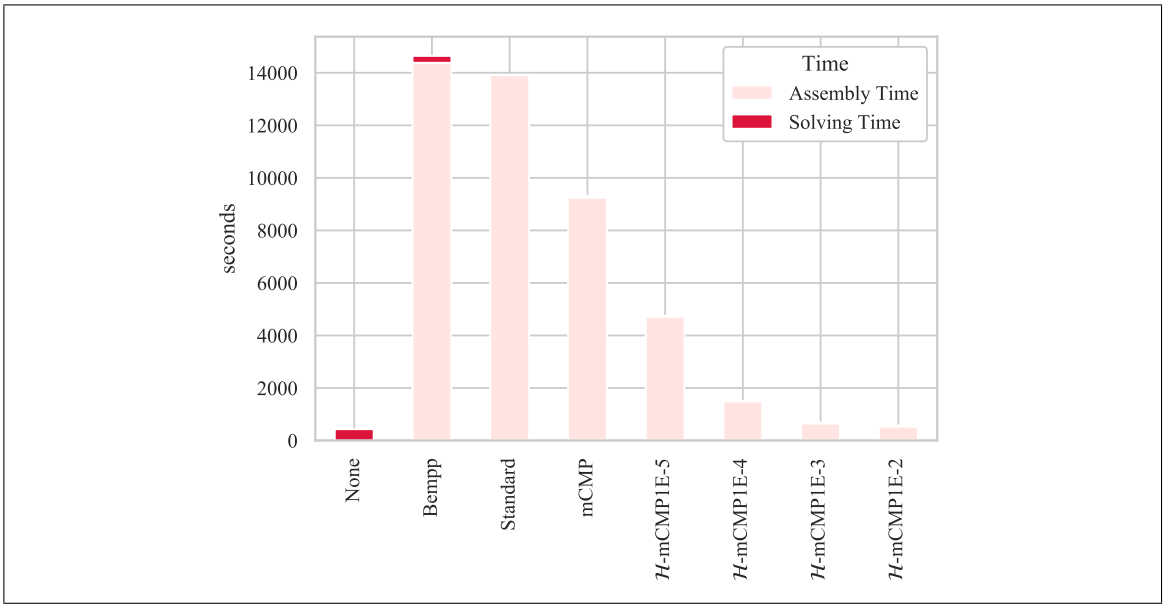

Figure 5.9. Assembly and solving time for preconditioned and unpreconditioned systems over the Fichera cube for the Dirirchlet problem. Solving times for the preconditioned systems vary between 11 and 60 seconds.

It can also be seen in 5.9, that the standard preconditioner takes almost half more time than mCMP and that the BEMPP preconditioner takes, in this case, more time than the standard CMP built for this work.

5.2.1.1. Submarine-like shape object

The boundary condition $\gamma_D u = \hat{n}_z e^{i\kappa z}$, with $\kappa = 8$ and leafsize of 50 were used in this case.

The GMRES(20) iteration counts in Table 5.5 show that the residuals do not converge in case that no preconditioner is applied, but do converge when a $\mathcal{H}$-mCMP or the mCMP is used. The diagram also shows that some of the hierarchical versions of the mCMP could work even better than the mCMP itself, which stands in favor of the robustness of the method, at least for the Dirichlet case. Again, the behavior of the residuals is coincident

with the spectral behavior of the preconditioned systems (figure 5.10), whose eigenvalues moved away from zero.

Table 5.5. GMRES iteration comparison between the unpreconditioned system and a family of mCMP for the Dirichlet problem, with $\kappa = 8$, over a submarine grid.

None	mCMP	$\mathcal{H}$-mCMP1E-5	$\mathcal{H}$-mCMP1E-4	$\mathcal{H}$-mCMP5E-3	$\mathcal{H}$-mCMP1E-2
>20,000	700	760	780	220	340

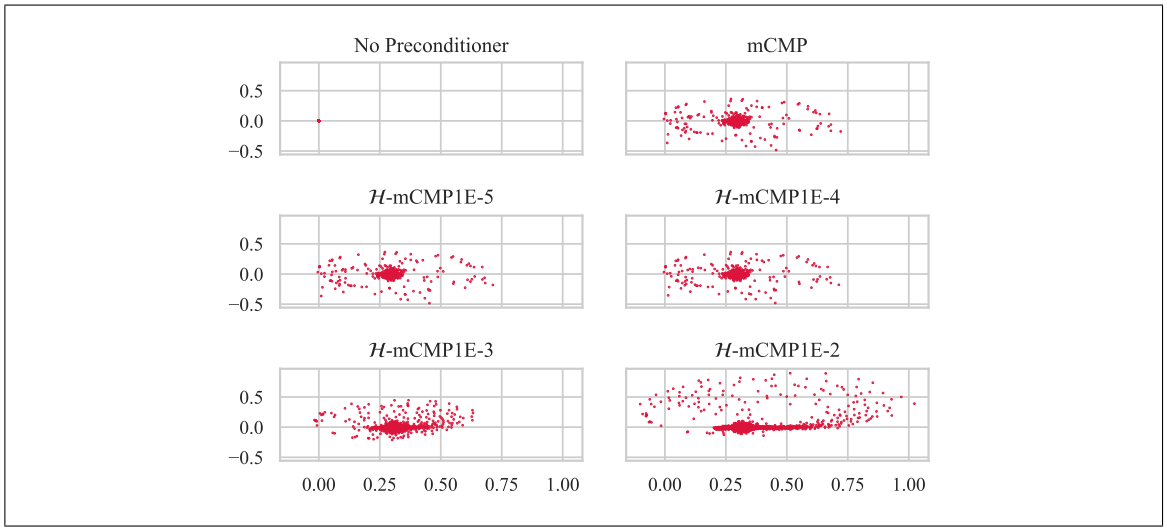

Figure 5.10. Spectral behavior comparison between the unpreconditioned system and a family of $\mathcal{H}$-mCMP for the Dirichlet problem, with $\kappa = 8$, over a submarine grid.

In this case, results only for the mCMP and its hierarchical versions are available, because of the great amount of dofs in the barycentric mesh so we only show results for the mCMP. As before, Figure 5.11 shows that the mCMP is not efficient enough for the problem, but $\mathcal{H}$-mCMP1E-2 does overcome the total timing of the unpreconditioned system,

which is the result we were expecting.

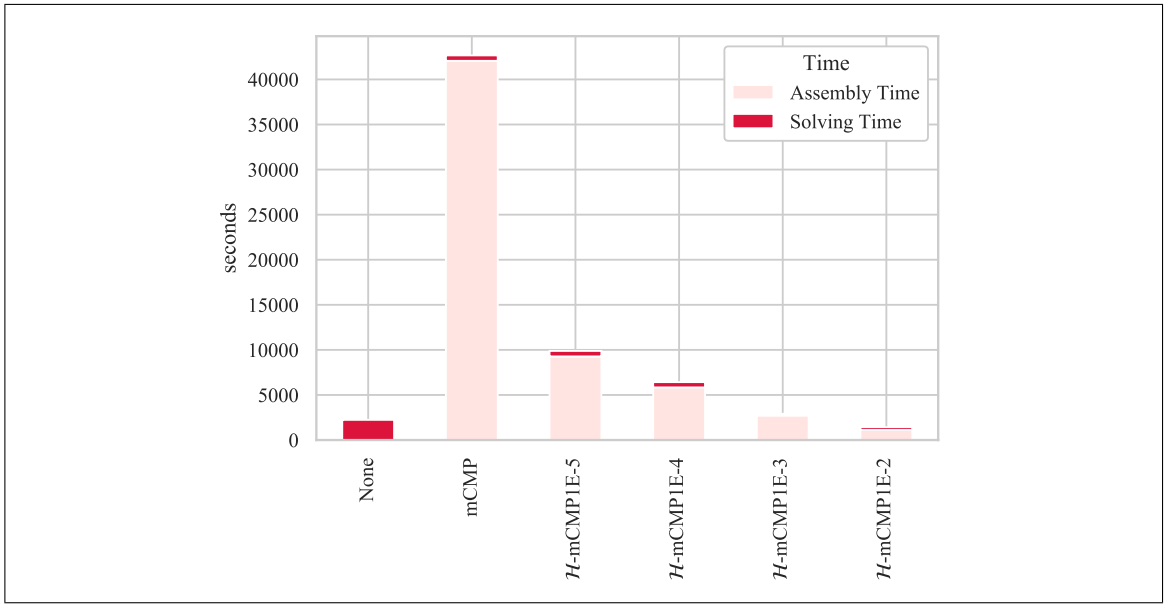

Figure 5.11. Assembly and solving time for preconditioned and unpreconditioned systems for the Dirichlet problem over the submarine grid (6,936 dofs).

6. CONCLUSIONS

The $\mathcal{H}$-mCMPs presented in this research provide significant reduction in computational time with respect to the original Calderón preconditioner. The proposed mCMP respects the spectral properties of the original CMP. Moreover, we have shown our technique to be more efficient than the CMPs currently used to precondition the Helmholtz BIOs. Robustness for larger leafsizes is also shown though this results in longer matrix assembly times. Also, the understanding of the influence of the ε parameter in the preconditioner assembly strategy turns out to be crucial to keep the efficiency of the method. Finally, though there still are cases in which Calderón preconditioning is not more effective than the unpreconditioned system, in general, the mCMPs are suitable for cases in which the unpreconditioned system takes a long time to be solved and thus, its applicability and improvements should remain as a subject of interest.

REFERENCES

Adams, R. J., & Champagne, N. J. (2004). A numerical implementation of a modified form of the electric field integral equation. *IEEE Transactions on Antennas and Propagation*, *52*(9), 2262–2266.

Adrian, S. B., Andriulli, F. P., & Eibert, T. F. (2018). On a refinement-free Calderón multiplicative preconditioner for the electric field integral equation. *arXiv preprint arXiv:1803.08333*.

Andriulli, F. P., Cools, K., Bagci, H., Olyslager, F., Buffa, A., Christiansen, S., & Michielssen, E. (2008). A multiplicative Calderón preconditioner for the electric field integral equation. *IEEE Transactions on Antennas and Propagation*, *56*(8), 2398–2412.

Bebendorf, M. (2008). *Hierarchical matrices*. Springer.

Bebendorf, M., Maday, Y., & Stamm, B. (2014). Comparison of some reduced representation approximations. In *Reduced order methods for modeling and computational reduction* (pp. 67–100). Springer.

Betcke, T., Arridge, S., Phillips, J., Smigaj, W., & Schweiger, M. (2015). Solving boundary integral problems with bem++. *ACM Transactions on Mathematical Software*, *41*(6), 1–40.

Börm, S., Grasedyck, L., & Hackbusch, W. (2003). Introduction to hierarchical matrices with applications. *Engineering analysis with boundary elements*, *27*(5), 405–422.

Buffa, A., & Christiansen, S. (2007). A dual finite element complex on the barycentric refinement. *Mathematics of Computation*, *76*(260), 1743–1769.

Chen, K. (2005). *Matrix preconditioning techniques and applications* (Vol. 19). Cambridge University Press.

Chen, Y. P., Sun, S., Jiang, L., & Chew, W. C. (2014). A Calderón preconditioner for the electric field integral equation with layered medium green's function. *IEEE Transactions on Antennas and Propagation*, *62*(4), 2022–2030.

Christiansen, S. H., & Nédélec, J.-C. (2002). A preconditioner for the electric field integral

equation based on Calderón formulas. *SIAM journal on numerical analysis*, *40*(3), 1100–1135.

Colton, D., & Kress, R. (2012). *Inverse acoustic and electromagnetic scattering theory* (Vol. 93). Springer Science & Business Media.

Cools, K., Andriulli, F. P., & Michielssen, E. (2011). A Calderón multiplicative preconditioner for the pmchwt integral equation. *IEEE transactions on Antennas and Propagation*, *59*(12), 4579.

Dobbelaere, D., De Zutter, D., Van Hese, J., Sercu, J., Boonen, T., & Rogier, H. (2015). A Calderón multiplicative preconditioner for the electromagnetic Poincaré-Steklov operator of a heterogeneous domain with scattering applications. *Journal of Computational Physics*, *303*, 355–371.

Escapil-Inchauspé, P., & Jerez-Hanckes, C. (2019). Fast Calderón preconditioning for the electric field integral equation. *IEEE Transactions on Antennas and Propagation*.

Faustmann, M., Melenk, J., & Praetorius, D. (2017). Existence of H-matrix approximants to the inverse of BEM matrices: the hyper-singular integral operator. *IMA Journal of Numerical Analysis*, *37*(3), 1211–1244.

Gossye, M., Huynen, M., Ginste, D. V., De Zutter, D., & Rogier, H. (2018). A Calderón preconditioner for high dielectric contrast media. *IEEE Transactions on Antennas and Propagation*, *66*(2), 808–818.

Hackbusch, W. (2015). *Hierarchical matrices: algorithms and analysis* (Vol. 49). Springer.

Hiptmair, R. (2006). Operator preconditioning. *Computers and mathematics with Applications*, *52*(5), 699–706.

Hiptmair, R., Jerez-Hanckes, C., & Urzúa-Torres, C. (2013). *Optimal operator preconditioning for boundary elements on open curves* (Vol. 2013; Tech. Rep. No. 2013-48). Seminar for Applied Mathematics, ETH Zürich.

Hiptmair, R., Jerez-Hanckes, C., & Urzúa-Torres, C. (2017). *Optimal operator preconditioning for weakly singular operator over 3d screens* (Vol. 2017; Tech. Rep. No. 2017-13). Seminar for Applied Mathematics, ETH Zürich.

Hiptmair, R., & Urzúa-Torres, C. (2016). *Dual mesh operator preconditioning on 3D screens: Low-order boundary element discretization* (Vol. 2016; Tech. Rep. No. 2016-14). Seminar for Applied Mathematics, ETH Zürich.

Langer, U., & Steinbach, O. (2007). Coupled finite and boundary element domain decomposition methods. In *Boundary element analysis* (pp. 61–95). Springer.

Lee, J.-F., Lee, R., & Burkholder, R. J. (2003). Loop star basis functions and a robust preconditioner for EFIE scattering problems. *IEEE Transactions on Antennas and Propagation*, *51*(8), 1855–1863.

McLean, W. C. H. (2000). *Strongly elliptic systems and boundary integral equations*. Cambridge university press.

Meurant, G., & Tebbens, J. D. (2015). The role eigenvalues play in forming gmres residual norms with non-normal matrices. *Numerical Algorithms*, *68*(1), 143–165.

Saad, Y. (2003). *Iterative methods for sparse linear systems* (Vol. 82). siam.

Saad, Y., & Schultz, M. H. (1986). Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on scientific and statistical computing*, *7*(3), 856–869.

Sauter, S. A., & Schwab, C. (2010). *Boundary element methods*. Springer.

Steinbach, O., & Wendland, W. L. (1998). The construction of some efficient preconditioners in the boundary element method. *Advances in Computational Mathematics*, *9*(1-2), 191–216.

Stephanson, M. B., & Lee, J.-F. (2009). Preconditioned electric field integral equation using Calderón identities and dual loop/star basis functions. *IEEE Transactions on Antennas and Propagation*, *57*(4), 1274–1279.

Wu, S. F. (2015). *The Helmholtz equation least-squares method*. Springer.

Yan, S., Jin, J.-M., & Nie, Z. (2010). EFIE analysis of low-frequency problems with loop-star decomposition and Calderón multiplicative preconditioner. *IEEE Transactions on Antennas and Propagation*, *58*(3), 857–867.

APPENDIX

A. THEORETICAL RESULTS

For the sake of completeness recall (Hiptmair, 2006, Theorem 2.1)

Theorem 0.1. *On two Hilbert spaces $V,\ W$ we consider continuous sesquilinear forms $a \in \mathcal{L}(X \times X)$ and $b \in \mathcal{L}(Y \times Y)$ and $t \in \mathcal{L}(X \times Y)$, with norms $\|a\|$, $\|b\|$ and $\|t\|$.*

***(C1)** If a, b and t satisfy a discrete inf-sup condition with constants c_A, c_B, $c_T > 0$, respectively, on their corresponding discrete spaces, and*

***(C2)** $\dim X_h = M = dim Y_h$, choosing any discrete bases $\{\varphi_j\}_{j=1}^M$ of X_h and $\{\phi_i\}_{i=1}^M$ of Y_h. Then, the associated Galerkin matrices:*

$$\mathbf{A_h} := (a(\varphi_i, \varphi_j))_{i,j=1}^M, \mathbf{B_h} := (b(\phi_i, \phi_j))_{i,j=1}^M, \mathbf{T_h} := (t(\varphi_i, \phi_j))_{i,j=1}^M$$

satisfy

$$\kappa_2(\mathbf{T_h}^{-1}\mathbf{B_h}\mathbf{T_h}^{-H}\mathbf{A_h}) \leq \frac{\|a\|\ \|b\|\ \|t\|}{c_A c_B c_T^2} \tag{A.1}$$

where κ_2 designates the spectral condition number.

The next result yields an error estimate for the hierarchical Calderón preconditioner.

Lemma 0.1. *Let $\mathbf{A}_h$ be a discretization of V_κ or W_κ. Then, the hierarchical version of $\mathbf{A}_h$, denoted $\mathbf{A}^{\mathcal{H}}$ converges linearly with ε to $\mathbf{A}_h$ in spectral norm.*

PROOF. Though this result was already demonstrated in Faustmann, Melenk, and Praetorius (2017), for the sake of completeness we repeat the proof. We start by recalling Lemma 6.32 in Hackbusch (2015). Let $M \in R^{N\times N}, \mathcal{P}$ be a partitioning of $\mathcal{I} \times \mathcal{I}$. Then,

$$\|M\|_2 \leq \max\{1, C_{sp}\text{depth}(T)\} \max\{\|M_{\tau\times\sigma}\|_2\},$$

where C_{sp} is the sparsity constant defined in Hackbusch (2015):

$$C_{sp}(X) = \max\{\max_{\tau\in T(I)} |\{\sigma \in T(J) : \tau \times \sigma \in X\}|, \max_{\sigma\in T(J)} |\{\tau \in T(I) : \tau \times \sigma \in X\}|\}$$

and $\text{depth}(T) = \text{depth}(T(I \times J)) = \min\{\text{depth}(T)(I), \text{depth}(T)(J)\}$. Applying this bound to $\mathbf{A}_h - \mathbf{A}^{\mathcal{H}}$ leads to

$$\left\|\mathbf{A}_h - \mathbf{A}^{\mathcal{H}}\right\|_2 \leq \max\{1, C_{sp}\text{depth}(T)\} \max\{\left\|(\mathbf{A}_h - \mathbf{A}^{\mathcal{H}})|_{\tau\times\sigma}\right\|_2\}.$$

Given the definition of full matrices and the accuracy of ACA:

$$\left\|(\mathbf{A}_h - \mathbf{A}^{\mathcal{H}})|_{\tau\times\sigma}\right\|_2 = \begin{cases} 0 & \text{for full matrices} \\ \varepsilon & \text{for low-rank matrices} \end{cases}$$

Hence,

$$\left\|\mathbf{A}_h - \mathbf{A}^{\mathcal{H}}\right\|_2 \leq \varepsilon \max\{1, C_{sp}\text{depth}((T))\}.$$

as stated. □

Based on the previous lemma, we can bound the condition number for the Modified Calderón preconditioner of either the weakly or hyper-singular operators.

Theorem 0.2. *Given a pair of BIOs,* A *and* B *under assumptions like in 0.1, along with their suitable Galerkin discretizations* $\mathbf{A}_h$ *and* $\mathbf{B}_h$ *over primal and dual meshes, respectively. Let* $\mathbf{P}'_h$ *be a dual preconditioner (*$\mathbf{P}'_h\mathbf{A}_h = \mathbf{G}_h^{-1}\mathbf{B}'_h\mathbf{G}_h^{-T}\mathbf{A}_h$*) assembled via the method described in Section 3.2. The condition number of* $\mathbf{P}'_h\mathbf{A}_h$ *is bounded by:*

$$\kappa_2(\mathbf{P}'_h\mathbf{A}_h) \leq K\kappa_2(\mathbf{G}_h^{-1}\mathbf{B}'_h\mathbf{B}_h^{-1}\mathbf{G}_h) \tag{A.2}$$

where K *is a bound for* $\kappa_2(\mathbf{P}_h\mathbf{A}_h)$ *given by 0.1.*

PROOF. Given the triangular discretizations of a pair of BIOs, A and B, ($\mathbf{A}_h$ and $\mathbf{B}_h$ respectively) and a quadrilateral discretization for A ($\mathbf{A}'_h$), a bound for the condition number of $\mathbf{P}'_h\mathbf{A}_h$ can be found, by noticing that

$$\|\mathbf{P}'_h\mathbf{A}_h\|_2 = \left\|\mathbf{P}'_h(\mathbf{P}_h^{-1}\mathbf{P}_h)\mathbf{A}_h\right\|_2 \leq \left\|\mathbf{P}'_h\mathbf{P}_h^{-1}\right\|_2 \|\mathbf{P}_h\mathbf{A}_h\|_2$$
$$\left\|(\mathbf{P}'_h\mathbf{A}_h)^{-1}\right\|_2 = \left\|(\mathbf{P}'_h(\mathbf{P}_h^{-1}\mathbf{P}_h)\mathbf{A}_h)^{-1}\right\|_2 \leq \left\|(\mathbf{P}'_h\mathbf{P}_h^{-1})^{-1}\right\|_2 \left\|(\mathbf{P}_h\mathbf{A}_h)^{-1}\right\|_2$$

and then

$$\kappa_2(\mathbf{P}'_h\mathbf{A}_h) \leq \left\|\mathbf{P}'_h\mathbf{P}_h^{-1}\right\|_2 \|\mathbf{P}_h\mathbf{A}_h\|_2 \left\|(\mathbf{P}'_h\mathbf{P}_h^{-1})^{-1}\right\|_2 \left\|(\mathbf{P}_h\mathbf{A}_h)^{-1}\right\|_2 = \kappa_2(\mathbf{P}_h\mathbf{A}_h)\kappa_2(\mathbf{P}'_h\mathbf{P}_h^{-1}).$$

Finally, considering the equality $\mathbf{P}'_h\mathbf{P}_h^{-1} = \mathbf{G}_h^{-1}\mathbf{B}'_h\mathbf{B}_h^{-1}\mathbf{G}_h$

$$\kappa_2(\mathbf{P}'_h\mathbf{A}_h) \leq \kappa_2(\mathbf{P}_h\mathbf{A}_h)\kappa_2(\mathbf{P}'_h\mathbf{P}_h^{-1}) \leq K\kappa_2(\mathbf{G}_h^{-1}\mathbf{B}'_h\mathbf{B}_h^{-1}\mathbf{G}_h) \tag{A.3}$$

as stated. □

Finally, we obtain the following bound for the condition number of the system preconditioned by the $\mathcal{H}$-mCMP

Theorem 0.3. *Given a pair of BIOs,* A *and* B*, under assumptions like in 0.1, along with their suitable Galerkin discretizations* $\mathbf{A}_h$ *and* $\mathbf{B}_h$ *over primal and dual meshes, respectively. Let* $\mathbf{P}_h$ *be a dual preconditioner (*$\mathbf{P}_h\mathbf{B}_h = \mathbf{G}_h^{-T}\mathbf{A}_h\mathbf{G}_h^{-1}\mathbf{B}_h$*).*

If $\varepsilon \max\{1, C_{sp}depth(T)\}\sigma_{\max}\kappa_2(\mathbf{G}_h) < 1$*, then the condition number of* $\kappa_2(\mathbf{P}^{\mathcal{H}}\mathbf{B}_h)$ *can be controled by* ε *and is bounded by:*

$$\kappa_2(\mathbf{P}_h\mathbf{B}_h)\frac{1+\varepsilon \max\{1, C_{sp}depth(T)\}/\sigma_{\min}\kappa_2(\mathbf{G}_h)}{1-\varepsilon \max\{1, C_{sp}depth(T)\}/\sigma_{\min}\kappa_2(\mathbf{G}_h)} \tag{A.4}$$

PROOF. By definition, it holds

$$\begin{aligned}\kappa_2(\mathbf{P}^{\mathcal{H}}\mathbf{B}_h) &= \left\|\mathbf{P}^{\mathcal{H}}\mathbf{B}_h\right\|_2 \left\|(\mathbf{P}^{\mathcal{H}}\mathbf{B}_h)^{-1}\right\|_2 \\ &= \left\|\mathbf{P}^{\mathcal{H}}\mathbf{B}_h(\mathbf{P}_h\mathbf{B}_h)^{-1}(\mathbf{P}_h\mathbf{B}_h)\right\|_2 \left\|(\mathbf{P}_h\mathbf{B}_h)^{-1}(\mathbf{P}_h\mathbf{B}_h)(\mathbf{P}^{\mathcal{H}}\mathbf{B}_h)^{-1}\right\|_2 \\ &\leq \left\|\mathbf{P}^{\mathcal{H}}\mathbf{B}_h(\mathbf{P}_h\mathbf{B}_h)^{-1}\right\|_2 \|\mathbf{P}_h\mathbf{B}_h\|_2 \left\|(\mathbf{P}_h\mathbf{B}_h)^{-1}\right\|_2 \left\|(\mathbf{P}_h\mathbf{B}_h)(\mathbf{P}^{\mathcal{H}}\mathbf{B}_h)^{-1}\right\|_2 \\ &= \kappa_2(\mathbf{P}_h\mathbf{B}_h) \left\|\mathbf{P}^{\mathcal{H}}\mathbf{P}_h^{-1}\right\|_2 \left\|(\mathbf{P}^{\mathcal{H}}\mathbf{P}_h^{-1})^{-1}\right\|_2 \end{aligned} \tag{A.5}$$

Since $\delta(\mathbf{P}^{\mathcal{H}}) = \mathbf{P}^{\mathcal{H}} - \mathbf{P}_h$, then :

$$\begin{aligned} &= \kappa_2(\mathbf{P}_h\mathbf{B}_h) \left\|(\mathbf{P}^{\mathcal{H}} - \mathbf{P}_h + \mathbf{P}_h)\mathbf{P}_h^{-1}\right\|_2 \left\|((\mathbf{P}^{\mathcal{H}} - \mathbf{P}_h + \mathbf{P}_h)\mathbf{P}_h^{-1})^{-1}\right\|_2 \\ &= \kappa_2(\mathbf{P}_h\mathbf{B}_h) \left\|(\delta(\mathbf{P}^{\mathcal{H}})\mathbf{P}_h^{-1} + \mathbf{I})\right\|_2 \left\|(\delta(\mathbf{P}^{\mathcal{H}})\mathbf{P}_h^{-1} + \mathbf{I})^{-1}\right\|_2 . \end{aligned}$$

Observe that

$$\delta(\mathbf{P}^{\mathcal{H}})\mathbf{P}_h^{-1} = (\mathbf{G}_h^{-1}\delta(\mathbf{A}^{\mathcal{H}})\mathbf{G}_h^{-T})(\mathbf{G}_h^{-1}\mathbf{A}_h\mathbf{G}_h^{-T})^{-1} = \mathbf{G}_h^{-1}\delta(\mathbf{A}^{\mathcal{H}})\mathbf{A}_h^{-1}\mathbf{G}_h. \tag{A.6}$$

Using the triangular inequality and calculating the inverse matrix by the Neumann series, under the assumption that $\varepsilon \max\{1, C_{sp}\text{depth}(T)\}/\sigma_{\min}\kappa_2(\mathbf{G}_h) < 1$, we can write

$$\kappa_2(\mathbf{P}^{\mathcal{H}}\mathbf{B}_h) \leq \kappa_2(\mathbf{P}_h\mathbf{B}_h)\frac{1 + \varepsilon \max\{1, C_{sp}\text{depth}(T)\}/\sigma_{\min}\kappa_2(\mathbf{G}_h)}{1 - \varepsilon \max\{1, C_{sp}\text{depth}(T)\}/\sigma_{\min}\kappa_2(\mathbf{G}_h)} \tag{A.7}$$

where $\sigma_{\min}$ is the smallest singular value of A. □

B. COMPLEMENTARY CONTENT

B.1. Krylov Subspaces, a short insight

Given a linear system:

$$\mathbf{A}\mathbf{x} = \mathbf{b},$$

where $\mathbf{A}$ is an invertible, complex-valued matrix of $n \times n$ entries and $\mathbf{b} \in \mathbb{C}^n$, methods for solving this problem fall in two different classes: direct methods and iterative methods.

On one hand, direct methods find in a certain number of iterations, with a computational time complexity of $O(n^3)$ steps and then, when linear systems are large and dense, direct methods require a prohibitive computational time to find a solution. Thus, iterative schemes are considered as an alternative to direct methods. Among these, stationary relaxation-type methods as Jacobi and Gauss-Seidel methods can be found, but have the disadvantage of beeing well suited for certain classes of matrices only. To solve this problem, projection methods are used, because they have more general use and are more robust.

A general projection method for solving the system $\mathbf{Ax} = \mathbf{b}$ is a method which seeks an approximate solution x_m from an affine subspace $x_0 + \mathcal{K}_m$ of dimension m, were x_0 is an arbitrary initial guess to the solution. The approximate solution x_m also has to fullfill the Galerkin condition $\mathbf{b} - \mathbf{Ax}_m \perp \mathcal{L}_m$, where $\mathcal{L}_m$ is also a space of dimension m and a Krylov subspace method is such that the subspace

$$\mathcal{K}_m(\mathbf{A}, r_0) = \text{span}\{r_0, \mathbf{A}r_0, \mathbf{A}^2 r_0, \ldots, \mathbf{A}^{m-1} r_0\}, \tag{B.1}$$

where $r_0 = \mathbf{b} - \mathbf{Ax}_0$ (Saad, 2003).

From another perspective, a Krylov subspace $\mathcal{K}_m$ is the subspace of all vectors in $\mathbb{R}^n$ which can be written as $x = p(\mathbf{A})v$ where p is a polynomial of degree not exceeding $m - 1$.

One of the most important applications of the Krylov subspaces is Arnoldi's method, which is an orthogonal projection method onto $\mathcal{K}_m$ for general non-hermitian matrices. This leads to a way for finding approximations for the eigenvalues of such matrices. The algorithm reads:

Algorithm 4 Arnoldi

1: Choose a vector v_1 such that $\| v_1 \|_2 = 1$
2: **for** $j = 1, \ldots, m$ **do**
3: Compute $h_{ij} = (\mathbf{A}v_j, v_i)$ for $i = 1, \ldots, j$
4: Compute $w_j = \mathbf{A}v_j - \sum_{i=1}^{j} h_{ij} v_i$
5: $h_{j+1,j} = \| w_j \|_2$
6: If $h_{j+1,j} = 0$ then stop
7: $v_{j+1} = \frac{w_j}{h_{j+1,j}}$
8: **end for**

Arnoldi's method has several versions depending on the method used to orthonormalize the basis of v_i vectors, but the idea is essencially the same, which is to form an orthonormal basis fo the $\mathcal{K}_m$ subspace.

At the same time, Arnoldi's method is used by several algorithms to solve linear systems. The Full Orthogonalization Method (FOM) and its variations are supported by it, but the one that concerns to this investigation is the Generalized Minimum Residual Method (GMRES), which is a projection method based on taking the Krylov subspace, $\mathcal{K} = \mathcal{K}_m$ and $\mathcal{L} = \mathbf{A}\mathcal{K}_m$, where $\mathcal{K}_m$ is the mth Krylov subspace. The GMRES algorithm reads:

Algorithm 5 GMRES

1: Compute $r_0 = b - \mathbf{A}x_0$, $\beta := \| r_0 \|_2$ and $v_1 = \frac{r_0}{\beta}$
2: Define the $(m+1) \times m$ matrix $\bar{H}_m = \{h_{ij}\}_{1 \leq i \leq m+a, 1 \leq j \leq m}$, set $\bar{H}_m = 0$
3: Compute the arnoldi iteration for $j = 1, \ldots, m$
4: Compute y_m, the minimizer of $\beta e_1 - \bar{H}_m y$ and $x_m = x_0 + V_m y_m$

Then, the GMRES method converges in m iterations, where m is not determined a priori, but in general, there is a maximum, $m_{\max}$ in which convergence can be achieved. If convergence is not achieved in that amount of iterations, then a restarting is triggered, which means that the GMRES method returns to iteration zero, with an initial solution given by $\mathbf{x}_{m_{\max}}$.

B.2. Matrix Preconditioning

Given a nonsingular matrix $\mathbf{A}$ of $n \times n$ entries, assume that its data is perturbed by εE, then according to Saad (2003), the solution $x(\varepsilon)$ of the perturbed system satsfies the equation:

$$(\mathbf{A} + \varepsilon \mathbf{E})x(\varepsilon) = b + \varepsilon e$$

and its relative variation is such that:

$$\frac{\| x(\varepsilon) - x \|_p}{\| x \|_p} \leq \varepsilon \| \mathbf{A} \|_p \| \mathbf{A}^{-1} \|_p \left(\frac{\| e \|_p}{\| b \|_p} + \frac{\| \mathbf{E} \|_p}{\| \mathbf{A} \|_p} \right) + o(\varepsilon).$$

The quantity $\kappa_p(\mathbf{A}) = \| \mathbf{A} \|_p \| \mathbf{A}^{-1} \|_p$ is called the condition number of the linear system. For the spectral norm, $\| \cdot \|_2$, the condition number is given by:

$$\kappa_2(\mathbf{A}) = \frac{\sigma_{\max}(\mathbf{A})}{\sigma_{\min}(\mathbf{A})} \tag{B.2}$$

B.3. Algebraic preconditioners

As stated in the previous section, algebraic preconditioning techniques seeks to find a matrix $\mathbf{P}$, such that its application to an ill conditioned linear system transforms such system into a matrix spectrally similar to the identity. On these terms, the best preconditioner that can be built is the inverse matrix itself, and then, finding a preconditioner could be summarized into constructing a matrix similar to the original system. This holds the key behind algebraic preconditioners.

In this section, some of the most acknowledged methods to build algebraic preconditioners are presented.

B.3.1. Jacobi, Gaus Seidel and SOR preconditioners

Saad (2003) presents some basic iterative methods. To build a preconditioner, a spliting of a matrix $\mathbf{A} = \mathbf{M} - \mathbf{N}$ is necessary. Then, the preconditioned system reads:

$$\mathbf{M}^{-1}\mathbf{A}\mathbf{x} = \mathbf{M}^{-1}\mathbf{b}$$

Depending on the method, the splitting in (B.3.1) changes to $\mathbf{A} = \mathbf{D} - \mathbf{E} - \mathbf{F}$, where $\mathbf{D}$ is a diagonal matrix. Then the splitting factor in (B.3.1) changes to:

$$\begin{aligned}
\mathbf{M}_{Jacobi} &= \mathbf{D} \\
\mathbf{M}_{Gauss-Seidel} &= \mathbf{D} - \mathbf{E} \\
\mathbf{M}_{SOR} &= (\mathbf{D} - \omega\mathbf{E}) \\
\mathbf{M}_{SSOR} &= (\mathbf{D} - \omega\mathbf{E})\mathbf{D}^{-1}(\mathbf{D} - \omega\mathbf{F})
\end{aligned}$$

B.3.2. Incomplete LU factorization

The idea behind this method is to perform a partial LU factorization of a matrix $\mathbf{A}$, that is, a sparse lower triangular matrix, $\mathbf{L}$ and sparse upper triangular matrix $\mathbf{U}$, so the residual matrix $\mathbf{R} = \mathbf{L}\mathbf{U} - \mathbf{A}$ satisfies some constraints.

According to K. Chen (2005) :

> Although theoretically the optimal $\mathbf{L}, \mathbf{U}$ may be dense, even for a general sparse matrix, we can impose some pattern restrictions on $\mathbf{L}, \mathbf{U}$ to maintain efficiency. If $\mathbf{A}$ is a dense matrix, we may make a suitable threshold on entries of $\mathbf{A}$ before considering an ILU preconditioner.

Saad (2003) describes the ILU preconditioning technique as a set of preconditioners, $\text{ILU}_{\mathcal{P}}$, where $\mathcal{P}$ is a zero pattern used for dropping entries in order to compute a sparse version of $\mathbf{L}$ and $\mathbf{U}$. The zero pattern is such that:

$$\mathcal{P} \subset \{(i,j)|i \neq j; 1 \leq i,j \leq n\}$$

Then, the algorithm for the $\text{ILU}_{\mathcal{P}}$ is performed as in 6.

Algorithm 6 ILU_P

1: **for** $k = 1, \ldots, n-1$ **do**
2: **for** $i = k+1, \ldots, n$ and if $(i,k) \notin \mathcal{P}$ **do** $a_{ik} := a_{ik}/a_{kk}$
3: **for** $j = k+1, \ldots, n$ and for$(i,j) \notin \mathcal{P}$ **do** $a_{ij} := a_{ij} - a_{ik}a_{kj}$
4: **end for**
5: **end for**
6: **end for**

Thus, the quality of the preconditioner is determined by the $\mathcal{P}$ pattern, so the question that arises now is how to find such pattern. There are different approaches to answer this question. The most classical is the so called $\text{ILU}_{(0)}$ factorization, which takes the $\mathcal{P}$ pattern from the zero entries of the original matrix, $\mathbf{A}$, but ,ore accurate approaches can be found, such as $\text{ILU}_{(1)}$, which "results from taking $\mathcal{P}$ to be the zero pattern of the product $\mathbf{LU}$ of the factors $\mathbf{L}, \mathbf{U}$ obtained from $\text{ILU}_{(0)}$" (Saad, 2003).

B.3.3. Approximate inverse preconditioners

There are several aproaches to build this kind of preconditioners. The first one is to use the fact that $\mathbf{A}^{-1}$ can be expressed by the Neumann series (if $\rho(\mathbf{B}) < 1$):

$$(\mathbf{I} - \mathbf{B})^{-1} = \mathbf{I} + \mathbf{B} + \mathbf{B}^2 + \mathbf{B}^3 + \ldots$$

Using $\mathbf{B}$ as $\mathbf{I} - \mathbf{A}$, then he the best kth low degree polynomial is targeted as a preconditioner.

For a diagonalizable matrix, $\mathbf{A} = \mathbf{X}\mathbf{D}\mathbf{X}^{-1}$ a different approach is to find the best low kth degree polynomial:

$$p_k(\mathbf{A}) = c_k\mathbf{A}^k + c_{k-1}\mathbf{A}^{k-1} + \cdots + c_1\mathbf{A} + c_0\mathbf{I}$$

and solve the problem:

$$\min_{p_k(\mathbf{A})} \parallel \mathbf{I} - p_k(\mathbf{A})\mathbf{A} \parallel_\infty \leq \kappa(\mathbf{X}) \min_{p_k(\mathbf{A})} \max_{\lambda \in \Lambda(\mathbf{A})} |1 - p_k(\lambda)\lambda|$$

Where $\Lambda(\mathbf{A})$ is the set of eigenvalues of $\mathbf{A}$. For a fixed integer k Chebyshev polynomials are used to find a solution to (B.3.3).

Another approach aims to compute a $\mathbf{A}^{-1}$ by minimizing $\parallel \mathbf{I} - \mathbf{A}\mathbf{M} \parallel$ instead of finding a polynomial $\mathbf{M} = p_k(\mathbf{A})$.

In particular, the SPAI preconditioning strategy considers a right preconditioner $\mathbf{M}$ with a sparsity pattern $\mathcal{S}$, such that $\mathbf{M}$ will be the best matrix that has the pattern $\mathcal{S}$ and minimizes the functional

$$\min_{\mathbf{M}} \parallel \mathbf{A}\mathbf{M} - \mathbf{I} \parallel_F^2 = \min_{\mathbf{M}} \parallel \mathbf{M}\mathbf{A} - \mathbf{I} \parallel_F^2$$

thus, by using the Frobenius norm the problem decouples into n least squares problems

$$\min_{m_j} \parallel \mathbf{A}m_j - e_j \parallel_2^2, \ j = 1, \ldots, n$$

Here (B.3.3) usually represents an usually small-sized least squares problem to solve if $\mathbf{A}$ is sparse, so parallel algorithms are required to solve them efficiently.

B.4. Some definitions on Hierrchical matrices

(i) **Partition:** Let $\mathcal{I}, \mathcal{J} \subset \mathbb{N}$. A subset $P \subset \mathcal{P}(\mathcal{I} \times \mathcal{J})$ of the set of subsets of $\mathcal{I} \times \mathcal{J}$ is called partition if

$$\mathcal{I} \times \mathcal{J} = \bigcup_{b \in P} b$$

and if $b_1 \cap b_2 = \emptyset$ implies $b_1 = b_2$ for all $b_1, b_2 \in P$. The elements $b \in P$ are called index blocks.

(ii) **Tree:** (r, V, E, L) is a tree, T, with a root r, if the following holds:

- V is a non-empty set of vertices and $E \subset V \times V$ is a set of edges.
- $r \in V$ and for all $v \in V$ there exists a unique path from r to v, i.e, a tuple of vertices, $(v_i)_{i=0}^{l}$, such that $(v_{i-1}, v_i) \in E$ holds for all $i \in \{1, \ldots, l\}$ and $v_i \neq v_j$ if $i, j \in \{1, \ldots, l\}$, $i \neq j$, and $v_0 = r$, $v_l = v$

(iii) **Cluster Tree:** A tree is called a cluster tree, if the following conditions hold:

- root$(T) = \mathcal{I}$
- For all $v \in V$, it holds sons$(v) = \emptyset$, or $v = \cup_{s \in \text{sons}(v)} s$

(iv) **Diameter of a cluster** (diam):

$$\text{diam}(\tau) := \max\{\|x' - x''\|_2 : x', x'' \in X_\tau\}, \quad \tau \subset \mathcal{I}$$

(v) **Distance between clusters** (dist):

$$\text{diam}(\tau, \sigma) := \min\{\|x - y\|_2 : x \in X_\tau, y \in Y_\sigma\}, \quad \tau \subset \mathcal{I}, \quad \sigma \subset \mathcal{J}$$

(vi) **Level:** Let T be a tree, let t be a leaf and $t_0, \ldots, t_m$ be a sequence of ancestors, then $m \in \mathbb{N}_0$ is called the level of T.

(vii) **Depth:** The maximal level of a tree is called depth.

(viii) **Cardinality of a cluster:** Is the number of elements inside a cluster.

(ix) **Leafsize:** Is the minumum cardinality of a cluster.

(x) **Geometric and cardinality balanced clustering:** There are different methods for clustering, two of them are frequently used:

- Cardinality balanced clustering: This cardinality clustering method provides a balanced cluster tree. It starts with a d-dimensional box that contains all the points of interest x_i (in this case, the nodes of the mesh). Then, this box is split in two boxes, such that each one contains the same number of points. This process is repeated as long as the size of the corresponding clusters is greater than a given leafsize.
- Geometric Clustering: This method is based on partitioning the smallest d-dimensional box that contains all the points of interest x_i (in this case, the nodes of the mesh), $B_{\mathcal{I}} := [a_1, b_1] \times \cdots \times [a_d, b_d]$. Then, $B_{\mathcal{I}}$ is split in the direction of maximal extent, creating a disjoint partition of $B_{\mathcal{I}}$. This procedure is then applied recursively to the bounding boxes of the new partition as long as the size of the corresponding clusters is greater than a given leafsize.

(xi) **Admissibility:** Let $\eta > 0$, $t \in T_{\mathcal{I}}$ and $s \in T_{\mathcal{J}}$ two clusters and Ω_t, Ω_s their corresponding cluster supports. The cluster pair (t, s) is η-admissible if there holds:

$$\min\{\text{diam}(\Omega_t)\text{diam}(\Omega_s)\} \leq \eta \text{dist}(\Omega_t, \Omega_s)$$

(xii) **Low rank matrix:** A matrix $A \in C_k^{m \times n}$ is called a matrix of low rank if $k(m + n) < m \cdot n$

The low-rank reppresentation for a matrix of $m \times n$ entries is given by:

$$A = \sum_{i=1}^{r} u_i v_i^*, \quad u_i \in \mathbb{C}^m, \quad v_i \in \mathbb{C}^n$$

Where r is the rank of the approximation and $u_i v_i^*$ is an outer product. Hence, instead of storing the $m \cdot n$ entries of $A \in C_k^{m \times n}$, the vectors $u_i, v_i, i = 1, \ldots, k$ require $k \cdot (m + n)$ units of storage. Then, the matrix-vector product is given by:

$$Ax = \sum_{i=1}^{r} s_i u_i, \quad u_i \in \mathbb{C}^m, \quad s_i = v_i^* x$$

Hence, instead of $2m \cdot n$ arithmetic operations which are required in the full representation, the outer-product form amounts to $2k(m+n) - k$ operations.