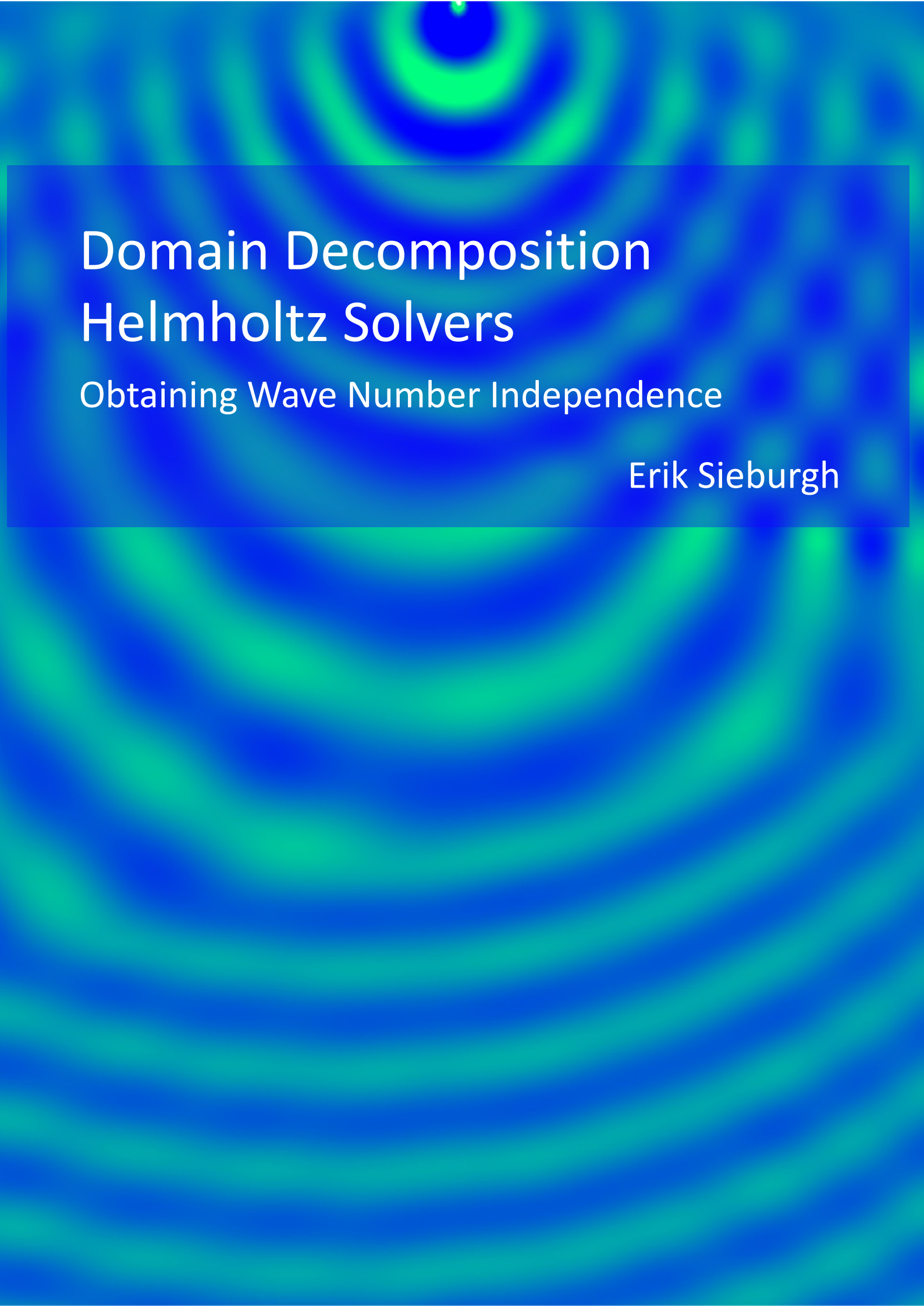


Domain Decomposition Helmholtz Solvers

Obtaining Wave Number Independence

Erik Sieburgh

Domain Decomposition Helmholtz Solvers

Obtaining Wave Number Independence

By
Erik Sieburgh

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Monday October 31, 2022 at 16:00.

Student number:	4611799	
Project duration:	January 1, 2022 – October 31, 2022	
Thesis committee:	dr. A. Heinlein,	TU Delft, supervisor
	dr. V. Dwarka,	TU Delft, supervisor
	Prof. dr. C. Vuik,	TU Delft
	Prof. dr. H.M. Schuttelaars,	TU Delft

Delft University of Technology

Faculty of Electrical Engineering,
Mathematics and Computer Science

Contents

Abstract

Wave phenomena play an important role in many different applications such as MRI scans, seismology and acoustics [41, 49, 47]. At the core of such applications lies the Helmholtz equation, which represents the time-independent version of the wave equation. Simulating a Helmholtz problem numerically with accurate numerical solutions for large wave numbers is challenging. Numerical solvers for the Helmholtz problem have to balance having accurate numerical solutions, requiring a number of iterations to reach convergence that is independent of the wave number and solving with linear time complexity with respect to the grid nodes. Currently, there is no numerical Helmholtz solver that can satisfy these requirements at once.
We developed Schwarz domain decomposition preconditioners which leads to wave number independent convergence for wave numbers in 2D and 3D, while remaining to have accurate numerical solutions. The preconditioners use two-level Schwarz preconditioners, with the coarse problem being constructed using higher-order interpolation with quadratic rational Bézier curves. The developed domain decomposition preconditioners are designed to leverage parallel computing in the future in an attempt for the preconditioners to acquire the ability to solve with linear time complexity.

In this research, the preconditioner resulting in wave number independent convergence and the lowest iteration count is the two-level scaled hybrid Schwarz preconditioner with a coarse problem constructed using higher-order Bézier interpolation. This preconditioner uses a deflation method to remove unwanted eigenvalues. Removing these unwanted eigenvalues results in a clustering of the eigenvalues which is more favourable for GMRES. Currently, all the developed preconditioners suffer from high computational cost for large wave numbers, due to the coarse problem becoming large. Decreasing the coarse problem size of the preconditioners, while remaining to have wave number independent convergence, has shown to been unsuccessful. To better understand the required conditions for wave number independent convergence of the preconditioners, we investigated the relationship between the number of coarse grid nodes and the wave length, too see if there is anything generalizable about this relationship and wave number independent convergence of the solvers.
In conclusion, the balancing for a Helmholtz solver to have accurate numerical solutions, requiring a number of iterations to reach convergence that is independent of the wave number and solving with linear time complexity is again shown to be difficult. This work provides the initial development and testing of promising wave number independent Helmholtz solvers, from which more research should follow that tackle its biggest computational problems.

Preface

With this thesis, I am finishing the masters degree of Applied Mathematics at the Delft University of Technology. In July of 2019, I finished the Bachelor of Mechanical Engineering at the TU Delft, after which I started the pre-master program for the Applied Mathematics master. During the pre-master, which I finalized in July of 2022, I learned a lot about my capabilities and interests. Transitioning to Applied Mathematics was not always easy and fun, but this was not surprising to me.
When I started with the master Applied Mathematics in September of 2022, I was interested in my different topics, but ended up choosing the Computational Science and Engineering specialisation. Looking back, I am happy with the choices that I made and that I pursued this masters degree. Currently, I am still unsure what I will be doing in the future. My interest in pursuing a academic and scientific career are still high after this thesis, which is why I will be looking at PhD candidate positions in Applied Mathematics.

While working on this thesis I was supported by many people, who I would like to thank personally. The first people I would like to thank are my supervisors Vandana and Alexander. Without your feedback, ideas, time, and motivation this work would have not been possible. Besides teaching me a lot about the problems surrounding iterative methods for the Helmholtz problem, Vandana also helped me with my scientific programming, was always enthusiastic about my research and helped me be critical of my work. From Alexander, I always got fast and thoughtful feedback on everything related to my thesis and more. You always seemed to make time for me and be interested in what I had been up to, for which I can not thank you enough. Also on the topic of domain decomposition I learned a lot from you and we had many interesting discussions on this topic related to my research.
Furthermore, I want to thank Kees Vuik for being the chair of my thesis committee and giving his thoughts on my work during my literature presentation and greenlight meeting. Additionally, I want to thank Henk Schuttelaars for being in my thesis committee and being an inspiring teacher during the first and tough semester of my pre-master.
Finally, I want to thank some of my friend who helped me with my thesis and made these last few much months a lot of fun. My thanks goes to Casper, Titus, Willem, Teun, Antoine and Leon, for your help, time and friendship. A special thanks has to go to Sylle, with whom I have spend most of my time working on this thesis, which made it much more fun. Taking my mind of my thesis by traveling to Stockholm or just having a coffee and chatting helped a lot and I will miss that.

I have really enjoyed working on this thesis over the past couple of months. Thank you for taking the time to read it, and I hope you will find it as interesting as I do!

Erik Sieburgh
Den Haag, October 2022

1 Introduction

The Helmholtz equation is a second order partial differential equation that models wave phenomena. The equation appears in many applications in fields such as nuclear fusion, seismology and MRI scanners [49, 47, 41]. For example, in seismic exploration, we are interested in how waves travel through the layers of the earth's crust. Another example is in nuclear fusion. Electromagnetic waves are used when studying charged particle, which is essential for the development of nuclear fusion for powering society. Also, studying the Helmholtz equation is relevant for the study of more complex wave propagation in the time domain.
The Helmholtz equation itself seems nice and simple, but simulating the Helmholtz problem with accurate numerical solutions for large wave numbers is difficult due to several problems. The one dimensional Helmholtz linear system matrix, which arises from discretizing the one dimensional Helmholtz problem, becomes indefinite when the wave number becomes larger than π [42]. An error appears in the numerical solutions of the Helmholtz problem which increases with the wave number for a fixed problem size. In order to avoid this error the grid has to be refined when the wave number increases. This refinement of the grid results in the linear system becoming very large as the wave number increases. Therefore, iterative solvers have to be used to solve the linear system, since direct solution methods become rapidly computationally expensive and unusable for large linear systems. For standard iterative methods, the number of iterations to reach convergence increases as the wave number increases, i.e., the standard iterative solvers have wave number dependent convergence.

Despite years of research into numerical solvers for the Helmholtz problem, no iterative method has been developed that has accurate numerical solutions, requires a number of iterations to reach convergence that is independent of the wave number and solves with linear time complexity with respect to the grid nodes. The Complex Shifted Laplacian preconditioner (CSLP) is a good and frequently used preconditioner which accelerates the convergence rate of standard iterative methods [20, 28]. But the CSLP is unusable for high frequency Helmholtz problems, since the number of iterations required to reach convergence is wave number dependent and the computational cost becomes high for large wave numbers.
A more recently Helmholtz solver, which makes use of multi-level higher-order deflation vectors to accelerate the iterative solver, performs very well [43]. This solver gives accurate numerical solutions while requiring a number of iterations to reach convergence which is close to independent of the wave number and the solver has close to linear time complexity with respect to the grid nodes.

One of the ways of tackling the wave number dependence and time complexity problem is by making use of domain decomposition methods to accelerate iterative solvers. This is because domain decomposition methods can provide scalability with respect to the number of grid nodes, while also being suitable for parallelization [11, 18, 21]. Using a domain decomposition method that is suitable for parallel computing could be beneficial in obtaining linear time complexity of the Helmholtz solver. Therefore, in this research domain decomposition methods are studied. Specifically, using two-level Schwarz domain decomposition preconditioners which can accelerate the iterative method are investigated.
Besides domain decomposition, this work also makes use of a higher-order approximation scheme from the above mentioned developed Helmholtz solver [42, 43]. The idea of the higher-order approximation scheme is suitable and promising and can be combined with domain decomposition preconditioners.

The ultimate goal is to develop a numerical Helmholtz solver which gives accurate numerical solutions, is wave number independent and has linear time complexity with respect to the grid nodes. Developing such a solver is done through domain decomposition methods and a higher-order approximation scheme. We call a solver efficient if the numerical solutions are accurate with respect to the exact solution. Additionally, we call a solver numerically scalable if the following two requirements are satisfied. Firstly, the number of iterations required for the solver to reach convergence is wave number independent. Secondly, the solver is required to have a time complexity with respect to the number of grid nodes of $\mathcal{O}(n)$, i.e. linear time complexity.
In this thesis we focus on developing a Helmholtz solver that is both efficient and has wave number independent convergence, and choose to ignore the time complexity of the solver for now. The development of such a solver is done by answering the following research questions:

1. Does a Helmholtz solver constructed using a two-level additive Schwarz preconditioner combined with **first-order** grid coarse problem show **wave number independent convergence** and **efficiency**?
    - How can the performance of the Helmholtz solver using a first-order grid coarse problem be explained?
2. Does a Helmholtz solver, constructed using a two-level additive Schwarz preconditioner and a **higher-order Bézier** coarse problem, show **wave number independent convergence** and **efficiency**?

- What causes the solver to be inscalable if this is the case?

3. Does a Helmholtz solver, constructed using different variants of two-level Schwarz preconditioners and a **higher-order Bézier** coarse problem, show **wave number independent convergence** and **efficiency**?
    - For the scaled additive Schwarz preconditioner, does the solver show wave number independent convergence and efficiency and if so, why?
    - For the scaled hybrid Schwarz preconditioner, which uses deflation, does the solver show wave number independent convergence and efficiency and if so, why?

The methodology is the following. It is split up into two phases: numerical results and eigenvalue analysis.
The first phase, is about performing numerical experiments and analysing the numerical results of the developed Helmholtz solvers. These experiments are performed by fixing the number of grid nodes per subdomain and increasing the wave number and number of subdomains at the same time. By doing this the total number of grid nodes increases with the wave number, i.e., the grid is refined with the wave number increasing. The tested preconditioners impact the number of iterations that are required for the iterative method to reach convergence. As the wave number and the number of subdomains grow, we look at the number of iterations that are required for the iterative method to reach convergence and see how the number of iterations that are required changes as the wave number grows, as a consequence of changing the preconditioner. The numerical experiments are classed into two categories with different goals. The first category are the experiments used to see whether the solver reaches convergence in a number if iterations that is independent of the wave number while also being efficient. The other category of experiments, only used when wave number independent convergence is found, has the goal to see how the solver performs on model problems that are more like the real engineering problems, i.e., a nonconstant wave number or a 3D problem.
In the other phase, an eigenvalue analysis is performed to learn more about the behaviour and performance of the developed solvers. One of the ways this analysis is done is by plotting the eigenvalues of the preconditioned linear system. This gives insights into the performance of the solver, since the clustering, numerical range and location of the eigenvalue influences the number of iterations required to reach convergence of the iterative solver.

This thesis is structured as follows. In chapter two, the derivation of the Helmholtz equation and the one dimensional analytical solution of the Helmholtz equation are given. After that, in chapter three, details on modeling and numerically solving the Helmholtz problem are presented. Additionally, the main problems that arise when solving the Helmholtz problem numerically are given here. Details about Schwarz domain decomposition preconditioners and theory are given in chapter four, since Schwarz domain decomposition preconditioning will be used in order to accelerate an iterative solver. In chapter five, details the construction of grid coarse problems, considered in this thesis, are given. The discussion of the numerical results of the developed solvers are split in chapter six and seven. In chapter six, two-level additive Schwarz preconditioner using a first-order grid coarse problem are tested and analysed. In chapter seven, the first-order grid coarse problem from the previous chapter is replaced by a higher-order Bézier grid coarse space. Additionally, different variants of two-level Schwarz preconditioners such as the hybrid Schwarz preconditioner and scaled hybrid Schwarz preconditioner are tested and analysed. Finally, in chapter eight a conclusion and discussion of the research in this thesis are given. This thesis also includes several appendices where additional information can be found.

2 | The Helmholtz Equation

This chapter is the start of the analysis of the Helmholtz equation. The Helmholtz equation is named after the physicist Hermann von Helmholtz who is the creator of the equation. It is a second order partial differential equation that models time harmonic wave phenomena.
The structure of this chapter is as follows: first the derivation of the Helmholtz equation with a constant wave number from the homogeneous wave equation is given, followed by an explanation of different boundary conditions and, the construing of the analytical solution to the one dimensional nonhomogeneous Helmholtz equation with a constant wave number. Finally, we show how make the boundary value problem dimensionless.

2.1 Derivation

The homogeneous wave equation with constant c is

$$\left(\frac{\partial^2}{\partial t^2} - c^2\nabla^2\right) u(\mathbf{x}, t) = 0, \tag{2.1}$$

where $u(\mathbf{x}, t)$ denotes the wave displacement.

Let us now apply the method of separation of variables. We begin by separating the solution $u(\mathbf{x}, t)$ in a time-dependent and time-independent part. This solution is then written in the following form:

$$u(\mathbf{x}, t) = h(t)\phi(\mathbf{x}). \tag{2.2}$$

Substituting (2.2) into (2.1) gives the results

$$\left(\frac{\partial^2}{\partial t^2} - c^2\nabla^2\right) h(t)\phi(\mathbf{x}) = 0. \tag{2.3}$$

By moving the time dependent parts to the left side, the space dependent parts to the right side and setting this equation equal the separation constant which we pick to be $-k^2$, we get the following equation

$$\frac{1}{c^2 h(t)}\frac{d^2 h(t)}{dt^2} = \frac{1}{\phi(\mathbf{x})}\nabla^2\phi(\mathbf{x}) = -k^2. \tag{2.4}$$

Thus we obtain two equations

- $$\frac{d^2 h(t)}{dt^2} = -k^2 c^2 h(t), \tag{2.5}$$
- $$\nabla^2\phi(\mathbf{x}) = -k^2\phi(\mathbf{x}). \tag{2.6}$$

Now, (2.6) gives is the Helmholtz equation, which can be rewritten to the more conventional form of

$$\left(-\nabla^2 - k^2\right)\phi(\mathbf{x}) = 0. \tag{2.7}$$

with k being the wave number, which is defined as

$$k = \frac{2\pi}{\lambda}, \tag{2.8}$$

where λ is the wave length.

The nonhomogeneous Helmholtz equation is given when the right hand side of (2.7) is replaced by a function $f(\mathbf{x})$. For this research, the nonhomogeneous part is defined as the source function given by

$$f(\mathbf{x}) = \delta(\mathbf{x} - \mathbf{x}_0). \tag{2.9}$$

The nonhomogeneous Helmholtz equation is therefore given by

$$\left(-\nabla^2 - k^2\right)\phi(\mathbf{x}) = \delta(\mathbf{x} - \mathbf{x}_0). \tag{2.10}$$

2.2 Boundary Conditions

In order to solve the Helmholtz equation on a domain Ω, boundary conditions are necessary for the problem to not be ill-posed. In problems with the Helmholtz equation the following boundary conditions are generally used. Also note that for a domain Ω the boundary of that domain is indicated by $\partial\Omega$.

- **Dirichlet Boundary Conditions**:
  Also sometimes called boundary conditions of the first type, are boundary conditions that specify a value on the boundary of the domain. Nonhomogeneous Dirichlet boundary can generally be written as

$$\phi(\mathbf{x}) = g(\mathbf{x}), \quad \text{for } \mathbf{x} \in \partial\Omega. \tag{2.11}$$

  In the homogeneous case, so $g(\mathbf{x}) = 0 \, \forall \mathbf{x} \in \partial\Omega$, the Dirichlet boundary conditions is sometimes referred to as a vanishing boundary conditions

- **Neumann Boundary Conditions**:
  or second-type boundary conditions, are boundary conditions that specify a value to the derivative on the boundary. Generally, this is boundary conditions is expressed as

$$\frac{\partial\phi(\mathbf{x})}{\partial\mathbf{n}}\phi(\mathbf{x}) = h(\mathbf{x}), \quad \text{for } \mathbf{x} \in \partial\Omega. \tag{2.12}$$

  $\mathbf{n}$ denotes unit vector pointing outward with respect to the boundary Ω. The homogeneous, so $h(\mathbf{x}) = 0 \, \forall \mathbf{x} \in \partial\Omega$, Neumann boundary condition is sometimes called the reflective boundary conditions.

- **Robin Boundary Conditions**:
  Robin boundary conditions are a weighted combination of the first two boundary conditions. Its nonhomogenous case is therefore expressed as

$$\left(a\frac{\partial\phi(\mathbf{x})}{\partial\mathbf{n}} + b\phi(\mathbf{x})\right) = k(\mathbf{x}), \quad \text{for } \mathbf{x} \in \partial\Omega, \tag{2.13}$$

  with $a, b \in \mathbb{C}$ being the weight coefficients. Similarly as before, the homogeneous case is given when $k(\mathbf{x}) = 0 \, \forall \mathbf{x} \in \partial\Omega$.

- **Sommerfeld Radiation Condition**:
  Often, the Sommerfeld radiation condition is used for the Helmholtz problem to introduce some damping to the problem. The damping is beneficial since it allows the Helmholtz problem to be controlled more easily. It should be easier to construct an iterative Helmholtz solver that is wave number independent for the Helmholtz problem when the Sommerfeld radiation condition is applied to boundary, compared to when the homogeneous Dirichlet boundary condition is applied to the Helmholtz problem. The Sommerfeld radiation condition is a more specific version of the Robin boundary conditions. Sometimes the Sommerfeld radiation conditions is also referred to as the absorbing boundary condition and it is represented as

$$\left(\frac{\partial\phi(\mathbf{x})}{\partial\mathbf{n}} - ik\phi(\mathbf{x})\right) = 0, \quad \text{for } \mathbf{x} \in \partial\Omega, \tag{2.14}$$

  where i is the imaginary number and k is the wave number in the Helmholtz equation.

2.3 Analytical Solution of the One Dimensional Helmholtz Equation

In this section we will give the analytical solution to the one dimensional nonhomogeneous Helmholtz equation with a constant wave number on the domain $\Omega = [0, L]$ and with homogeneous Dirichlet boundary conditions. From now on, we will use $u(x)$ instead of $\phi(x)$. The boundary value problem is given by

$$\begin{cases} -\frac{d^2u(x)}{dx^2} - k^2u(x) = \delta(x - \frac{L}{2}), & \text{in } \Omega = [0, L] \subset \mathbb{R}. \\ u(x) = 0 & \text{for } x \in \partial\Omega. \\ k \in \mathbb{N}\backslash\{0\}. \end{cases} \tag{2.15}$$

Note that the harmonic source term is placed in the middle of the domain and note that k is independent of x. For some problems, a wave number which is dependent on x is chosen ($k = k(x)$). This is for example the case when modeling the Helmholtz equation in a nonhomogeneous medium.

The boundary value problem of (2.15) can be expressed in terms of the Green's function $G(x, x')$. First we note that the Sturm-Liouville operator given by

$$\mathcal{L} = \frac{d}{dx}\left[p(x)\frac{d}{dx}\right] + q(x). \tag{2.16}$$

Setting $p(x) = -1$ and $q(x) = -k^2$, we find the Sturm-Liouville operator for the boundary value problem of (2.15).
Therefore, we can now write the boundary value problem with the Green's function, with a general location x' instead of $\frac{L}{2}$ for the source location, as

$$\begin{cases} \mathcal{L}(G(x, x')) = \delta(x - x'), & \text{in } \Omega = [0, L] \subset \mathbb{R}. \\ G(x, x') = 0 & \text{for } x \in \partial\Omega. \end{cases} \tag{2.17}$$

An eigenfunction ϕ must satisfy

$$\mathcal{L}(\phi) = -\lambda\sigma(x)\phi. \tag{2.18}$$

The eigenfunctions which belong to different eigenvalues are orthogonal with the weight $\sigma(x)$,i.e. ϕ_n and ϕ_m are orthogonal with weight $\sigma(x)$ for $\lambda_n \neq \lambda_m$.

We take $\sigma(x) = -1$. Thus (2.18) gives us the ODE

$$\frac{d^2\phi}{dx^2} + k^2\phi = -\lambda\phi, \tag{2.19}$$

$$\frac{d^2\phi}{dx^2} + \underbrace{(k^2 + \lambda)}_{\alpha}\phi = 0 \tag{2.20}$$

$$\frac{d^2\phi}{dx^2} = -\alpha\phi \tag{2.21}$$

Independent solution can often be obtained in the form of exponential, $\phi = e^{rx}$. When this expression is substituted into the ODE of (2.21) we find the polynomial

$$r^2 = -\alpha. \tag{2.22}$$

We can distinguish three different cases:

- $\alpha > 0$,
- $\alpha = 0$,
- $\alpha < 0$.

α can not be complex, since α needs to be real for the boundary value problem to have nontrivial solutions. The three cases are analysed separately.
First, $\alpha = 0$. This implies that $r = 0$, which means that we get the only trivial solutions due to the Dirichlet boundary conditions.

Second, $\alpha < 0$. This means that $r = \pm\sqrt{-\alpha}$, which suggests a solution of the form

$$\phi(x) = c_1\ \cosh(\sqrt{-\alpha}x) + c_2\ \sinh(\sqrt{-\alpha}x). \tag{2.23}$$

Applying the boundary conditions, $\phi(0) = \phi(L) = 0$, gives us that $\phi(x) = 0$. This is also a trivial solution.

Finally, we look at the case $\alpha > 0$. This means that $r = \pm i\sqrt{\alpha}$. This suggests that the general solution is given by

$$\phi(x) = c_1\cos(\sqrt{\alpha}x) + c_2\sin(\sqrt{\alpha}x). \tag{2.24}$$

Applying the boundary conditions we find that $c_1 = 0$ and $\sin(\sqrt{\alpha}x) = 0$. Thus we find the eigenfunctions and eigenvalues

$$\phi_n = c_2\ \sin(\frac{n\pi x}{L}), \tag{2.25}$$

$$\alpha_n = \frac{n^2\pi^2}{L^2}\text{ , for } n = 1, 2, 3, \ldots . \tag{2.26}$$

But since $\alpha = k^2 + \lambda$, we find

$$\phi_n = \sin(\frac{n\pi x}{L}), \tag{2.27}$$

$$\lambda_n = \frac{n^2\pi^2}{L^2} - k^2 \text{ , for } n = 1, 2, 3, \dots . \tag{2.28}$$

c_2 is an arbitrary multiplicative constant, so we can say $c_2 = 1$.

We will seek to solve (2.15) by using the method of eigenfunction expansion. This gives us that the solution $u(x)$ can be expressed as a series of sines in the following way

$$u(x) = \sum_{n=1}^{\infty} \alpha_n(x') \sin(\frac{n\pi}{L}x). \tag{2.29}$$

We also know from (2.17) that

$$u(x) = G(x, x'). \tag{2.30}$$

$G(x, x')$ can now be expressed, since we know that $\lambda_n \neq 0 \, \forall n \in \mathbb{N}\backslash\{0\}$. This give the following

$$u(x) = G(x, x') = \sum_{n=1}^{\infty} \frac{\phi_n(x)\phi_n(x')}{-\lambda_n \int_0^L \phi_n^2(x)\sigma \, dx}. \tag{2.31}$$

$$= \frac{2}{L} \sum_{n=1}^{\infty} \frac{\sin(\frac{n\pi}{2}) \sin(\frac{n\pi}{L}x)}{\left(\frac{n^2\pi^2}{L^2} - k^2\right)}, \text{ for } k^2 \neq \frac{n^2\pi^2}{L^2} \text{ and } n = 1, 2, 3, \dots . \tag{2.32}$$

Now we have expressed the solution $u(x)$ to the boundary value problem of (2.15).

2.4 Dimensionless Helmholtz Model

Equation (2.15) can be made dimensionless. The goal is to map the problem onto the unit domain $[0, 1]$. Equation (2.15) uses the arbitrary domain $[0, L]$. To get a dimensionless model we introduce the new variable $\widehat{x}$ such that

$$\widehat{x} = \frac{x}{L}, \tag{2.33}$$

from which it follows that

$$\frac{d\widehat{x}}{dx} = \frac{1}{L}. \tag{2.34}$$

Now equation (2.15) can be written as

$$\begin{cases} -\frac{d^2u(\widehat{x})}{d\widehat{x}^2} - \widehat{k}^2 u(\widehat{x}) = L^2\delta(\widehat{x} - \frac{L}{2}), & \text{in } \Omega = [0, 1] \subset \mathbb{R}. \\ u(\widehat{x}) = 0 & \text{for } \widehat{x} \in \partial\Omega. \\ \widehat{k} = Lk, \text{ with } k \in \mathbb{N}\backslash\{0\}. \end{cases} \tag{2.35}$$

If not otherwise specified, we will use the notation $\widehat{k} = k$ and $\widehat{x} = x$ for the following chapters. The Helmholtz model that are used in this thesis all have domains with $L = 1$. When going to real-life Helmholtz problems, the dimensionless Helmholtz model can be useful, since it allows for better comparison between different sized models.

3 | Iterative Methods & Preconditioners

It becomes very difficult to solve the Helmholtz equation analytically in dimensions higher than one. Therefore, instead of solving the Helmholtz equation analytically, it is more convenient to solve the problem by creating a numerical model and using a numerical method to solve the problem. For the Helmholtz problem, the numerical methods of modeling that are used are the finite element method and the finite difference method. For the research in this report we will only focus the finite difference method numerical scheme.
Modeling the problem numerically is done by discretization of the problem. The one dimensional Helmholtz linear system matrix, which arises from discretizing the one dimensional Helmholtz problem, becomes indefinite when the wave number becomes larger than π [42]. As will become clear in the next section, a large wave number results in the size of the problem becoming large. For medium sized numerical problems, it is still possible to use direct numerical solution methods. In the case of the Helmholtz problem using direct numerical solution methods is not realistic anymore. Therefore, for the Helmholtz problem iterative solution methods are used. Direct solution numerical methods can still serve a purpose as they are often used as solvers for subdomain problems in domain decomposition methods and in multigrid methods.

Before describing the structure of this chapter, it is important to give a few clear definitions of somewhat general terms.

- An *efficient* numerical solver is defined as a iterative numerical method, often combined with a preconditioner, that produces solutions very close to the exact solutions, i.e. the numerical solutions give a good representation of the real problem and are close to the exact solution.
- A *numerically scalable* numerical solver is defined as a iterative numerical method, often combined with a preconditioner, that has two attributes. First, the number of iterations needed to reach convergence does not grow when the problem size increases. Second, the solver needs a time complexity with respect to the number of grid nodes of $\mathcal{O}(n)$.
- A *parallel scalable* numerical solver is defined as a solver where the number of iterations needed to reach convergence does not grow when the number of subdomains (or parallel processes) increases.

This chapter is constructed in the following way. In the first section an introduction and problems arising are given for solver the Helmholtz problem numerically. After that, the boundary value problems are constructed used in this thesis are given. From Section 3.3 onward, background information on general numerical solution methods and numerical solutions methods for the Helmholtz problem are discussed.

3.1 Introduction and Complications when Solving Helmholtz Problems Numerically

When trying to solve the Helmholtz equation numerically there are two major problems that arise [42]. The first problem is that for large wave numbers a so-called pollution error appears in the numerical solution [13, 16, 17]. This pollution error has to do with a phase misalignment of the exact and numerical solution. To deal with this pollution error, the grid has to be kept very fine. Specifically, when a second order finite difference scheme is used the requirement $k^3h^2 \leq 1$, with k being the wave number and h the mesh size, has to hold in order to avoid the pollution error [16]. Thus, as the wave number becomes large, the grid has to be refined further.
Instead of completely avoiding the pollution error, we can also keep the error small. For second order accurate finite difference discretizations it is a rule of thumb to have a minimum of 10 grid points per wave length λ [33], resulting in the reduction of the pollution error. Therefore we define

$$\kappa_h = kh = \frac{2\pi}{10} \approx 0.625. \tag{3.1}$$

This rule of thumb hold for wave numbers that are not too large. For larger wave number it might be necessary to use 20 or 30 grid points per wave length. In general, this rule of thumb is not very good. In [33] it is found that even if κ_h is kept small this does not avoid the pollution error appearing for high wave numbers. Instead, the requirement $k^3h^2 \leq \epsilon$ is introduced. While this requirement does avoid the pollution error entirely when a second order accurate finite difference scheme is used, it causes the problem size to increase very quickly when the wave number increases. Therefore, $\kappa_h \leq 0.625$ is used in this work. The rule of thumb should result in the pollution error being avoided enough and the problem size not becoming too large when k becomes

large.
The second problem that arises when trying to solve the Helmholtz equation numerically is a consequence of the first problem. Due to the grid being required to be refined when k increases the size of the problem also increases and becomes large. Since the size of the problem is large, iterative methods are used as numerical solvers for the Helmholtz problem. As the grid becomes finer the computation cost of most iterative solvers increases, which is a problem. Therefore, a numerically scalable iterative solver has to be used for the Helmholtz problem with large wave numbers.
An iterative Helmholtz solvers that is efficient and almost numerically scalable has been developed [43]. The number of iterations for the solver to converge is more or less wave number independent. It is not really numerically scalable since the solver has a nonlinear time complexity with respect to the number of grid nodes.

3.2 Problem Description

The one dimensional boundary value problem with constant wave number $k > 0$ and a Dirichlet boundary condition is

$$\begin{cases} -\frac{d^2u(x)}{dx^2} - k^2u(x) = \delta(x - x'), & \text{in } \Omega \in \mathbb{R}. \\ u(x) = 0 & \text{for } x \in \partial\Omega, \\ k \in \mathbb{N}\backslash\{0\}. \end{cases} \tag{3.2}$$

This model problem is referred to 1D MP-1 for the remained of this thesis. The two and three dimensional version of this model problem are referred to as 2D MP-1 and 3D MP-1, respectively. Since the 2D MP-1 model problem is used mostly, it is often written without the 2D notation.
The one dimensional boundary value problem for the Helmholtz equation with constant wave number $k > 0$ and a Sommerfeld radiation condition is given by:

$$\begin{cases} -\frac{d^2u(x)}{dx^2} - k^2u(x) = \delta(x - x'), & \text{in } \Omega \in \mathbb{R}. \\ \left(\frac{\partial u(x)}{\partial \mathbf{n}} - iku(x)\right) = 0, & \text{for } x \in \partial\Omega, \\ k \in \mathbb{N}\backslash\{0\}, \end{cases} \tag{3.3}$$

The model problem above is referred to as 1D MP-2 and its two dimensional version is referred to as 2D MP-2, where the 2D part is often left out. The difficulty in using Sommerfeld radiation conditions is that the eigenvalues of the resulting system matrix can not be expressed in closed-form and spectral analysis becomes more difficult but it can provide useful heuristics [42]. For this reason, some of the analysis is performed on the 1D boundary value problem without the Sommerfeld radiation condition.
For both 2D MP-1 and 2D MP-2 the domain $\Omega = [0, 1] \times [0, 1]$ is used. Additionally, the right hand side function is defined as in Equation 2.9 with the source location in the middle of the domain.

The numerical experiments performed in this thesis are mostly with MP-1 and MP-2. In order to see how solver behave on easier numerical experiment, a model problem that does not use the Helmholtz equation is also used. This model problem is referred to as MP-3. The problem is a two dimensional nonhomogeneous Poisson's equation problem on the domain $\Omega = [0, 1] \times [0, 1]$, which is described by the following equation

$$-\nabla^2 u(x, y) = f(x, y) \text{ on } \Omega. \tag{3.4}$$

The problem is supplied with nonhomogeneous Dirichlet boundary conditions

$$u(x, y) = u_0(x, y) \text{ on } \partial\Omega. \tag{3.5}$$

Now assume that the exact solution to this problem is given by $u_{ex}(x, y) = \sin(xy)$. This gives us that the source term $f(x, y)$ is given by

$$f(x, y) = (x^2 + y^2)\sin(xy). \tag{3.6}$$

And the boundary data is given by

$$u_0(x, y) = \begin{cases} 0 & \text{if } x = 0 \\ \sin(y) & \text{if } x = 1 \\ 0 & \text{if } y = 0 \\ \sin(x) & \text{if } y = 1 \end{cases} \tag{3.7}$$

Finally, there are two more problem that are used in this thesis which need introduction. Both problems are Helmholtz problems that have an element of a more real-life engineering Helmholtz problem. The first of these

problems is referred to as the wedge model problem, which is a two dimensional Helmholtz problem with nonconstant wave number $\dot{k}(x,y) > 0$. The wedge problem is given by the following boundary value problem.

$$\begin{cases} -\frac{d^2u(x,y)}{d(x,y)^2} - \dot{k}(x,y)^2 u(x,y) = \delta(x-\frac{1}{2}, y-1), & \text{in } \Omega = [0,1]\times[0,1] \in \mathbb{R}^2. \\ \left(\frac{\partial u(x,y)}{\partial \mathbf{n}} - i\dot{k}(x,y)u(x,y)\right) = 0, & \text{for } (x,y) \in \partial\Omega, \\ \dot{k}(x,y) \in \mathbb{R}\backslash\{0\} & \text{for } (x,y) \in \Omega. \end{cases} \tag{3.8}$$

As opposed to the previous boundary value problems, this boundary value problem has a harmonic source term on the top boundary of the unit square domain. Additionally, the wedge boundary value problem has a Sommerfeld radiation condition on the whole boundary and a wave number that is spatially dependant. The following definition is given for $\dot{k}(x,y)$:

$$\dot{k}(x,y) = \begin{cases} k & \text{,if } (x,y) \in \{x,y \in \Omega \mid y \geq 0.75 - 0.2x\} \\ k - 0.5k & \text{,if } (x,y) \in \{x,y \in \Omega \mid (y < 0.75 - 0.2x) \wedge (y > 0.25 + 0.2x)\}\,, \\ k & \text{,if } (x,y) \in \{x,y \in \Omega \mid y \leq 0.25 - 0.2x\} \end{cases} \quad \text{with } k \in \mathbb{N}. \tag{3.9}$$

This equation for $\dot{k}(x,y)$ translates to a domain that looks as given in Figure 3.1 with A = 0.75, B = 0.55, C = 0.45 and D = 0.25.

The second problem which has an element of a real-life engineering Helmholtz problem is the 3D MP-1 problem, since most engineering problems are 3D problems. The boundary value problem for this problem is not given, since it is just the 3D version of MP-1.

The next section shows the discretization of the domain and boundary conditions and the application of a finite difference approximation to the PDE. Which results in a linear system of equations that describes a certain boundary value problem. These steps are shown only for MP-1 and MP-2.

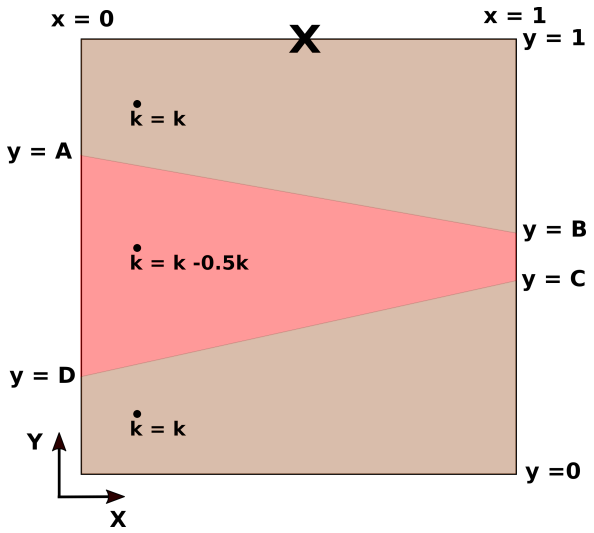


Figure 3.1: Illustration the wedge problem domain with X indicating the location of the harmonic source term.

3.3 Finite Difference Method

In this section the finite difference method in the 2D case of MP-1 of equation (3.2) and MP-2 of equation (3.3) are discussed. First the domain is discretized. Suppose we discretize the domain $\Omega = [0,1]\times[0,1]$ uniformly with mesh size $h = 1/(n-1)$ with n being the number of grid nodes in one dimension. This results in the discretized grid G_h, which has n^2 nodes including the boundary nodes and the discrete grid is defined as

$$G_h = \{(x_i, y_j) |\, x_i = (i-1)h, y_j = (j-1)h;\, h = 1/(n-1), 1 \leq i,j \leq n, n \in \mathbb{N}\}\,. \tag{3.10}$$

And since we also need a discretization of the physics we also have

$$u(x_i,y_j) \approx u^h(x_i,y_j) \text{ for } (x_i,y_j) \in G_h,$$
$$f(x_i,y_j) \approx f^h(x_i,y_j) \text{ for } (x_i,y_j) \in G_h,$$

A global ordering of the grid nodes is defined in order to construct a linear system formulation. The boundary nodes are included in linear system formulation. For the internal and boundary nodes x-lexicographical ordering is introduced. The nodes with coordinates (i,j) are assigned a global index by

$$I = i + (j-1)(n), \text{ for } 1 \leq i,j \leq n+1 \tag{3.11}$$

At the end of this section we want to have a linear system with the discrete solution $\mathbf{u}^h$ which describes MP-1 and MP-2. A linear system is defined as

$$A^h \mathbf{u}^h = \mathbf{f}^h. \tag{3.12}$$

The exact solution, evaluated on the grid nodes, is denoted by $\mathbf{u}^h_{ex}$.
On the grid G_h, a second order accurate finite difference scheme to discretize the PDE of equation (3.2) or (3.3) is applied. The finite difference approximation for the internal nodes of the PDE is given by

$$\frac{-u^h_{i,j-1} - u^h_{i-1,j} + 4u^h_{i,j} - u^h_{i,j+1} - u^h_{i+1,j}}{h^2} - k^2 u^h_{i,j} = f^h_{i,j}\,, \text{ for } 2 \leq i,j \leq n-1. \tag{3.13}$$

The 5-point stencil for the internal nodes is given by:

$$h^{-2} \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 - k^2h^2 & -1 \\ 0 & -1 & 0 \end{bmatrix}.$$

For the 2D boundary value problem of equation (3.3) with Sommerfeld radiation conditions stencils for the boundary nodes can be given. For the boundary nodes there are two cases:

- 4 corner nodes,
- remaining boundary nodes.

Starting with the remaining boundary nodes. Looking, for example, at $u^h_{1,j}$ with $2 \leq j \leq n-1$, then we find the stencil

$$h^{-2} \begin{bmatrix} 0 & -1 & 0 \\ 0 & 4 - k^2h^2 - 2ikh & -2 \\ 0 & -1 & 0 \end{bmatrix}.$$

In order to obtain symmetry of the linear system matrix A^h later on, a rescaling is often performed. This gives the following stencil

$$h^{-2} \begin{bmatrix} 0 & -\frac{1}{2} & 0 \\ 0 & 2 - \frac{k^2h^2}{2} - ikh & -1 \\ 0 & -\frac{1}{2} & 0 \end{bmatrix},$$

where we replace $f^h_{1,j}$ by $\frac{1}{2} f^h_{1,j}$ for $2 \leq j \leq n-1$. For the other boundary nodes of the remaining boundary nodes case, the stencils can be found in the same way.
For the 4 corner nodes case something similar is done. For example, for the corner node of $u^h_{1,1}$, the following stencil is found

$$h^{-2} \begin{bmatrix} 0 & -2 & 0 \\ 0 & 4 - k^2h^2 - 4ikh & -2 \\ 0 & 0 & 0 \end{bmatrix},$$

Also for these boundary nodes rescaling is performed. This give the following stencil for $u^h_{1,1}$

$$h^{-2} \begin{bmatrix} 0 & -\frac{1}{2} & 0 \\ 0 & 1 - \frac{k^2h^2}{4} - ikh & -\frac{1}{2} \\ 0 & 0 & 0 \end{bmatrix},$$

where we replace $f^h_{1,1}$ by $\frac{1}{4} f^h_{1,1}$. The same thing is done for the other corner boundary nodes.

For the 2D boundary value problem with Dirichlet boundary conditions of equation (3.2), different stencils have to be defined for the boundary nodes. The stencil of the boundary nodes is then simply given by

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

with the value on the boundary replacing the corresponding value of $f^h_{i,j}$. In the case of using homogeneous Dirichlet boundary conditions the boundary value is 0 for the whole boundary, so all the boundary nodes get $f^h_{i,j}$ replaced by 0.

Now lets define the size of A^h. For the linear system of 2D MP-2 where the boundary nodes are **not** eliminated we have:

$$A^h \in \mathbb{C}^{n^2 \times n^2}, \text{ and } \mathbf{u}^h, \mathbf{f}^h \in \mathbb{C}^{n^2}$$

$\mathbb{C}$ above is replace by $\mathbb{R}$ for 2D MP-1.

Using the stencils, details about the matrix A^h can be given. It is straight forward to give the linear system formulation of 2D MP-1 of equation (3.2).

$$A^h = \frac{1}{h^2}\begin{pmatrix} h^2 I^h_{n+1} & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & \widehat{T}^h & -\widehat{I}^h & 0 & \cdots & \cdots & 0 \\ 0 & -\widehat{I}^h & \widehat{T}^h & -\widehat{I}^h & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & -\widehat{I}^h & \widehat{T}^h & -\widehat{I}^h & 0 \\ 0 & \cdots & \cdots & 0 & -\widehat{I}^h & \widehat{T}^h & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 & h^2 I^h_{n+1} \end{pmatrix} \in \mathbb{R}^{n^2 \times n^2} \tag{3.14}$$

with

$$\widehat{I}^h = \begin{pmatrix} 0 & 0 & 0 \\ 0 & I^h_{n-1} & 0 \\ 0 & 0 & 0 \end{pmatrix} \in \mathbb{R}^{n\times n}, \widehat{T}^h = \begin{pmatrix} h^2 & 0 & 0 \\ 0 & T^h & 0 \\ 0 & 0 & h^2 \end{pmatrix} \in \mathbb{R}^{n\times n}, \tag{3.15}$$

$$T^h = \begin{pmatrix} 4-k^2h^2 & -1 & 0 & \ldots & \cdots & 0 \\ -1 & 4-k^2h^2 & -1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \ldots & 0 & -1 & 4-k^2h^2 & -1 \\ 0 & \cdots & \cdots & 0 & -1 & 4-k^2h^2 \end{pmatrix} \in \mathbb{R}^{(n-2)\times(n-2)} \tag{3.16}$$

and I^h_{n-2}, I^h_n are identity matrices on $\mathbb{R}^{(n-2)^2\times(n-2)^2}$ and $\mathbb{R}^{n^2\times n^2}$, respectively. For this boundary value problem it is also possible to give a closed-form expression for the eigenvalues of the matrix (3.14).

The linear system formulation for 2D MP-2 of equation (3.3) is given by

$$A^h = \frac{1}{h^2}\begin{pmatrix} \widetilde{I}^h & \dot{I}^h & 0 & \cdots & \cdots & \cdots & 0 \\ \dot{I}^h & \widetilde{T}^h & \dot{I}^h & 0 & \cdots & \cdots & 0 \\ 0 & \dot{I}^h & \widetilde{T}^h & \dot{I}^h & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & \dot{I}^h & \widetilde{T}^h & \dot{I}^h & 0 \\ 0 & \cdots & \cdots & 0 & \dot{I}^h & \widetilde{T}^h & \dot{I}^h \\ 0 & \cdots & \cdots & \cdots & 0 & \dot{I}^h & \widetilde{I}^h \end{pmatrix} \in \mathbb{C}^{n^2\times n^2} \tag{3.17}$$

with

$$\widetilde{I}^h = \begin{pmatrix} \beta & -\frac{1}{2} & 0 & \ldots & \cdots & \cdots & 0 \\ -\frac{1}{2} & \alpha & -\frac{1}{2} & 0 & \cdots & \cdots & 0 \\ 0 & -\frac{1}{2} & \alpha & -\frac{1}{2} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \ldots & 0 & -\frac{1}{2} & \alpha & -\frac{1}{2} & 0 \\ 0 & \ldots & \ldots & 0 & -\frac{1}{2} & \alpha & -\frac{1}{2} \\ 0 & \cdots & \cdots & \cdots & 0 & -\frac{1}{2} & \beta \end{pmatrix} \in \mathbb{C}^{n\times n}, \tag{3.18}$$

where $\beta = 1 - \frac{k^2h^2}{4} - ikh$ and $\alpha = 2 - \frac{k^2h^2}{2} - ikh$ and

$$\widetilde{T}^h = \begin{pmatrix} \alpha & -1 & 0 & \ldots & \cdots & \cdots & 0 \\ -1 & \gamma & -1 & 0 & \cdots & \cdots & 0 \\ 0 & -1 & \gamma & -1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \ldots & 0 & -1 & \gamma & -1 & 0 \\ 0 & \ldots & \ldots & 0 & -1 & \gamma & -1 \\ 0 & \cdots & \cdots & \cdots & 0 & -1 & \alpha \end{pmatrix} \in \mathbb{C}^{n\times n}, \tag{3.19}$$

where $\gamma = 4 - k^2h^2$ and α is the same as before.

And finally,

$$\dot{I}^h = \begin{pmatrix} -\frac{1}{2} & 0 & 0 & \dots & \cdots & \cdots & 0 \\ 0 & -1 & 0 & 0 & \cdots & \cdots & 0 \\ 0 & 0 & -1 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & 0 & 0 & -1 & 0 & 0 \\ 0 & \dots & \dots & 0 & 0 & -1 & 0 \\ 0 & \cdots & \cdots & \cdots & 0 & 0 & -\frac{1}{2} \end{pmatrix} \in \mathbb{R}^{n \times n}, \tag{3.20}$$

As already noted earlier, for the boundary value problem with Sommerfeld radiation conditions enforced on the boundary it is not possible to give a closed-form expression of the eigenvalues of the coefficient matrix (3.17).

For writing convenience the discretized domain G_h is from now on just denoted with G. Also we now denote the vectors $\mathbf{u}^h$ and $\mathbf{f}^h$ by **u** and **f**, respectively. Finally, the matrix A^h is denoted by A for convenience.
Thus the linear system of equation (3.12) is now written as

$$A\mathbf{u} = \mathbf{f}, \text{ on } G. \tag{3.21}$$

For MP-1, we find that the coefficient matrix A is indefinite real symmetric. For MP-2, which has a Sommerfeld radiation condition on the whole boundary, the coefficient matrix A is complex symmetric, but non-Hermitian.

For the 1D MP-1 problem (Equation (3.2)) we have already found that the exact eigenvalues and eigenfunctions are given by

$$\phi_n = \sin(n\pi x), \tag{3.22}$$

$$\lambda_n = n^2\pi^2 - k^2 \text{ , for } n = 1, 2, 3, \dots \text{, with } k^2 \neq n^2\pi^2 \tag{3.23}$$

The discrete eigenvalues and eigenfunctions for the same boundary value problem are

$$\begin{aligned} \widehat{\lambda}_l &= \frac{1}{h^2}\left[2 - 2\cos(l\pi h) - k^2h^2\right], \\ l &= 1, 2, 3, \dots, n \end{aligned} \tag{3.24}$$

and

$$\widehat{\phi}_i = \sin(i\pi\mathbf{x}), 1 \leq i \leq n \tag{3.25}$$

where $\mathbf{x} = [x_i], 1 \leq i \leq n$ represents the grid vector on the 1D version of G.

For the 2D MP-1 problem the exact eigenvalue and eigenfunction expressions are given by

$$\phi_{n,m} = \sin(n\pi x)\sin(m\pi y) \text{ , for } n, m = 1, 2, 3, \dots \tag{3.26}$$

$$\lambda_{n,m} = n^2\pi^2 + m^2\pi^2 - k^2 \text{ , for } n, m = 1, 2, 3, \dots \text{, with } k^2 \neq n^2\pi^2 + m^2\pi^2. \tag{3.27}$$

The related discrete eigenvalue are given by

$$\begin{aligned} \widehat{\lambda}_{i,j} &= \frac{1}{h^2}\left[4 - 2\cos(i\pi h) - 2\cos(j\pi h) - k^2h^2\right] \\ & i, j = 1, 2, 3, \dots, n \end{aligned} \tag{3.28}$$

and the discrete eigenfunctions by

$$\widehat{\phi}_{i,j} = \sin(i\pi\mathbf{x})\sin(j\pi\mathbf{y}), 1 \leq i, j \leq n \tag{3.29}$$

with **x** and **y** being the grid vectors from the 2D discrete domain G.

3.4 Direct Method

Most of the information from this section is from the Scientific Computing lecture notes by C. Vuik and D.J.P. Lahaye [34]. One way of solving the system of linear equations of equation (3.21)is by using direct solution methods. These methods are often the method of choice when dealing with problems of moderate size. Often,

direct solution methods are used as subdomain solvers when using domain decomposition methods or multigrid methods.

In this thesis we use the Gaussian elimination method with an LU-decomposition. The direct method consists of two stages. First, the coefficient matrix A from equation (3.21) is decomposed into an upper triangular matrix U and a lower triangular matrix L where the diagonal of L are all 1's. The decomposition is done in such a way that $A = LU$.
After the decomposition, the linear system of equation (3.21) can be easily solved using a forward linear solve followed by a backward linear solve. The linear system is solved with the following algorithm

$$LU\mathbf{u} = \mathbf{f} \implies L\mathbf{y} = \mathbf{f}, \text{ followed by } U\mathbf{u} = \mathbf{y}. \tag{3.30}$$

This gives the solution **u** of equation (3.21). The computational cost of the LU-decomposition is $\mathcal{O}(\frac{2}{3}n^3)$. Because of this, the speed of this method highly depends on the size of the problem. It is for this reason that iterative methods are used when dealing with large systems of linear equations. Currently, the state-of-the-art iterative methods are preconditioned Krylov subspace methods.

3.5 Krylov Subspace Methods

This section studies Krylov subspace methods as iterative methods for solving systems of linear equations. The reason for using a Krylov subspace iterative method to solve a linear system is because these iterative solution methods are faster then direct methods for large linear system. As with the previous section, a lot of information of this section comes from [34]. Krylov subspace methods are designed to avoid matrix-matrix operations, since these operations are computationally costly. Instead, Krylov subspace methods rely on multiplying vectors by the matrix. For using Krylov subspace iterative methods, the sequence of iterations of is denoted by

$$\{\mathbf{u}^k\}_{k\geq 0} \text{ where } \mathbf{u}^k \to \mathbf{u} \text{ for } k \to \infty \tag{3.31}$$

For the Richardson iterative method, the iterative scheme can be written in terms of residuals $\mathbf{r}^k := \mathbf{f} - A\mathbf{u}^k$ as

$$\mathbf{u}^{k+1} = \mathbf{u}^k + M^{-1}\mathbf{r}^k,$$

with the initial guess $\mathbf{u}^0$ given, and where $A = M - N$ with M assumed to exist and be non-singular. With no additional information, it is recommended that the initial guess is zero [28]. Writing out the first few steps of the recursion shows that

$$\mathbf{u}^k \in \mathbf{u}^0 + \text{span}\left(M^{-1}\mathbf{r}^0, M^{-1}A(M^{-1}\mathbf{r}^0), ..., (M^{-1}A)^{k-1}(M^{-1}\mathbf{r}^0)\right)$$

The Krylov-space of dimension k of matrix A with initial residual r^0 is then defined as

$$K^k(A; \mathbf{r}^0) := \text{span}\left(\mathbf{r}^0, A\mathbf{r}^0, ..., A^{k-1}\mathbf{r}^0\right)$$

The most well known Krylov subspace iterative methods are CG(Conjugate Gradient), GMRES(generalized minimum residual) and Bi-CGSTAB(biconjugate gradient stabilized). These methods all have their requirements when they can be used and when they are optimal to use.

- CG: the matrix A should be symmetric positive definite (SPD). This method is one of the best iterative methods, but is limited due to its requirements.
- GMRES type methods: These iterative method can be used for general matrices. These methods have long recurrences, but they do have some optimality properties
- Bi-CG type methods: Applicable for general matrices. The methods have short recurrences, but do not have an optimality property.

Because the matrix A will not be SPD for the Helmholtz problem, the CG method can not be used and will therefore also not be discussed any further.

3.5.1 Bi-CGSTAB

The Bi-CG Stabilized algorithms was introduced in [9]. The method is a Krylov subspace method that can be used for general matrices. The precondtioned algorithm of the Bi-CGSTAB method can be found below. More details about preconditioned linear systems can be found in the following section. The information about Bi-CGSTAB given in this subsection is from [34]. More details about the method can also be found in [34].

Algorithm 1: *Preconditioned Bi-CGSTAB Algorithm*

$\mathbf{u}^0$ is an initial guess; $\mathbf{r} = \mathbf{f} - A\mathbf{u}^0$;
$\bar{\mathbf{r}}^0$ is an arbitrary vetor, such that $(\bar{\mathbf{r}}^0, \mathbf{r}^0) \neq 0$, e.g. for $\bar{\mathbf{r}}^0 = \mathbf{r}^0$;
$\rho_{-1} = \alpha_{-1} = \omega_{-1} = 1$; $\mathbf{v}^{-1} = \mathbf{p}^{-1} = 0$;
for $i = 0, 1, 2, ...$ **do**
  $\rho_{\mathrm{i}} = (\bar{\mathbf{r}}^0)^T \mathbf{r}^{\mathrm{i}}; \beta_{\mathrm{i}-1} = (\rho_{\mathrm{i}}/\rho_{\mathrm{i}-1})(\alpha_{\mathrm{i}-1}/\omega_{\mathrm{i}-1})$;
  $\mathbf{p}^{\mathrm{i}} = \mathbf{r}^{\mathrm{i}} + \beta_{\mathrm{i}-1}(\mathbf{p}^{\mathrm{i}-1} - \omega_{\mathrm{i}-1}\mathbf{v}^{\mathrm{i}-1})$;
  $\hat{\mathbf{p}} = \mathbf{M}^{-1}\mathbf{p}^{\mathrm{i}}$;
  $\mathbf{v}^{\mathrm{i}} = \mathrm{A}\hat{\mathbf{p}}$;
  $\alpha_{\mathrm{i}} = \rho_{\mathrm{i}} / \left(\bar{\mathbf{r}}^0\right)^{\mathrm{T}} \mathbf{v}^{\mathrm{i}}$;
  $\mathbf{s} = \mathbf{r}^{\mathrm{i}} - \alpha_{\mathrm{i}}\mathbf{v}^{\mathrm{i}}$
  if $\|\mathbf{s}\| < \mathcal{O}^n$ where $n > 0$ is small, **then**
    $\mathbf{u}^{\mathrm{i}+1} = \mathbf{u}^{\mathrm{i}} + \alpha_{\mathrm{i}}\hat{\mathbf{p}}$;
  end if
  $\mathbf{z} = \mathrm{M}^{-1}\mathbf{s}$;
  $\mathbf{t} = \mathrm{A}\mathbf{z}$;
  $\omega_{\mathrm{i}} = \mathbf{t}^{\top}\mathbf{s}/\mathbf{t}^{\top}\mathbf{t}$;
  $\mathbf{u}^{\mathrm{i}+1} = \mathbf{u}^{\mathrm{i}} + \alpha_{\mathrm{i}}\hat{\mathbf{p}} + \omega_{\mathrm{i}}\mathbf{z}$;
  if $\mathbf{u}^{\mathrm{i}+1} < \mathcal{O}^n$ where $n > 0$ is small, **then**
    $\mathbf{r}^{\mathrm{i}+1} = \mathbf{s} - \omega_{\mathrm{i}}\mathbf{t}$;
  end if
end for

3.5.2 GMRES

The General Minimal RESidual (GMRES) method is based on the MINRES method. The details about the GMRES method come from [34].
Instead of using the Lanczos method, the GMRES algorithm uses the Arnoldi method. The GMRES algorithm can be found below.

Algorithm 2: *GMRES*

Choose $\mathbf{u}^0$ and compute $\mathbf{r}^0 = \mathbf{f} - A\mathbf{u}^0$ and $\mathbf{v}^1 = \mathbf{r}^0 / \left\|\mathbf{r}^0\right\|_2$,
for $j = 1, \ldots, k$ **do**
  $\mathbf{v}^{\mathrm{j}+1} = A\mathbf{v}^{\mathrm{j}}$;
  for $i = 1, \ldots, j$ **do**
    $h_{\mathrm{ij}} := \left(\mathbf{v}^{\mathrm{j}+1}\right)^{\top} \mathbf{v}^{\mathrm{i}}, \mathbf{v}^{\mathrm{j}+1} := \mathbf{v}^{\mathrm{j}+1} - h_{\mathrm{ij}}\mathbf{v}^{\mathrm{i}}$,
  end for
  $h_{\mathrm{j}+1,\mathrm{j}} := \left\|\mathbf{v}^{\mathrm{j}+1}\right\|_2, \mathbf{v}^{\mathrm{j}+1} := \mathbf{v}^{\mathrm{j}+1}/h_{\mathrm{j}+1,\mathrm{j}}$
end for
The entries of upper $k+1 \times k$ Hessenberg matrix $\bar{H}_k$ are the scalars h_{ij}.

3.6 Preconditioning

Typically, iterative methods are combined with preconditions to increase the speed of convergence. The speed of convergence of iterative methods often depends of the condition number κ of the system matrix A. If the eigenvalues are not well clustered, the condition number will be large which would cause slow convergence. Preconditioning is used to cluster the eigenvalues more favorably and therefore speed up the iterative method. In order to understand the following to section, knowledge of the reader about basic iterative methods is necessary because these are the building blocks preconditioning in a Krylov subspace context.

The preconditioner is a matrix M which is similar to the matrix A. By using a preconditioner the linear system

of Equation (3.21) is transformed into the linear system

$$M^{-1}A\mathbf{u} = M^{-1}\mathbf{f}. \tag{3.32}$$

M^{-1} has the following requirements:

- the eigenvalues of $M^{-1}A$ needs to be clustered more favourably then the eigenvalues of A,
- computing $M^{-1}\mathbf{f}$ should be low cost.

When the matrix A is SPD the condition number is given by

$$\kappa(A) = \frac{\lambda_{max}(A)}{\lambda_{min}(A)}. \tag{3.33}$$

If the matrix is not SPD, then the condition number is more difficult to define. As noted earlier, when the Sommerfeld radiation condition is used the coefficient matrix becomes complex symmetric, but non-Hermitian, which introduces difficulties. The matrix A being non-normal results the condition number of the eigenvector matrix being larger than one. For GMRES it has been shown that if the coefficient matrix A is non-normal but diagonalizable the convergence of A behaves the same as if A was normal. This means that the convergence rate of non-normal A for GMRES does not explicitly depend on its eigenvalues, but the eigenvalues do influence the rate of convergence significantly.

3.6.1 Complex Shifted Laplacian Preconditioner

One of the preconditioners studied for the Helmholtz problem is the Complex Shifted Laplacian Preconditioner (CSLP) [22, 23]. The CSLP is defined as

$$M_{CSLP} := -\nabla_h^2 - (\beta_1 + i\beta_2)k^2 I_h = -\Delta_h - (\beta_1 + i\beta_2)k^2 I_h, \quad \beta_1, \beta_2 \in [0, 1], \tag{3.34}$$

where Δ_h is the discrete Laplacian operator and i is the imaginary unit. β_1 and β_2 represent the real and complex shift in the preconditioner, where the preconditioner is optimal for $(\beta_1, \beta_2) = (1, 0.5)$ [24].
Using the CSLP greatly benefits the convergence behavior of the iterative method. The CSLP clusters the eigenvalues on the complex plane on a unit circle with the left side of the circle being the origin, if $\beta_1 = 1$ [28]. When the wave number is small only some eigenvalues lay around the origin, but when the wave number increase the number of eigenvalues close to 0 increases. Consequentially, this makes the iterative solver with the CSLP not numerically scalable since the smallest eigenvalue keeps becoming smaller, which causes the number of iterations needed for convergence to keep increasing.
The matrix M_{CSLP} can be approximately or exactly inverted. In [22, 23, 42] one multigrid iteration is used to approximately invert the preconditioning matrix.
Specifically, one $V(1,1)$ multigrid iteration is often used to approximately invert the preconditioning matrix. $V(1,1)$ means that the multigrid method uses a V type cycle with 1 pre-smoothing step and 1 post-smoothing step.
It should be noted that when $\beta_1 = 1$, we find that the smaller β_2 the fewer eigenvalues are close to the origin. And the fewer eigenvalues close to the origin is favorable for the convergence of the preconditioned Krylov method. But, also note that the smaller β_2 the harder it is to do an approximate inversion of the preconditioner using multigrid, i.e. the higher β_2, the longer it takes for the multigrid method to converge and find the approximate inverse of the preconditioning matrix [38, 28].

3.6.2 Deflation

Deflation preconditioning is a method of removing the smallest eigenvalues. The removing is done by deflating the small eigenvalues all the way to 0. Eigenvalues that are 0 do not harm the convergence behavior and are ignored when computing the condition number of the preconditioned matrix $\hat{A}$.

Details about the deflation method are given in the case of having a 1D problem. Assume we have the one dimensional linear system of equations $A\mathbf{u} = \mathbf{f}$, with $A \in \mathbb{R}^{n\times n}$. In order to describe the deflation method the projection P_D is defined as

$$P_D = I - P = I - AZ\left(Z^T AZ\right)^{-1} Z^T, \quad Z \in \mathbb{R}^{n\times r}. \tag{3.35}$$

In the equation above I is the identity matrix, and the column space of the matrix Z is the deflation subspace. The deflation subspace is the space that is to be projected out of the residual. Furthermore, we assume that Z

has rank r and $r \ll (n)$. We also define the matrix $E \in \mathbb{R}^{r \times r}$ where $E = Z^T AZ$. The matrix P is a projection which projects an input vector $\boldsymbol{\nu} \in \mathbb{R}^n$ to the deflation subspace. But we are not interested in the deflation subspace, we are interested in the null space of the deflation subspace. P_D projects on the null space of P. Thus P_D projects on the null space of the deflation subspace.We know that another projector is defined as

$$P_D^T = I - ZE^{-1}Z^T A. \tag{3.36}$$

The solution **u** of equation (3.21) can be written as

$$\mathbf{u} = (I - P_D^T)\mathbf{u} + P_D^T\mathbf{u}. \tag{3.37}$$

Only $P_D^T\mathbf{u}$ needs to be computed, since $\left(I - P_D^T\right)\mathbf{u}$ is computed very easily as

$$\left(I - P_D^T\right)\mathbf{u} = ZE^{-1}Z^T A\mathbf{u} = ZE^{-1}Z^T\mathbf{f}.$$

We know that $P_D A = AP_D^T$, thus the deflated system is given by

$$P_D A\tilde{\mathbf{u}} = P_D\mathbf{f} \tag{3.38}$$

Using a Krylov subspace iterative method the solution $\tilde{\mathbf{u}}$ can now be found. Multiplying $\tilde{\mathbf{u}}$ by P_D^T and adding it to equation (3.37) gives the solution to the linear system of equation (3.21).
Note that the null space never enters the iterative method. Therefore, the zero eigenvalues have no influence on the convergence of the iterative method. This is the reason for using the deflation method.

Applying deflation preconditioning comes with some difficulties. Deflation preconditioning is often not used in combination with the GMRES method, since the deflation method requires the original system matrix to be positive semi definite and self-adjoint. For the Helmholtz problem these requirements do not hold. Because the requirements do not hold it could be that the projections are ill-defined. Therefore, conditions for the matrix A and AZ can be given such that deflation is possible when the system matrix A is not positive semi definite and self-adjoint [31]. The following theorem is given in [31]

Theorem 1 (Deflated GMRES)**.** *Let $A\boldsymbol{u} = \boldsymbol{f}$ be a linear system with A non-singular, $A \in \mathbb{C}^{n\times n}$, $b \in \mathbb{C}^n$ and Z a subspace of $\mathbb{C}^{n\times n}$ with dimension $\dim(Z) = r < n$. Furthermore, let $\theta_{Z,AZ} < \frac{\pi}{2}$, where $\theta_{Z,AZ}$ denotes the principle angle between the subspaces Z and AZ. Then the project matrix $P_D := P_D\boldsymbol{u} = (I - P)\boldsymbol{u} = \boldsymbol{u} - AZ\langle Z, AZ\rangle^{-1}\langle Z, \boldsymbol{u}\rangle$, $\boldsymbol{u} \in \mathbb{C}^n$ is well-defined. Moreover, for all initial guesses $\boldsymbol{u}_0$, the GMRES method applied to the deflated system is well-defined.*

Proof. The proof of this theorem can be found in [31], in section 3.3 theorem 3.9 □

To conclude, all that is needed in order to apply the deflated GMRES to the projection matrix P_D is that the original system matrix A is non singular and that the principle angle between the subspaces Z and AZ is less than $\frac{\pi}{2}$. This condition holds for all the subspaces used in this thesis.
Because this theorem does not apply to Bi-CGSTAB Krylov method, and because there is no equivalent theorem for the Bi-CGSTAB, the Bi-CGSTAB iterative method will not be used or discussed any further. The application of deflation preconditioning and Schwarz domain decomposition is essential in the work of this study. Therefore, this study will continue by using the GMRES Krylov method as the basis of its iterative solvers.

With the general background information about the deflation given, it is now possible to choose the deflation vectors as the columns of $Z \in \mathbb{C}^{(n)\times r}$ for the Helmholtz problem. Many options are available for Z and the choice of the deflation vectors is crucial for the success of the deflation method.
The goal of deflation is to remove the eigenvalues corresponding to the r smallest eigenvalues from the solution subspace. It is therefore natural to pick the eigenvectors corresponding to these r smallest eigenvalues as columns of Z. Choosing the matrix Z as such, results in the r smallest eigenvalues becoming zero. Even though this choice of column of Z is effective, it is also expensive. Computing the eigenvectors corresponding to the r smallest eigenvalues is computationally hard to do for large matrices.
Instead of computing the eigenvectors exactly, it is also possible to compute approximations of the eigenvectors corresponding to the r smallest eigenvalues. One possible approximation of the eigenvectors are the constant deflation vectors [4]. The domain Ω is divided into subdomains and the eigenvectors are approximated with constant deflation vectors in each subdomain. Each column of the matrix Z will then have ones at the grid index of the corresponding subdomain and zero everywhere else.
Another possibility is to use linear deflation vectors. These require more computational work compared to the constant deflation vectors, but they approximate the eigenvectors of the subdomains better.

Besides eigenvalue-based deflation, it is also possible to construct physics-based deflation vectors or algebraic deflation vectors. Only eigenvalue-based deflation is discussed in this report.
In the following chapter domain decomposition as a preconditioner is discussed. Deflation and domain decomposition are methods with a lot of similarities.

3.6.3 Deflation-based Preconditioner

In [42] several deflation preconditions for the Helmholtz problem are given and tested numerically. The goal in the article is to construct a preconditioner that is wave number independent. The preconditioners in the article mostly consist of a CSLP in combination with deflation using a coarse correction operator as the deflation matrix Z from equation (3.35). This means that $Z = I_{2h}^h$. Several different definition for I_{2h}^h varying in order and if a weight is included in the coarse correction operator are given in [42].
Combining Deflation and CSLP preconditioning results in having to solve the following linear system

$$M_{CSLP}^{-1} P_D A \mathbf{u} = M_{CSLP}^{-1} P_D \mathbf{f}. \tag{3.39}$$

When using a standard linear grid interpolation function the preconditioning scheme is referred to as a DEF-based preconditioner. The most promising preconditioner from the article, based on the numerical results, is given when the coarse correction operator in 1D is defined as

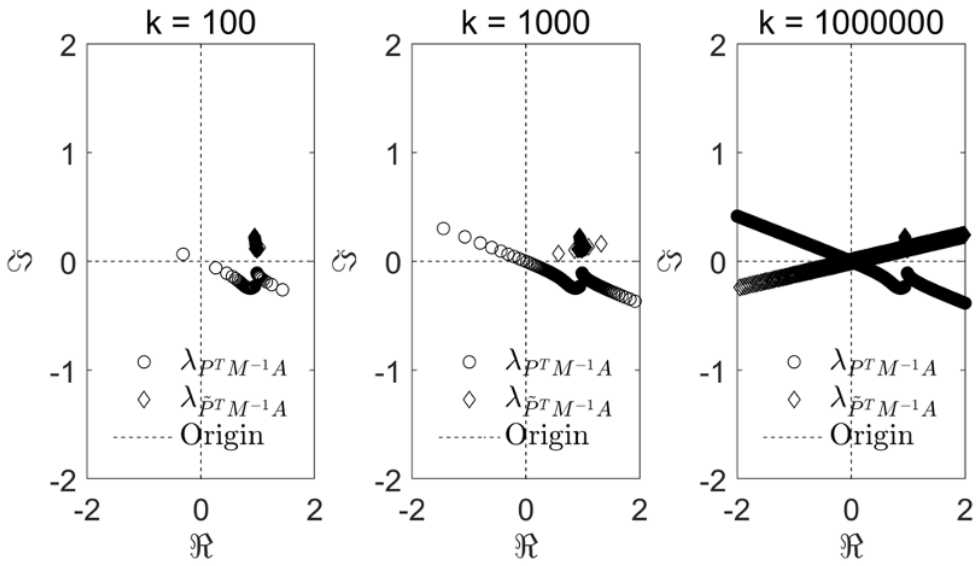
$$I_{2h}^h \left[u_{2h}\right]_i = \left\{ \begin{array}{cc} \left(\frac{1}{8}\left[u_{2h}\right]_{(i-2)/2} + \left(\frac{3}{4} - \varepsilon\right)\left[u_{2h}\right]_{(i)/2} + \frac{1}{8}\left[u_{2h}\right]_{(i+2)/2}\right) & \text{if } i \text{ is even,} \\ \frac{1}{2}\left(\left[u_{2h}\right]_{(i-1)/2} + \left[u_{2h}\right]_{(i+1)/2}\right) & \text{if } i \text{ is odd} \end{array} \right) \tag{3.40}$$

for $i = 1, ..., n$ and $\varepsilon > 0$. The definition for ε is given by

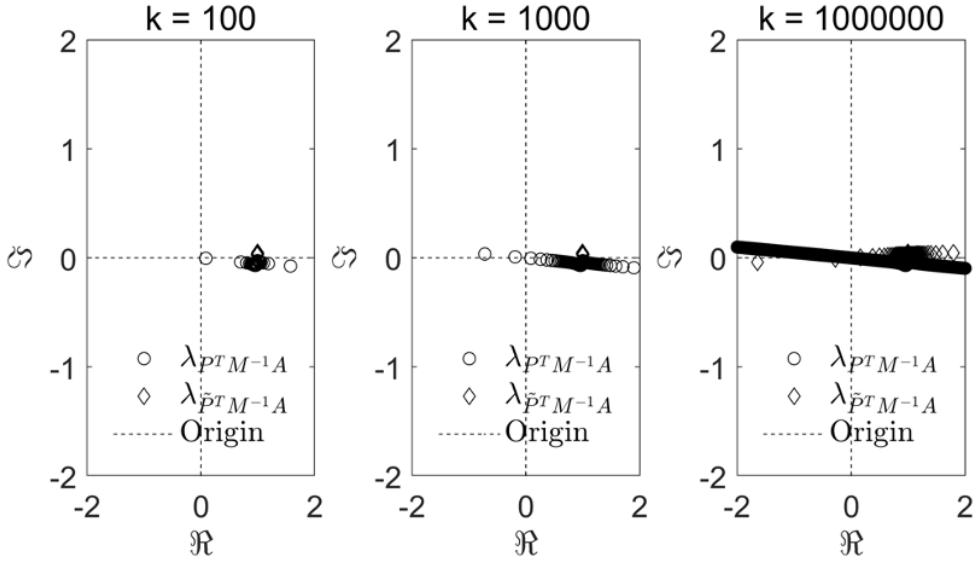
$$\varepsilon = 0.75 - \left(\cos(l\pi h) - \frac{1}{4}\cos(2l\pi h)\right). \tag{3.41}$$

The method of preconditioning which uses equation (3.40) is referred to as the the Adapted Preconditioned DEF scheme (APD). This method uses higher-order interpolation polynomials for the coarse correction operator, which is a quadratic approximation using the rational Bézier curve. Most often equation (3.41) is used to compute the optimal value for ε, but numerical tests are also performed using $\varepsilon = 0$.
After some numerical testing the article shows close to wave number independent convergence using the APD Helmholtz solver.

In order to keep track of the DEF-based scheme and the APD scheme we introduce that notation $\sim$ for P_D (or simply P) of the APD scheme. In Figure 4.3 the real and imaginary eigenvalues of the two deflation preconditioners (DEF and APD) are plotted for different wave numbers and different values of κ. One of the things we can conclude from the figures is that for very large wave numbers both preconditioner still have problems. This can seen since both preconditioned linear systems have eigenvalues that are not well clustered in the right column plots. Additionally, for the middle and left column we can see that the APD preconditioner performs better than the DEF precondtioner, since the eigenvalue clustering of the APD preconditioner is more favourable.



(a) $\kappa = 0.625$



(b) $\kappa = 0.3125$

Figure 3.2: The eigenvalues of the linear system with APD scheme indicated by $\tilde{P}M^{-1}A$ (diamond marker) and of the linear system with DEF-based scheme indicated by $PM^{-1}A$ (dot marker). In the top row $\kappa = 0.625$ and in the bottom row $\kappa = 0.3125$

Building on top of the APD scheme findings, a multilevel deflation preconditioner is constructed in [43]. The original APD is called a two-level deflation preconditioner, since it only uses a fine and a coarse grid. By introducing more layers of grids, the preconditioner becomes a multilevel deflation preconditioner.
It is important to note that instead of using a multigrid method on each level this article uses an exact solve on the coarsest level or when n is lower then a certain threshold and a few-GMRES iterations for the other levels. Earlier we said that due to the multigrid method β_2 had to be kept large, but when it is possible to use $\beta_2 = 1/k$, which provides optimal convergence when using GMRES as an iterative method and a direct method [36].
Using the multilevel deflation preconditioner the article finds by numerical experiments that the solver is almost numerically scalable and efficient. When Sommerfeld radiation conditions are used in the numerical experiments the article finds wave number independence for the number of iterations needed for convergence. It also find that the time complexity with respect to the number of grid nodes is somewhere between $\mathcal{O}(n^{1.5})$ and $\mathcal{O}(n)$, i.e. more or less linear time complexity.

4 | Domain Decomposition Methods as a Preconditioner

As seen in the previous chapter, it sometimes occurs that a linear system becomes difficult to be solved as a consequence of the problem size of the linear system. The basic idea of domain decomposition is that instead of solving the large linear system problem, the domain is split into subdomains and these subdomain problems are then solved. Due to the fact that domain decomposition methods solve many local subdomain problems, the method allows for parallel computing.
Domain decomposition methods (DDMs) are either iterative methods or preconditioners for iterative methods. For this research, we are interested in using DDMs as preconditioners to accelerate the convergence of Krylov subspace methods. This means that DDMs are used to define M^{-1}, which approximates the inverse of the coefficient matrix of the linear system of equations A^{-1}. More information about the field of domain decomposition methods is found in [11, 18, 21, 35].

The first example of a domain decomposition method is the alternating method by the mathematician H.A. Schwarz, which was published in 1870 [1]. Schwarz originally developed the method as a tool for proving the existence of a unique solution to the Laplace problem for an arbitrary domain [30]. In his method he used overlapping domains as in Figure 4.1. For this reason, overlapping domain decomposition methods are still referred to as Schwarz domain decomposition methods. P.L. Lions developed abstract theory on the alternating Schwarz method [52, 7, 53]. Domain decomposition methods are often divided into two classes. These classes are overlapping methods, also called Schwarz methods, and non-overlapping methods, sometimes called substructuring methods. In this thesis, only Schwarz domain decomposition methods are used.

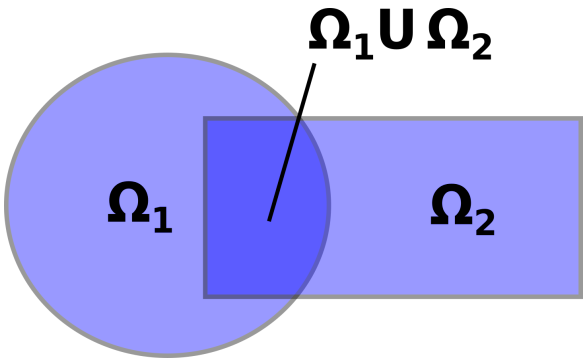


Figure 4.1: 2D domain with overlapping subdomains.

In the previous chapter, iterative Krylov subspace solution methods were introduced to solve a linear system of equations arising from a finite difference method discretization. There, we saw that the convergence behavior of a Krylov subspace method strongly depends on the condition number of the system matrix A. In order to improve the distribution of the eigenvalues, and therefore to improve the condition number, a preconditioner is used. Also in the previous chapter, the CSLP and deflation preconditioners were discussed as preconditioners to obtain a more favourable spectrum for the linear system.
In this chapter, domain decomposition methods as preconditioners are explored. This means that the different domain decomposition methods yield definitions for the preconditioner operator M^{-1}. Some Schwarz domain decomposition preconditioners are well-suited for parallel computing. Ultimately, we are interested in constructing a Helmholtz solver that is numerically scalable and efficient. Therefore, we will look at Schwarz preconditioners that can be implemented in parallel. A solver that allows for parallelization could result in the solver having a linear time complexity, but this is not investigated in this research.

This chapter is divided in the following way. In the first section, the Schwarz theory and one-level Schwarz preconditioners are discussed. After that, details about two-level Schwarz preconditioners are given. Next, different types of coarse spaces, which appear in two-level Schwarz preconditioners, are discussed. The order of the topics is chosen such that the topics naturally build on top of each other. In Section 4.4 details are given on using Schwarz preconditioners in a finite difference methods setting. Finally, preliminary results of the one-level preconditioners of this chapter can be found in the final section of this chapter.

4.1 One-Level Schwarz Domain Decomposition Methods

The most simple Schwarz domain decomposition methods are the one-level Schwarz domain decomposition methods. These methods decompose the domain into overlapping subdomains and solve the subdomain

problems instead of the global problem. One-level Schwarz domain decomposition preconditioners were first introduced in [2, 3].
It should be noted that the literature on domain decomposition methods is mainly written for finite element methods (FEM). Therefore, this chapter is also written for finite element methods.

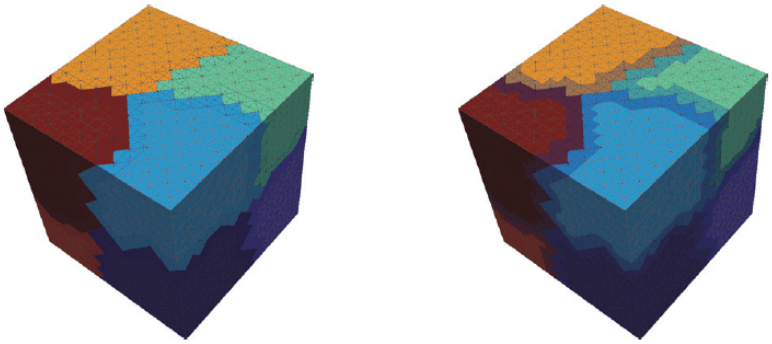


Figure 4.2: On the left a 3D mesh with non-overlapping subdomains $\{\Omega_i\}$ is given. On the right a 3D mesh with overlapping subdomains $\{\Omega_i'\}$ is given with $\delta = 1h$. Taken from [37]

Consider an elliptic partial differential equation with homogeneous Dirichlet boundary conditions on a polygonal domain $\Omega \subset \mathbb{R}^d$, with either $d = 2, 3$. Next, the domain is discretized leading to a triangulation $\mathcal{T}_h$ of Ω, with maximum mesh size h. A finite element approximation is applied to the problem which results in a sparse linear system

$$A\mathbf{u} = \mathbf{f}, \tag{4.1}$$

where **u** is the vector of which its coefficients are the discrete values of the approximate solution of the boundary value problem in the elements of $\mathcal{T}_h$. The domain Ω is partitioned into N non-overlapping subdomains, which results in the set of non-overlapping subdomains $\{\Omega_i\}_{i=1}^N$ with diameter H_i indicating the maximum diameter of the subdomains. For these subdomains non-overlapping local meshes $\mathcal{T}_i$ are given by the elements of $\mathcal{T}_h$ which belong to subdomain Ω_i.
The non-overlapping subdomains $\{\Omega_i\}$ can be extended by adding layers of elements in such a way that the boundary of the extend subdomain does not cut through any elements of the mesh $\mathcal{T}_h$. For the subdomains that adjacent to the boundary of Ω we can do the same thing, but we do not extend outside of Ω. This now gives rise to N overlapping subdomains Ω_i' with a maximum diameter of the subdomains H_i'. The local mesh $\mathcal{T}_i'$ of Ω_i' consist of the elements of $\mathcal{T}_h$ which belong to the overlapping subdomain Ω_i'. This holds for all the overlapping subdomains. Figure 4.2 illustrates non-overlapping and overlapping subdomains for a 3D mesh where the elements have a maximum mesh size of h. The space V is a continuous, piecewise linear finite element space on $\mathcal{T}_h$, where the elements vanish on the boundary $\partial\Omega$. For $\mathcal{T}_i'$ the finite element solution spaces V_i are defined by

$$V_i = \left\{u \in H_0^1(\Omega_i') \mid u_{|_K} \in \mathbb{P}_1, K \in \mathcal{T}_i'\right\}, \quad 1 \leq i \leq N, \tag{4.2}$$

where K indicates an element in a local overlapping mesh.
The natural extension operators are

$$R_i^T : V_i \to V, \tag{4.3}$$

sometimes also referred to as prolongation operators, and respective R_i operators are often called restriction operators. The space V is assumed to decompose as

$$V = \sum_{i=1}^{N} R_i^T V_i. \tag{4.4}$$

Note that the decomposition will not necessarily be a direct sum of subspaces. A component of V can be represented in terms of components of V_i, but its representation is not necessarily unique. This is a consequence of V_i being subspaces related to the overlapping subdomains. Also, V_i is not necessarily subspace of V, but $\{V_i\}$ are referred to as 'local spaces' or 'subspaces'.
Now, the local coefficient matrices of subdomains $\{\Omega_i\}$ and local space $\{V_i\}$ are given by

$$A_i = R_i A R_i^T, \quad \text{for } i = 1, ..., N. \tag{4.5}$$

The matrix A_i represents part of matrix A corresponding to the subdomain Ω_i. Equation (4.5) is for the case of using local exact solvers, which means that a direct solution method is used for solving the local subproblems $\{A_i^{-1}\}$. For the Helmholtz problem, it has also been proposed to use inexact solvers to solve the local problems [45]. Inexact solvers can be particularly useful for 3D high frequency Helmholtz problems for which the local subdomain problem might be too large for a direct solution method. The proposed inexact solution method in [45] is a two-level deflation preconditioner with flexible GMRES, which shows promising results.

We define B_i by

$$B_i = R_i^T A_i^{-1} R_i, \quad \text{for } i = 1, ..., N. \tag{4.6}$$

Next, we define project-like operators

$$P_i = B_i A, \quad \text{for } i = 1, ..., N, \tag{4.7}$$

which are used to define the Schwarz operators. One of the properties of the P_i operators is that an operator is a projection, i.e. $P_i^2 = P_i$.
Now we can define a number of different Schwarz preconditioners.

4.1.1 Multiplicative Schwarz Method

The use of alternating Schwarz methods on boundary value problems has become to be known as the multiplicative Schwarz (MS) method. We can now define the MS preconditioner as

$$M_{\text{MS}}^{-1} = I - (I - B_N)(I - B_{N-1})...(I - B_1), \tag{4.8}$$

with I being the appropriately sized identity matrix and $\{B_i\}_{i=1}^N$ are as defined in Equation (4.6) [11].

The MS method requires sequential computations of the subdomain problems. Using the MS method for the one-level part of Schwarz preconditioners is omitted from this research, since we are interested in developing preconditioners that allow for parallelization. Without parallel computing, the MS method is faster than the additive Schwarz method as a preconditioner, since the convergence rate is better [35]. When looking at two-level or multilevel Schwarz DDMs, the multiplicative coupling might again be useful.

4.1.2 Additive Schwarz Method

Summing the operators $\{B_i\}$ results in an additive Schwarz (AS) preconditioner. Besides just the AS method, some other variants of the AS method are also given in the next subsection to construct other Schwarz preconditioners. These other Schwarz methods are called the restricted additive Schwarz (RAS) method [12, 44] and the scaled additive Schwarz (SAS) method [44].
The one-level AS method was first introduced in [5] and [8]. The preconditioner matrix for the one-level AS method is defined as

$$M_{\text{AS1}}^{-1} = \sum_{i=1}^{N} B_i = \sum_{i=1}^{N} R_i^T A_i^{-1} R_i. \tag{4.9}$$

One of the advantages of the AS method is that it is suitable for parallel computing, because the local problems can all be solved independently of each other. Hence, the AS method has gained popularity in the last two decades.

An upper bound for the condition number of the matrix $M_{\text{AS1}}^{-1} A$ is given in [18, p. 95]. For the condition number it is found that

$$\kappa(M_{\text{AS1}}^{-1} A) \leq C(\frac{1}{\delta^2 H^2}), \tag{4.10}$$

where δ is the width of the overlap between two subdomains and C depends on N^c, which is the maximum number of overlapping subdomains an element can belong to. The rate of convergence of the one-level AS method combined with a Krylov method improves as that overlap of the subdomains increase up until some point. When the overlap becomes roughly equal to or larger than $H/2$ the rate of convergence often deteriorates again. This deterioration occurs due to N^c becoming larger, which increases C in Equation (4.10).

4.1.3 Restricted and Scaled Additive Schwarz Method

The restriction/extension operators in Equation (4.3) can be seen as the sum of two terms $R_i = R_i^0 + R_i^c$, where R_i^0 only includes the non-overlapping elements and R_i^c only includes the elements in the overlapping region [18]. Using this, Cai and Sarkis introduced a new AS preconditioner [12] called the restricted additive Schwarz (RAS) method, for which the preconditioner is defined as

$$M_{\mathrm{RAS1}}^{-1} = \sum_{i=1}^{N} \hat{B}_i = \sum_{i=1}^{N} (R_i^0)^T A_i^{-1} R_i. \tag{4.11}$$

The benefit of this new preconditioner is that when a parallel implementation is used for this preconditioner, the communication cost can be lowered. This is because computing $(R_i^0)u$ with $u \in V$ does not involve data exchange with neighbouring subdomains. Additionally, it often results in faster convergence, although the RAS preconditioner matrix is not symmetric anymore.

Due to the overlapping subdomains the following is not true $\sum_{i=1}^{N} R_i^T R_i = I$ with I being the appropriately sized identity matrix. By introducing diagonal D_i which satisfies

$$\sum_{i=1}^{N} R_i^T D_i R_i = I, \tag{4.12}$$

the double, or even more, counting of the elements in the overlapping regions is resolved [44]. Using this matrices D_i, the scaled additive Schwarz (SAS) preconditioner is defined as

$$M_{\mathrm{SAS1}}^{-1} = \sum_{i=1}^{N} \bar{B}_i = \sum_{i=1}^{N} R_i^T D_i A_i^{-1} R_i. \tag{4.13}$$

The benefit of using the SAS preconditioners is that it often needs fewer number of iterations to reach convergence, without having much extra computational cost compared with the AS Schwarz preconditioner of Equation (4.9). At the moment there seems to be no domain decomposition convergence theory for the one-level SAS operator. The matrix M_{SAS1}^{-1} is not symmetric anymore, which is why developing theory can be difficult.

4.2 Two-Level Schwarz Domain Decomposition Methods

It has been shown that the one-level Schwarz domain decomposition methods are not scalable with respect to the number of subdomains. Additionally, it is quite easy to see from (4.10) that, if H decreases due to the increase in the number of subdomains, the upper bound for the condition number increases for a fixed overlap δ. By using a two-level Schwarz domain decomposition method, which includes a solve of a coarse problem, this subdomain scalability problem can be avoided [35].
One-level overlapping Schwarz methods were originally extended to the two-level form in [5, 6, 8, 50] and more details on two-level Schwarz methods can be found in [11, 18, 21].

Two-level Schwarz methods are given in terms of partitions of the domain Ω into subdomains Ω_i, which are unions of finite elements and the subdomains have a diameter of order H. Additionally, a coarse shape-regular mesh $\mathcal{T}_{CH}$ with a maximum coarse element diameter of CH is introduced. For the purpose of explaining the two-level Schwarz methods, we assume that coarse mesh elements are given by the subdomain meshes $\mathcal{T}_i$, i.e. the coarse mesh has elements with diameter H ($CH = H$). We introduce the coarse finite element space V_0, which is defined as

$$V_0 = \left\{ u \in H_0^1(\Omega) \mid u_{|_K} \in \mathbb{P}_1, K \in \mathcal{T}_H \right\}, \tag{4.14}$$

and the coarse prolongation operator is given by

$$R_0^T : V_0 \to V. \tag{4.15}$$

There are different options for choosing the coarse operator R_0^T and the type of coarse space. More details on the definitions for the coarse level can be found in the next section.
By introducing a coarse space to the one-level AS preconditioner, the convergence rate of the algorithm can be made independent of the number of subdomains [15, 35]. Using the one-level AS preconditioner and the one-level SAS preconditioner from the previous section, two-level Schwarz preconditioners are constructed by adding a coarse problem to the one-level preconditioners.
Next, let us define preconditioner operators for different two-level preconditioners used in this thesis.

4.2.1 Two-Level Additive Schwarz Methods

The first of these two-level Schwarz preconditioning operators is the two-level AS operator, which is of the form

$$M_{\mathrm{AS2}}^{-1} = \sum_{i=0}^{N} B_i = R_0^T A_0^{-1} R_0 + \sum_{i=1}^{N} R_i^T A_i^{-1} R_i, \tag{4.16}$$

where B_0 corresponds to the coarse space V_0 and the remaining B_i are associated with the local problems on the subdomains from the set $\{\Omega_i\}_{i=1}^N$. B_0 is defined as

$$B_0 = R_0^T A_0^{-1} R_0, \tag{4.17}$$

and the matrix A_0 is called the coarse problem, which is given by $A_0 = R_0 A R_0^T$. In this thesis, both the local subproblems and the coarse problem will be solved using a direct solution method. Specifically, the Gaussian elimination method with LU-decomposition.

The convergence behavior of the two-level AS method can be analysed by estimating the condition number of $M_{\mathrm{AS2}}^{-1}A$. For the Laplace problem, when using equation (4.16) and a coarse operator as in equation (4.15), an upper bound for the condition number is given by

$$\kappa(M_{\mathrm{AS2}}^{-1}A) \leq C\left(1 + \frac{H}{\delta}\right). \tag{4.18}$$

The upper bound of the condition number and the proof of equation (4.18) are found in [21, Theorem 3.13, p. 69]. The constant C depends again on N^c. For the Laplace problem, the constant C is independent of the mesh size of $\mathcal{T}_H$, the mesh size of $\mathcal{T}_h$, and the overlap of the subdomains δ.
Equation (4.18) indicates that the two-level AS method can be scalable with respect to the number of subdomains. Take each subdomain to include a certain number of elements and a certain number of elements of overlap. The size each element depends on h. Now, the ratio of H/δ can be kept constant by adding more of the same subdomains. Adding subdomains with a fixed number of elements per subdomain and a fixed number of elements per overlap results in having to decrease the mesh size h, since the total number of elements increases. This shows that we can keep adding subdomains, with H/δ constant, while increasing to total problem size and keeping the upper bound of Equation (4.18) fixed. The two-level AS method having this subdomain scalability property is key for the development of a Helmholtz solver that is wave number independent and efficient, since the problem size has to increase with the wave number due to κ_h.

Another two-level Schwarz operator can be given using Equation (4.13). The two-level SAS preconditioning operator is defined as

$$M_{\mathrm{SAS2}}^{-1} = B_0 + \sum_{i=1}^{N} \bar{B}_i = R_0^T A_0^{-1} R_0 + \sum_{i=1}^{N} R_i^T D_i A_i^{-1} R_i. \tag{4.19}$$

Also for the two-level SAS method, there seems to be no literature on convergence theory of this method at the moment, which is not surprising since there is also nothing on the one-level SAS method.

4.2.2 Two-Level Hybrid Schwarz Methods

Up until here, all the Schwarz preconditioner, besides the one-level multiplicative preconditioner, are constructed by summing over the operators $\{B_i\}$. If a Schwarz method uses a computation of summing multiplying the operators $\{B_i\}$ it is called a hybrid Schwarz method. One of the possible two-level hybrid Schwarz (HS) preconditioners is given in [26]. In this section a HS method is given that is found in [44]. It is possible to implement deflation into two-level Schwarz operators [44]. The two-level HS preconditioner from [44] uses an 'adapted deflation technique' and is defined as

$$M_{\mathrm{HS2}}^{-1} = B_0 + \left(\sum_{i=1}^{N} R_i^T A_i^{-1} R_i\right)(I - AB_0), \tag{4.20}$$

where I is the identity matrix of appropriate size and $P_0 = (I - AB_0)$ is a projection. P_0 is similar to the project P_D from Equation 3.35. So P_0 projects onto the null space of AB_0.
Finally, the scaling of the SAS operators can also be added to the HS operator, resulting in the two-level scaled hybrid Schwarz (SHS) preconditioner, which is defined as

$$M_{\mathrm{SHS2}}^{-1} = B_0 + \left(\sum_{i=1}^{N} R_i^T D_i A_i^{-1} R_i\right)(I - AB_0), \tag{4.21}$$

with I being the identity matrix of appropriate size. For both the two-level HS and SHS operator, no domain decomposition convergence theory seem to be available.

With these two-level Schwarz domain decomposition preconditions, the only thing missing from them is a definition for the coarse operator R_0.

4.3 Coarse Spaces for Two-Level Schwarz Methods

Our goal in this section is to give details on types of coarse spaces that can be used in two-level Schwarz methods. Often, the poor convergence performance of the one-level Schwarz methods is caused by slow modes, which are the eigenvectors corresponding to the low frequency eigenvalues. The goal of a coarse space is to incorporate these slow modes into the coarse space.
The construction of a coarse space V_0 can be done is several ways. The most natural way is the grid coarse space method [44]. A coarse space that is based on a coarse mesh is called a 'grid coarse space'. The coarse mesh that makes most sense to use is $\mathcal{T}_H$. For the coarse operator R_0^T, we defining it to be an interpolation operator from the coarse mesh to the fine mesh. Consider the mesh $\mathcal{T}_H$ and corresponding finite element space $V_0 \subset V$. Let $I^h : V_0 \to V$ be the nodal interpolation operator and R_0^T be the corresponding matrix. In this thesis, only grid coarse spaces are used. The main ingredient missing for constructing the different two-level Schwarz preconditioners is this interpolation operator I^h, which will be defined in the next chapter.
Different options are available to define I^h. For example, linear interpolation or higher-order interpolation can be used. This thesis focuses on coarse interpolation such as I_h in [42]. Details about linear and Bézier coarse interpolation is found in Chapter 5.

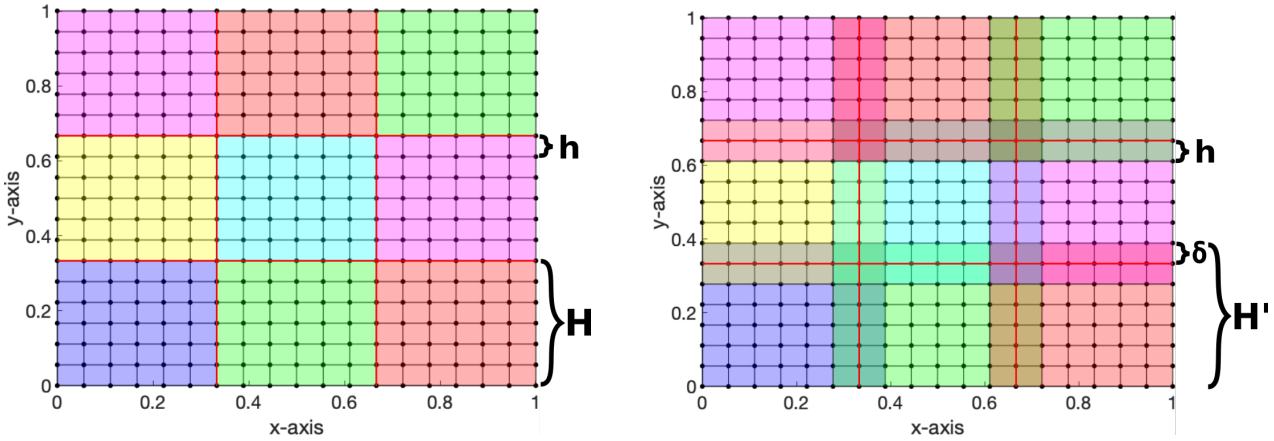
Another possible type of coarse space for the two-level Schwarz method is the 'spectral coarse space' type. In the construction of such a coarse space, spectral information from appropriate local eigenproblems is incorporated. The goal here is to incorporate the relevant spectral information in order to prevent slow convergence. The Dirichlet-to-Neumann (DtN) coarse space is a spectral coarse space based on local DtN eigenproblems on the subdomain interfaces. More details about this coarse space can also be found in [29] and [40]. The generalised eigenproblems in the overlap (GenEO) coarse space is another spectral coarse space. Using a GenEO-type coarse space is often more flexible and robust compared to a DtN coarse space [44]. GenEO-type coarse space are being constructed specifically for the Helmholtz problem with two-level Schwarz preconditioner solvers showing reasonable results for low wave numbers [44].

4.4 Schwarz Preconditioners for Finite Difference Discretizations

As mentioned at the beginning of this chapter, the literature on domain decomposition methods is mostly written from a finite elements method. In this thesis, we make use of the finite difference method, which means that some time should be taken to clearly transfer the ideas from this chapter to a finite difference method perspective.

From now on, all subdomains are quasi uniform square (2D problems) or cube (3D problems) overlapping subdomains. The subdomains are quasi uniform square or cube since sometimes not all subdomains are the same size, due to the subdomain on the boundary not being able to extend outside the domain, this also means they are sometimes not square. In Figure 4.3 we can see how the subdomain change from being uniform squares (**a**), to non uniform squares and rectangles (**b**).
The scenario with minimum overlapping subdomains Ω_i, the scenario in Figure 4.3 (**a**), is taken as the standard subdomain scenario. These subdomains are called 'standard subdomains' and they are from now on indicated by $\bar{\Omega}_i$. In the scenario of Figure 4.3 (**a**), the subdomain diameter of $\bar{\Omega}_i$ is H and we indicate the amount of overlap of the $\bar{\Omega}_i$ standard subdomains with $\delta = 0$. The overlap is not 0, since from Figure 4.3 (**a**) it can be seen that the standard subdomains share grid nodes on each interfaces of the standard subdomains that are not on the boundary of Ω ($\partial\bar{\Omega}_i \backslash \partial\Omega$). We can increase the amount of overlap of the standard subdomain $\bar{\Omega}_i$, resulting in $\delta > 0$ and the maximum subdomain diameter now being indicated by H'.
From now on, a standard subdomain $\bar{\Omega}_i$ with $\delta > 0$ is indicated by Ω_i and it has both a H' and a H diameter with often $H' = H + 2\delta$. Note that $\bar{\Omega}_i \subset \Omega_i \, \forall i \in \{1, 2, ..., N\}$. H always stays unchanged when δ increases, since it is the standard subdomain diameter. In Figure 4.3 (**b**) the overlap is increased to $\delta = 1$, since we have extended the subdomains with $1h$ in all possible directions. Additionally, sometimes $H' \neq H + 2\delta$, due to the subdomain being on the boundary of Ω. This is the case for the indicated H' in Figure 4.3 (**b**). In Figure 4.3

(**b**) it can be seen that for the subdomain in the middle of Ω the statement $H' = H + 2\delta$ does hold.



(a) 2D grid domain image partitioned in 9 subdomains with overlap $\delta = 0$, $\bar{n}^l = n_i^l = 49$ and $n = (n^*)^2 = 19^2 = 361$.

(b) 2D grid domain image partitioned in 9 subdomains with overlap $\delta = 1$, $\bar{n}^l = 49$, $n_i^l = 64$ or $n_i^l = 72$ or $n_i^l = 81$ and $n = (n^*)^2 = 19^2 = 361$.

Figure 4.3: Grid images of subdomains with $\delta = 0$ and $\delta = 1$. The subdomains are indicated by the coloured areas. Additionally, red lines mark the interfaces of the standard subdomains which are not on $\partial\Omega$. h indicates the fine mesh size, H indicates standard subdomain diameter, H' indicates the subdomain diameter with $\delta > 0$.

From now on, all subdomains are quasi uniform square (for 2D problems) or cube (for 3D problems) overlapping subdomains. The subdomains are quasi uniform square or cube since sometimes not all subdomains are the same size, due to the subdomain on the boundary not being able to extend outside the domain, this also means they are sometimes not square. In Figure 4.3 we can see how the subdomain change from being uniform squares (**a**), to non uniform squares and rectangles (**b**).
The scenario with minimum overlapping subdomains Ω_i, the scenario in Figure 4.3 (**a**), is taken as the standard subdomain scenario. These subdomains are called 'standard subdomains' and they are from now on indicated by $\bar{\Omega}_i$. In the scenario of Figure 4.3 (**a**), the subdomain diameter of $\bar{\Omega}_i$ is H and we indicate the amount of overlap of the $\bar{\Omega}_i$ standard subdomains with $\delta = 0$. The overlap is not 0, since from Figure 4.3 (**a**) it can be seen that the standard subdomains share grid nodes on each interfaces of the standard subdomains that are not on the boundary of Ω ($\partial\bar{\Omega}_i \backslash \partial\Omega$). We can increase the amount of overlap of the standard subdomain $\bar{\Omega}_i$, resulting in $\delta > 0$ and the maximum subdomain diameter now being indicated by H'.
From now on, a standard subdomain $\bar{\Omega}_i$ with $\delta > 0$ is indicated by Ω_i and it has both a H' and a H diameter with often $H' = H + 2\delta$. Note that $\bar{\Omega}_i \subset \Omega_i \, \forall i \in \{1, 2, ..., N\}$. H always stays unchanged when δ increases, since it is the standard subdomain diameter. In Figure 4.3 (**b**) the overlap is increased to $\delta = 1$, since we have extended the subdomains with $1h$ in all possible directions. Additionally, sometimes $H' \neq H + 2\delta$, due to the subdomain being on the boundary of Ω. This is the case for the indicated H' in Figure 4.3 (**b**). In Figure 4.3 (**b**) it can be seen that for the subdomain in the middle of Ω the statement $H' = H + 2\delta$ does hold.

In the finite element method setting, we had a domain Ω with mesh $\mathcal{T}_h$ being the finite element mesh of Ω. Instead of having a finite element mesh, we now use a uniform grid G_h, as given in Figure 4.3, with values at the grid nodes of the mesh and mesh length $h = \frac{1}{n^*-1}$ with n^* being the number of grid nodes in one dimension. The same goes for the subdomains Ω_i for which the grids are indicated by G_i. The mesh size of the G_i grids are still h and the grid includes all the grid nodes from G_h which belong to the subdomain Ω_i. A standard subdomain grid of a standard subdomain $\bar{\Omega}_i$ is indicated by $\bar{G}_i$. The number of grid nodes for a standard subdomain grid $\bar{G}_i$ is indicated by n^l. No subscript is required for n^l, since the standard subdomain grids all always have the same total number of grid nodes. The number of grid nodes of a subdomain grid G_i depends on in how many directions the standard subdomain can extend the overlap and on the amount of overlap. We indicated the total number of grid nodes of a subdomain G_i by n_i^l, also called the 'local subdomain size'. For example, in 4.3 (**b**) all the subdomain grids G_i have $n^l = n_i^l = 36$, $n^* = 19$ and $n = 361$.
The coarse mesh is now indicated by G_{CH}. We do not assume that the subdomain size is the same as the coarse mesh size anymore, which is why the CH notation for the coarse mesh size returns. Additionally, the total number of grid nodes in a coarse mesh is indicated by n^c, which is also called the 'coarse domain size'. The size of the matrices of the fine, subdomain and coarse problems are

$$A \in \mathbb{R}^{n \times n}, \; A_i \in \mathbb{R}^{n_i^l \times n_i^l}, \; A_0 \in \mathbb{R}^{n^c \times n^c}, \tag{4.22}$$

and the operators R_i^T,R_i,R_0^T,R_0 necessary for the Schwarz preconditioners have the sizes

$$R_i^T \in \mathbb{R}^{n \times n_i^l}, \; R_i \in \mathbb{R}^{n_i^l \times n}, \; R_0^T \in \mathbb{R}^{n \times n^c} \; R_0 \in \mathbb{R}^{n^c \times n}. \tag{4.23}$$

For κ_h of Equation (3.1) we need the number of grid nodes in one dimension of G_h, which is denoted by n^*.

4.5 Preliminary Results

In Chapters 6 and 7, numerical results using two-level Schwarz preconditioner are discussed. It is useful to also get an idea of the numerical results of the one-level Schwarz preconditioners on MP-3 from Section 3.2. These results are given in this section.

In Table 4.1, the results of the one-level Schwarz preconditioners for MP-3 are given. Having $H = 4h$ means that the standard subdomain has a fixed number of grid nodes per standard subdomain. In Table 4.1, the problem size increases, which results in the number of subdomain having to increase. GMRES uses a relative tolerance of $1e-7$. The results in Table 4.1 nicely show what is expected. For comparison, the number of iterations required to reach convergence for GMRES without a one-level Schwarz preconditioner is given.
The amount of overlap is fixed in Table 4.1, therefore the condition number of Equation (4.10) increase. This increasing condition number coincides with the increasing number of iterations when n^* increases. It is clear that using one-level Schwarz preconditioners does not result in problem size scalability. The expectation that SAS precondtioners perform slight better is met.

n^*	NoPrecon	AS_{4h} (Subdoms)	AS_{8h} (Subdoms)	SAS_{4h} (Subdoms)	SAS_{8h} (Subdoms)
33	111	32 (8)	26 (4)	30 (8)	24 (4)
65	218	60 (16)	46 (8)	58 (16)	43 (8)
97	x	88 (24)	66 (12)	85 (24)	63 (12)
129	x	117 (32)	86 (16)	112 (32)	82 (16)
161	x	145 (40)	107 (20)	139 (40)	102 (20)

Table 4.1: Number of iterations required to reach convergence with the one-level AS and SAS preconditioners on MP-3. In brackets behind the NOI results, the number of subdomains in one dimension are given. Additionally, the minimum amount of overlap is used, $\delta = 0$.

5 | Coarse Grid Interpolation

In this chapter, definitions for the operators R_0^T and R_0 are given. Inspired by the Helmholtz solver developed in [42] we construct a coarse operator using higher-order interpolation with quadratic rational Bézier curves. First, details about the construction of a first-order interpolation operator are given, followed by details on the construction of a higher-order Bézier interpolation operator at the end of this chapter.

This chapter only gives details on the coarse grid interpolation operators with a one dimension coarse grid size (CH) of $2h$, such as the coarse grid illustrated in Figure 5.1. Also, the details interpolation operators are introduced for a one dimensional domain. The interpolation operators for the higher dimensional domains follow in a tensor product way. See Appendix A for details on a one dimensional coarse grid with $CH = 4h$ and a higher-order Bézier interpolation scheme.

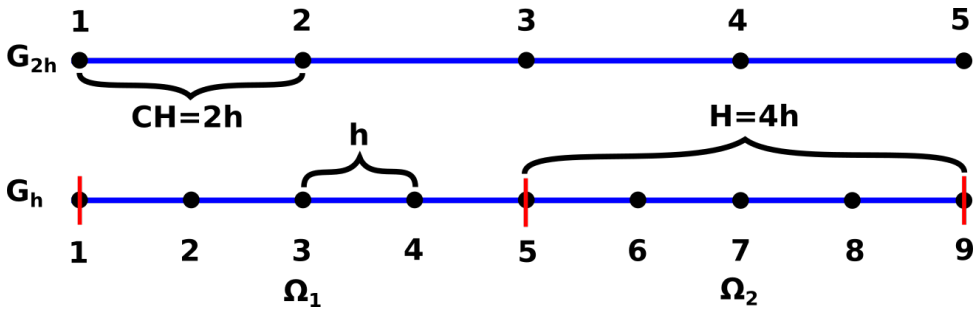


Figure 5.1: Fine and coarse one dimensional grid layout with subdomain diameter of $H = 4h$ and $CH = 2h$. The fine grid is partitioned into two subdomains indicated by Ω_1 and Ω_2 with grid node 5 being shared by both subdomains.

5.1 First-Order Grid Coarse Interpolation

The details about the first-order grid coarse interpolation operator are straightforward and therefore given in a brief manner. With the coarse grid G_{2h}, the first-order extension operator is given by

$$I^h : G_{2h} \rightarrow G_h, \tag{5.1}$$

with

$$I^h = I_{2h}^h \left[u_{2h}\right]_i = \left\{ \begin{array}{ll} \left[u_{2h}\right]_{(i+1)/2} & \text{if } i \text{ is odd,} \\ \frac{1}{2}\left(\left[u_{2h}\right]_{(i)/2} + \left[u_{2h}\right]_{(i+2)/2}\right) & \text{if } i \text{ is even} \end{array} \right) \tag{5.2}$$

for $i = 1, ..., n$ and the restriction operator is given by

$$(I^h)^T : G_h \rightarrow G_{2h}, \tag{5.3}$$

with

$$(I^h)^T = I_h^{2h} \left[u_h\right]_i = \frac{1}{2}\left[u_h\right]_{(m-1)} + \left[u_h\right]_m + \frac{1}{2}\left[u_h\right]_{(m+1)} \tag{5.4}$$

for $i = 1, ..., \frac{(n+1)}{2}$ with $m = 2i - 1$ and R_0^T and R_0 being the respective corresponding matrices. R_0^T and R_0 are the parts necessary for a two-level Schwarz preconditioner resulting in the first-order interpolation coarse problem now defined.
In a similar way, a first-order prolongation and restriction operator, and their corresponding matrices, can be constructed for a coarse grid with $CH = 4h$.

5.2 Higher-Order Bézier Grid Coarse Interpolation

Some general definition are required, before being able to give the higher-order Bézier prolongation and restriction operator for G_{2h}.

Definition 5.1 (Bézier Curve). *A Bézier curve $B(t)$ of degree n is a parametric curve defined by*

$$B(t) = \sum_{j=0}^{n} b_{j,n}(t) P_j, \quad 0 \leq t \leq 1, \text{ where the polynomials}$$
$$b_{j,n}(t) = \binom{n}{j} t^j (1-t)^{n-j}, j = 0, 1, \ldots, n,$$

are known as the Bernstein basis polynomials of order n. The points P_j are called control points for the Bézier curve.

Definition 5.2 (Rational Bézier Curve). *A rational Bézier curve $C(t)$ of degree n with control points $P_0, P_1, \ldots, P_n$ and scalar weights $w_0, w_1, \ldots, w_n$ is defined as*

$$C(t) = \frac{\sum_{j=0}^{n} w_j b_{j,n}(t) P_j}{\sum_{j=0}^{n} w_j b_{j,n}(t)}.$$

The rational Bézier curve $C(t)$ is just the Bézier curve $B(t)$ but with adjustable weights added. The rational Bézier curve can provide closer approximations to arbitrary shapes.
The Bézier coarse interpolations that are of interest in this research are coarse interpolations constructed using a Bézier curve of degree 2 (quadratic approximation). For a 1D $2h$ Bézier coarse interpolation of degree 2 Definition 5.3 can be given with $i = 1, 2, ..., n$ using Definitions 5.1 and 5.2.

Definition 5.3 ($2h$ coarse grid quadratic approximation). *Let $[u_{2h}]_{(i-1)/2}$ and $[u_{2h}]_{(i+3)/2}$ be the endpoints, P_0 and P_2 respectively, within a component span defined on the coarse grid. Then the prolongation and restriction operator can be characterized by a rational Bézier curve of degree 2 (see Definition 5.2) with polynomials*

$$b_{0,2}(t) = (1-t)^2,$$
$$b_{1,2}(t) = 2t(1-t),$$
$$b_{2,2}(t) = t^2,$$

and control point $[u_{2h}]_{(i+1)/2}$ (P_1), whenever i is odd. We wish to add more weight to the centre value, Therefore the weights are $w_0 = w_2 = \frac{1}{2}$ and $w_1 = \frac{3}{2}$. This results in the following equation

$$C(t) = \frac{\frac{1}{2}(1-t)^2 [u_{2h}]_{(i-1)/2} + \frac{3}{2} 2t(1-t) [u_{2h}]_{(i+1)/2} + \frac{1}{2}(t)^2 [u_{2h}]_{(i+3)/2}}{\frac{1}{2}(1-t)^2 + \frac{3}{2} 2t(1-t) + \frac{1}{2}(t)^2}. \tag{5.5}$$

For $t = \frac{1}{2}$ we find

$$C\left(\frac{1}{2}\right) = \frac{1}{8} [u_{2h}]_{(i-1)/2} + \frac{3}{4} [u_{2h}]_{(i+1)/2} + \frac{1}{8} [u_{2h}]_{(i+3)/2}. \tag{5.6}$$

For the even nodes in the fine grid, a standard equal weight average is used.

Using the upper scheme, the higher-order Bézier restriction and extension operator can be redefined in the following way. The higher-order Bézier extension operator for a G_{2h} coarse grid is given by

$$I^h = I^h_{2h} [u_{2h}]_i = \left\{ \begin{array}{cc} \left(\frac{1}{8} [u_{2h}]_{(i-1)/2} + \left(\frac{3}{4}\right) [u_{2h}]_{(i+1)/2} + \frac{1}{8} [u_{2h}]_{(i+3)/2}\right) & \text{if } i \text{ is odd,} \\ \frac{1}{2}\left([u_{2h}]_{(i)/2} + [u_{2h}]_{(i+2)/2}\right) & \text{if } i \text{ is even} \end{array} \right), \tag{5.7}$$

for $i = 1, ..., n$ and with $m = 2i - 1$ the higher-order Bézier restriction operator is given by

$$(I^h)^T = I^{2h}_h [u_h]_i = \frac{1}{8}\left([u_h]_{(m-2)} + 4 [u_h]_{(m-1)} + 6 [u_h]_m + 4 [u_h]_{(m+1)} + [u_h]_{(m+2)}\right), \tag{5.8}$$

for $i = 1, ..., \frac{(n+1)}{2}$ with respective corresponding matrices R_0^T and R_0. Now the matrices R_0^T and R_0 are defined when a higher-order Bézier approximation is used.

In [43], an additional variable ε is added to the middle weights of equation (5.7) and (5.8). This ε can be used in the Helmholtz solver of [42] in order to improve the distribution of the eigenvalues of the preconditioned system. The right value of ε in [42] is found by using the analytical eigenvalue expressions for the deflated preconditioned system. We do not have such analytical eigenvalue expressions for the two-level Schwarz preconditioned linear system at the moment, and hence we will use $\varepsilon = 0$.

6 | Numerical Results Using a First-Order Grid Coarse Problem

In this chapter, numerical results are discussed with the goal of answering the following research question: Does a Helmholtz solver that uses a two-level additive Schwarz preconditioner combined with first-order grid coarse problem show wave number independent convergence and efficiency? Additionally, the following research sub-question is attempted to be answered: How can the performance of the Helmholtz solver using a first-order grid coarse problem be explained?

To answer the research question of this chapter, we employ GMRES with a two-level AS preconditioner (see equation (4.18)) where the first-order coarse problem is constructed using equations (5.2) and (5.4).
Using the two-level AS preconditioner is abbreviated by $\mathrm{AS}_H/\mathbb{P}_1\text{-}\mathrm{CS}_{CH}$ where H is the subdomain diameter with minimum overlap and $\mathbb{P}_1\text{-}\mathrm{CS}_{CH}$ the notation for using a first-order coarse problem with CH indicating the coarse grid size in one dimension. An example of a specific setup for this preconditioner is $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$. Having $H = 4h$ means that the standard subdomains have a fixed number of grid nodes per subdomain, i.e., n^l is fixed while n increases, resulting in the number of subdomains increasing. Also, having $H = 4h$ means that the subdomain diameter H decreases as the problem size increases. See Figure 6.1 for an illustration of a two dimensional $H = 4$ and $H = 8h$ standard subdomain.

In this chapter, several model problems from Sections 3.2 and 3.3 are considered. The model problems for the numerical experiments are:

- A Poisson problem with a nonhomogeneous Dirichlet boundary condition. This model problem is referred to as MP-3.
- A constant wave number Helmholtz problem with a homogeneous Dirichlet boundary condition. This model problem is also referred to as MP-1.

For all numerical experiments, we employ GMRES using a relative tolerance of $1e-7$ and a maximum number of iterations (NOI) of 250.

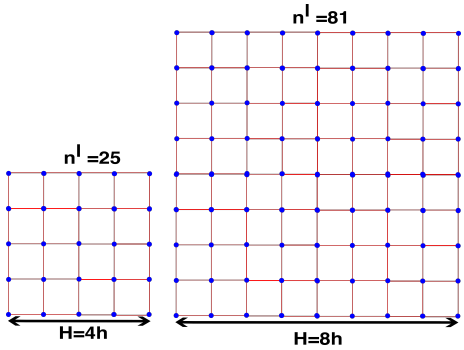


Figure 6.1: Illustration of two dimensional standard subdomains with $H = 4h$ and $H = 8h$. Above the grid n^l is given, which is the total number of grid nodes in the corresponding standard subdomain.

6.1 Two-level Additive Schwarz Preconditioner

In this section the $\mathrm{AS}_H/\mathbb{P}_1\text{-}\mathrm{CS}_{CH}$ preconditioner is applied to both model problems mentioned above. Additionally, eigenvalue analysis of the preconditioned system is performed to create a better understanding of the performance of the preconditioner and compare it to the preconditioners of the next chapter.

6.1.1 Numerical Experiment: MP-3

Subdoms	$\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$	$\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{4h}$	$\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{8h}$
64	14	15	21
256	14	14	23
576	14	14	22
1024	13	14	22
1600	13	14	22

Table 6.1: NOI results for MP-3 using the $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{CH}$ preconditioner with varying CH. Additionally, $\delta = 0$ and n^l is fixed.

Table 6.1 shows numerical results when using the $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-CS}_{CH}$ preconditioner on MP-3. In the table, n^l is fixed and n increases through increasing the number of subdomains. The columns show results for different coarse grid sizes CH. Table B.1 in the appendix is the same as Table 6.1, but then with $H = 8h$ instead of $H = 4h$.
From Table 6.1 and Table B.1 it follows that the preconditioner has a NOI required to reach convergence which is independent of the number of subdomains, i.e., the NOI results in Table 6.1 and Table B.1 stay constant while the number of subdomains increases. This aligns with what we expect from the two-level Schwarz method theory, since we are dealing with a Poisson problem. Increasing H, while keeping the coarse mesh size and overlap fix, results in a slight improvement of the performance of the preconditioner (See Table B.1). The first-order coarse problem with $CH = 2h$, $CH = 4h$ and $CH = 8h$ all result in a preconditioner with a NOI required to reach convergence that is independent of the number of subdomains for varying n^l.
In Appendix B.1 a similar table as Table 6.1 can be found with the $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-CS}_{CH}$ preconditioner with $\delta = 1$ (See Table B.2). Comparing Tables B.2 and 6.1, we conclude that with $\delta = 1$ the performance of the $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-CS}_{CH}$ preconditioner becomes less dependent on CH, meaning the NOI results of Table B.2 are more similar to each other then the NOI results in Table 6.1.

n	$\delta=0$	$\delta=1$	$\delta=2$
64	14	14	19
256	14	15	21
576	14	15	21
1024	13	15	22
1600	13	15	22

(a) $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-CS}_{2h}$

Subdoms	$\delta=0$	$\delta=1$	$\delta=2$	$\delta=4$
16	13	13	14	18
64	13	14	14	18
144	13	14	15	19
256	13	14	15	20
400	13	14	15	20

(b) $\mathrm{AS}_{8h}/\mathbb{P}_1\text{-CS}_{2h}$

Table 6.2: Number of iterations for MP-3 for $\mathrm{AS}_H/\mathbb{P}_1\text{-CS}_{2h}$ preconditioners with $H = 4h$ and $H = 8h$ with different degrees of overlap.

Next, we analyse the influence of δ further. In Tables 6.2 and B.3, the influence of δ on the performance of the preconditioner for varying CH is given. For preconditioners with $H = 4h$, we expect to run into a problem with $\delta = 2$, since that amount of overlap results in the overlap being equal to H. The overlap being equal to or larger than H results in N^c increasing and thus the upper bound of Equation (4.18) increasing. Too much overlap indeed deteriorates the performance of the precondition when looking at the results of Tables 6.2 and B.3. The same hold for preconditioner with $H = 8h$, but then it happens at $\delta = 4$.
From Tables 6.2 and B.3 we conclude that for $CH = 2h$, increasing the overlap only weakens the performance of the preconditioner. Contrary to this, the performance of the preconditioner with $CH = 8h$ improves when the overlap increases, up until there is too much overlap.
This finding does not coincide with the theory of Chapter 4, since the theory suggests that for a fixed H and CH, the upper bound for the condition number should decrease as δ increases. This means that increasing the overlap should improve the performance of the preconditioner up to a certain overlap, but for the preconditioner with $CH = 2h$ this is not the case. The reason for this misalignment between the theory and numerical results might be that the convergence theory for two-level AS preconditioner is based on the usage of a Laplace problem with a homogeneous Dirichlet boundary condition. Also, the theory uses a coarse mesh that has the same diameter as the subdomains. The numerical results above are from MP-3, which is a Poisson problem with a nonhomogeneous Dirichlet boundary condition. Having a nonhomogeneous boundary condition could be the reason for this unexpected result with $CH = 2h$. The other reason could be that H and CH are not always the same diameter in our numerical experiments.

6.1.2 Numerical Experiment: MP-1

For the MP-3 numerical results we expected the two-level AS preconditioner to have the property of having a NOI required to reach convergence which is independent of the number of subdomains. For the MP-1 problem we do not expect this independence of the number of subdomains in the NOI required to reach convergence when using the $\mathrm{AS}_H/\mathbb{P}_1\text{-CS}_{CH}$ preconditioner.
In order for the numerical results of the preconditioner on MP-1 to remain efficient, the aforementioned requirement of $\kappa_h \leq 0.625$ has to hold. The $\mathrm{AS}_H/\mathbb{P}_1\text{-CS}_{CH}$ preconditioner used in this section has a fixed n^l with either a $H = 4h$ or $8h$ subdomain size and the first-order coarse problems has coarse grid mesh size of $CH = 2h$, $4h$ or $8h$.

In Tables 6.3 and B.4, the numerical results of MP-1 using the above mentioned preconditioner with different

subdomain sizes and different first-order coarse grid problems are given. Additionally, the tables have different κ_h requirements.
The first note that can be made about the results in Tables 6.3 and B.4 is that, in general, increasing CH for a fixed overlap, subdomain size and κ_h, deteriorates the performance of the performance of the preconditioner. The best performing preconditioners are the preconditioners with $CH = 2h$. Also, from the tables it is clear that having a lower κ_h results in the NOI required to reach convergence to increase more slowly with the wave number, compared to a higher κ_h. In Appendix B Table B.5 is given, which is similar to Table 6.3, with the only difference being the amount of overlap. Comparing Table 6.3 and Table B.5 it is clear that increasing the amount of overlap worsens the performance of the preconditioner. This aligns with our finding from the previous section about $CH = 2h$ and increasing δ.
Non of the results given in the above mentioned tables show a NOI required to reach convergence that is independent of the wave number, i.e., the NOI of the preconditioner increases as the wave number increases for all the preconditioner setups. The best performing setup of the preconditioner is the $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$ preconditioner with $\kappa_h \approx 0.3125$, since this preconditioner has the slowest increasing NOI with the wave number increasing.

While κ_h is relevant for the solver to be efficient for the Helmholtz problems, we will investigate if the number of coarse grid nodes per wave length in relevant for wave number independent NOI results. In other words, does lowering κ_h for the best performing preconditioner eventually lead to a preconditioner which has a NOI required to reach convergence which is independent of the wave number? We define $\kappa_{CH} = k * CH$. So if $CH = 2h$, then $\kappa_{CH} = 2kh = 2\kappa_h$ and if $CH = 4h$, then $\kappa_{CH} = 4kh = 4\kappa_h$.
In Figure 6.2 the NOI required to reach convergence for the $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$ preconditioner on MP-1 are given with varying κ_{CH}. From this figure, it seems that there is no number of coarse grid nodes per wave length for which the preconditioner has a NOI required to reach convergence which is wave number independent. This is because in the figure it can be seen that there is no κ_{CH} for which the NOI does not increase as the wave number increases.

Hence, decreasing κ_h results in the NOI required to reach convergence to increase even slower when the wave number increases. But the NOI never appear to become independent of the wave number. Furthermore, in Table B.6 in the appendix the NOI required to reach convergence are given for the $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$ preconditioner on MP-1 with slowly increasing low wave numbers and a slowly increasing number of subdomains. From this table, it also follows that there is no κ_h for which the NOI required to reach convergence is independent of the wave number.
Not finding a preconditioner which has a NOI required to reach convergence that is independent of the wave number is what we expect from a two-level AS preconditioner with a first-order grid coarse problem.

Subdoms	k	$\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$	$\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{4h}$	$\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{8h}$
4	5	11	10	7
36	15	24	24	24
100	25	39	46	50
196	35	72	97	100
324	45	100	130	160
400	50	119	193	220
676	65	x	x	x

(a) $\kappa_h \approx 0.625$

Subdoms	k	$\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$	$\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{4h}$	$\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{8h}$
16	5	13	14	14
144	15	16	21	29
400	25	17	23	47
784	35	20	41	86
1296	45	21	46	113
1600	50	22	53	145
2704	65	33	75	233

(b) $\kappa_h \approx 0.3125$

Table 6.3: NOI results for MP-1/WO using the $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{CH}$ preconditioner with varying CH. The wave number and the number of subdomains increase such that $\kappa_h \approx 0.625$ or $\kappa_h \approx 0.3125$ remain true. Additionally, $\delta = 0$.

Analyzing the eigenvalues of the preconditioned linear systems of MP-1 might give more insights into the performance of the preconditioners on MP-1. Specifically, the eigenvalues might show us why the $\mathrm{AS}_H/\mathbb{P}_1\text{-}\mathrm{CS}_{CH}$ preconditioner results in the solver having a NOI required to reach convergence that is dependent on the wave number. Here, we specifically look at the $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$ preconditioner, since this is the best performing setup of the preconditioner.

Figure 6.3 and Appendix B.2 show eigenvalue distribution plots of the $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$ preconditioner linear system. Additionally, in Figure 6.3 histogram plots of the eigenvalues are given, to get an understanding of the distribution of the eigenvalues.

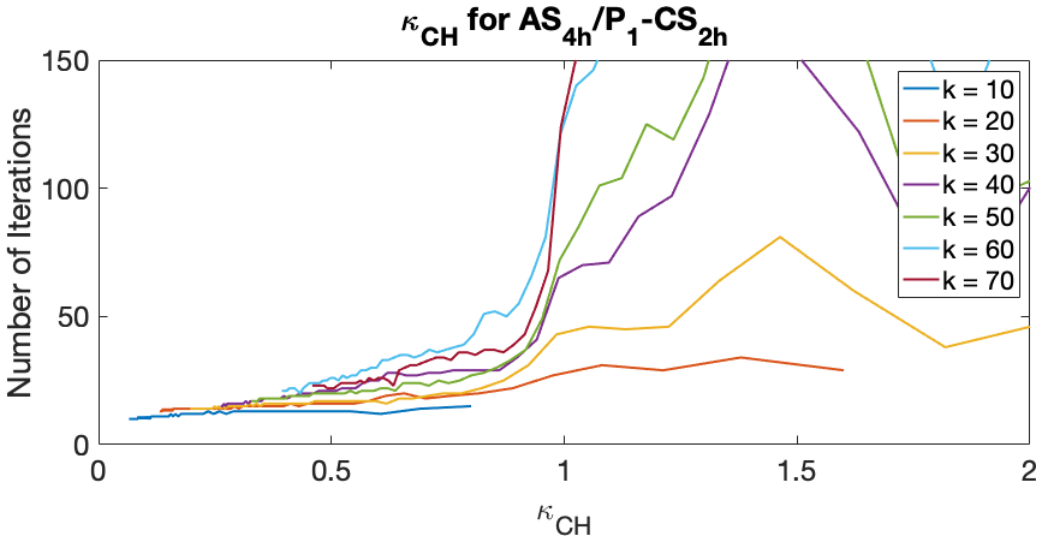
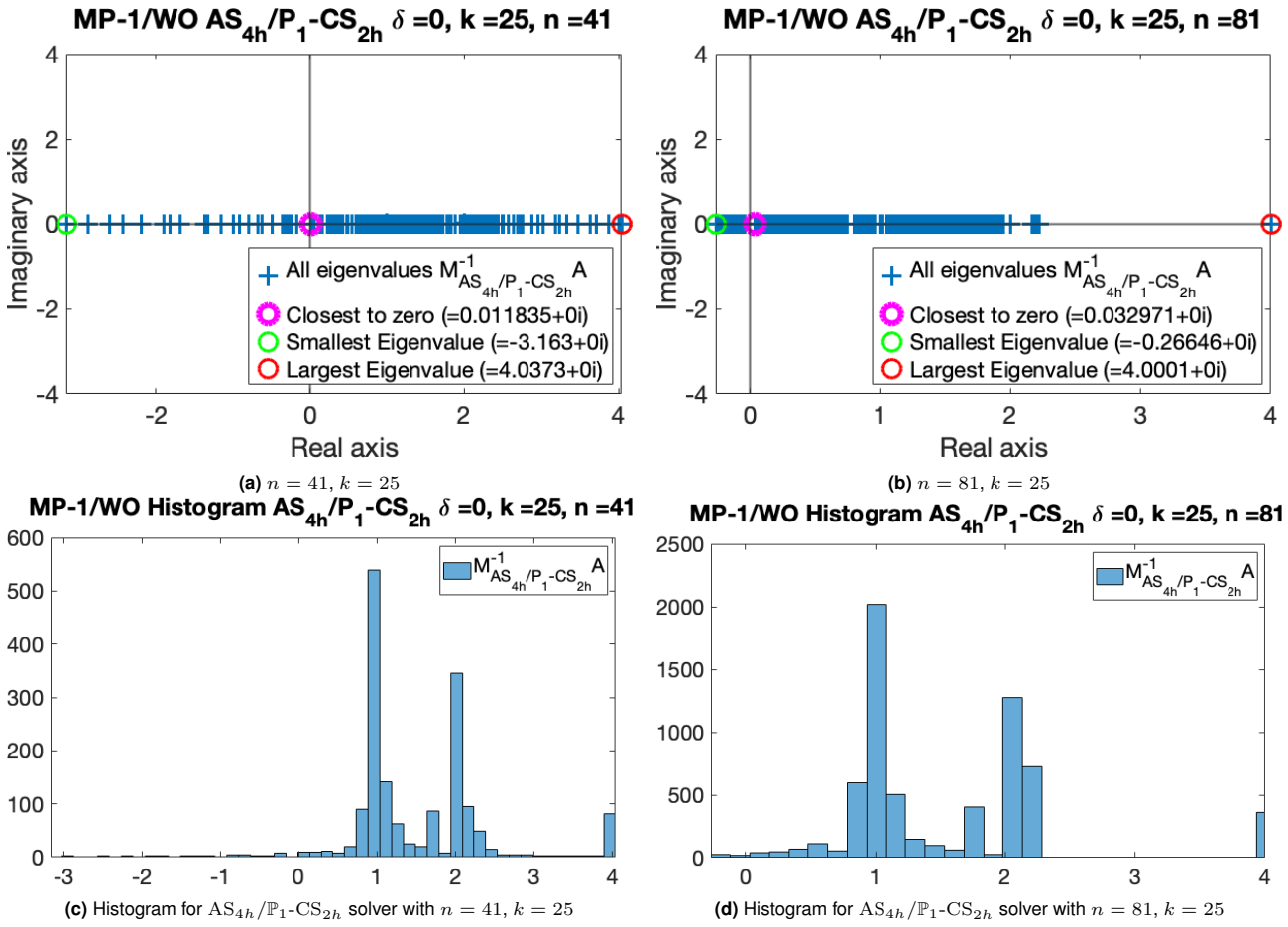


Figure 6.2: κ_{CH} plot of $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$ preconditioner.



(a) $n = 41, k = 25$

(b) $n = 81, k = 25$

(c) Histogram for $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$ solver with $n = 41, k = 25$

(d) Histogram for $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$ solver with $n = 81, k = 25$

Figure 6.3: Eigenvalue plots and histograms with $k = 25$ and either $\kappa_h = 0.625$ or $\kappa_h = 0.3125$ with the $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$ Helmholtz solver. Note that $\delta = 0$

When looking at both the eigenvalue plots in Appendix B.2 and in Figure 6.3 some remarks can be made. Firstly, the eigenvalue plots show negative eigenvalues. This is expected since the Helmholtz problem becomes indefinite already for small wave numbers [42]. Also, the largest eigenvalue is roughly 4 and the eigenvalue closest to 0 moves closer to 0 as the wave number and the problem size increase. These observations apply for both the $\kappa_h \approx 0.625$ and $\kappa_h \approx 0.3125$ eigenvalue plots, see Figure B.2 and B.3. The largest eigenvalue being roughly 4 is a consequence of using a Schwarz domain decomposition preconditioner since $N^c = 4$.

Clear differences between the eigenvalue plots with $\kappa_h \approx 0.3125$ and $\kappa_h \approx 0.625$ are the following. For $\kappa_h \approx 0.3125$, the overall clustering of the eigenvalues is closer together compared to the eigenvalue when $\kappa_h \approx 0.625$. Additionally, the smallest eigenvalue is more negative when $\kappa_h \approx 0.625$, and for both $\kappa_h \approx 0.625$ and $\kappa_h \approx 0.3125$ the smallest eigenvalue decreases as the wave number increases. Furthermore it can be noted that, in general, the positive eigenvalue do not come close to the largest eigenvalue when $\kappa_h \approx 0.3125$ is used, i.e., in the eigenvalue plots with $\kappa_h \approx 0.3125$, there are no eigenvalues between 2.2 and 4. There is a gap between the eigenvalues around 4 and the area where most of the eigenvalues are clustered. This is not the case for the eigenvalue with $\kappa_h \approx 0.625$. Finally, for $\kappa_h \approx 0.625$ the eigenvalue closest to zero moves closer to zero faster when the wave number increases compared to the closest to zero eigenvalue in the $\kappa_h \approx 0.3125$ plots.
Some speculations on an explanation for the results of Table 6.3 and the eigenvalue plots could be that the numerical range (also known as field of values) has an influence on the NOI required to reach convergence of the solvers. Another possible explanation for the results of Table 6.3 could be related to how small the eigenvalue closest to zero is. Thirdly, the eigenvalue plots do not give details about the density of the eigenvalue distributions close to zero. The histogram plots in Figure 6.3 do give more details about the distribution, but do not show any clear differences that could explain the performance of the preconditioners. Finally, the results might be explained be a combination of the speculations above.

Thorough spectral analysis needs to be performed on the Schwarz domain decomposition preconditioner to be able to draw concrete conclusions on what aspects of the eigenvalues influence the NOI required to reach convergence of the solver.

6.2 Concluding Remarks and Summary

To finalize this chapter, some concluding remarks and summarizing notes are given in this section.

- The MP-1 numerical results show that κ_h, and thus also κ_{CH}, is important not only for efficiency of the solver, but also for how fast the NOI increases as the wave number increases. Also, it seems that lowering κ_{CH} does not eventually lead to a solver with a NOI required to reach convergence that is independent of the wave number.
- Wave number independent convergence is not found for a two-level AS preconditioner with a $\mathbb{P}_1$-CS coarse problem for the MP-1 Helmholtz model problem.
- The eigenvalue plots show the influence of κ_h on the eigenvalues, that is, the closest to zero eigenvalue moves closer to zero more slowly for a lower κ_h.
- No clear answer can be given that explains the performance of the two-level AS preconditioner using a $\mathbb{P}_1$-CS, and why it results in a NOI that is wave number dependent.

7 | Numerical Results Using a Higher-Order Bézier Grid Coarse Problem

In the following chapter, we first try to answer research question 2 which is: Does a Helmholtz solver, constructed using a two-level additive Schwarz preconditioner and a higher-order Bézier grid coarse problem, show wave number independent convergence and efficiency? Also, the following research sub-question is attempted to be answered: What causes the solver to be inscalable if this is the case?
After answering research question 2, we will try to answer research question 3, which is: Does a Helmholtz solver, constructed using different variants of two-level Schwarz preconditioners and a higher-order Bézier coarse problem, show wave number independent convergence and efficiency? With the research sub-questions being: For the scaled additive Schwarz preconditioner, does the solver show wave number independent convergence and efficiency and if so, why? And, for the scaled hybrid Schwarz preconditioner, which uses deflation, does the solver show wave number independent convergence and efficiency and if so, why?

This chapter makes use of all five different model problems, which are 2D problems unless stated otherwise. Details about these models are found in Sections 3.2 and 3.3. These five different model problems are the MP-3 and MP-1 model problem, which were already used in the previous chapter, and:

- A constant wave number Helmholtz problem with a Sommerfeld radiation condition on the boundary. This model problem is referred to as MP-2.
- A nonconstant wave number wedge-domain Helmholtz problem. The model problem has a Sommerfeld radiation condition on the boundary. More details on this model problem can be found in Section 3.2.
- A 3D constant wave number Helmholtz problem with a homogeneous Dirichlet boundary condition. More details on the model problem is found in Section 3.2.

MP-3, MP-2 and MP-1 are model problems which increase in difficulty for numerical solvers, with MP-1 being the most difficult. After a preconditioner has shown to solve in a NOI independent of the wave number, the preconditioner is then applied to the wedge Helmholtz model problem and the 3D Helmholtz model problem.

The first section of this chapter focuses on numerical results and eigenvalue analysis of the solver with two-level AS preconditioning and a higher-order Bézier grid coarse problem. Here, we first try to find a setup for the preconditioner that yields wave number independence and efficiency. Additionally, an eigenvalue analysis is performed in order to get further insight into the behavior of the preconditioner for certain problems. After finding wave number independence and efficiency with two-level AS preconditioner using a higher-order Bézier coarse problem, numerical results of that preconditioner are presented on a Helmholtz wedge model problem and a 3D Helmholtz model problem. Finally, results are given showing the influence of κ_h on the wave number independence of the NOI required to reach convergence. In the next section, numerical results of a variety of different two-level Schwarz preconditioners with a higher-order Bézier coarse problem are given. In particular, we additionally consider, the SAS, HS and SHS preconditioners. Numerical results for solvers with these preconditioners are first given on MP-1, to see if the preconditioners provide wave number independence for the NOI results. Next, an eigenvalue analysis is performed to get a better understanding of the performance of the preconditioners and compare them. Additionally, numerical results of the preconditioners on the wedge model problem are given, to see how they deal with a more real model problem. As in the previous chapter, numerical results for κ_h are analysed, to understand the influence of κ_h on the wave number independence of the solvers better. The third section of this chapter briefly compares the convergence rate results of the different preconditioners used in this chapter. Finally, the chapter finishes with a section which contains concluding and summarizing remarks.

For all numerical experiments, we employ GMRES using a relative tolerance of $1e-7$ and a maximum number of iterations (NOI) of 100. Note that, for a fixed n and CH, the number of grid nodes of the higher-order coarse problem is the same as that of the first-order coarse problem when CH is the same, but the higher-order coarse problem is more dense, i.e. it captures more information from the fine grid.

7.1 Two-level Additive Schwarz Preconditioner

This section focuses on numerical results for the two-level AS preconditioner and a higher-order Bézier coarse problem. The preconditioner is abbreviate by $\mathrm{AS}_H/\mathrm{BCS}_{CH}$. An example of a specific setup for this solver is

$\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$. Having H depend on h means that the number of grid nodes in a standard subdomain grid is fixed, i.e., $H = 4h$ means $n^l = 25$, which is fixed (See Figure 6.1 for an illustration). More information about the grid coarse problems can be found in Chapter 5.

7.1.1 Numerical Experiments: Wave Number Independence

The solver using a two-level AS preconditioner with a BCS is initially tested on the MP-3. The NOI results for MP-3 are expected to be independent of the number of subdomains. In Table 7.1 and Table C.1 the NOI results for MP-3 are given for $H = 4h$ and $H = 8h$, respectively, for various CH and $\delta = 0$ with the number of subdomains increasing.
From Table 7.1 and Table C.1 it can be seen that the two-level AS preconditioners with a BCS yields NOI results which are independent of the number of subdomains. NOI results with the $\mathrm{AS}_{4h}/\mathrm{BCS}_{CH}$ preconditioner with $\delta = 1$ can be found in Table C.2. When the overlap of the $\mathrm{AS}_{4h}/\mathrm{BCS}_{CH}$ preconditioner for various CH is increased to $\delta = 1$, the NOI results remain independent of the number of subdomain. Additionally, as with the $\mathbb{P}_1$-CS, the NOI results of the $\mathrm{AS}_{4h}/\mathrm{BCS}_{CH}$ preconditioner with $\delta = 1$ are more similar to each other compared to the NOI results of Table 7.1.

Subdoms	$\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$	$\mathrm{AS}_{4h}/\mathrm{BCS}_{4h}$	$\mathrm{AS}_{4h}/\mathrm{BCS}_{8h}$
64	14	11	21
256	13	11	22
576	13	11	22
1024	13	11	22
1600	13	11	22

Table 7.1: NOI results for MP-3 using the $\mathrm{AS}_{4h}/\mathrm{BCS}_{CH}$ preconditioner with varying CH. Additionally, $\delta = 0$.

With the preconditioner performing well on the MP-3 numerical experiment the next step is to increase the difficulty of the problem by performing numerical experiments with model problem MP-2. For the number of grid points per wave length we take $\kappa_h \approx 0.3125$. The reason for taking $\kappa_h \approx 0.3125$ is that the results in Subsection 6.1.2 with the $\mathrm{AS}_H/\mathbb{P}_1\text{-}\mathrm{CS}_{CH}$ preconditioner with $\kappa_h \approx 0.3125$ were more promising than the results with $\kappa_h \approx 0.625$. κ_h could also be decreased even further then 0.3125. The consequence of, for example, taking $\kappa_h \approx 0.15625$ is that the coarse problem becomes four times as large for a fix CH. A coarse problem with a problem size that large and rapidly increasing is undesirable, since a direct solver is used on the coarse problem. Another consequence of using $\kappa_h \approx 0.15625$ instead of $\kappa_h \approx 0.3125$ is that n^l has to increase and/or the number of subdomains has to increase. In Subsection 7.1.2 the choice of $\kappa_h \approx 0.3125$ is investigated further.

Subdoms	k	$\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$	$\mathrm{AS}_{4h}/\mathrm{BCS}_{4h}$	$\mathrm{AS}_{4h}/\mathrm{BCS}_{8h}$
64	10	17	15	24
144	15	17	17	32
256	20	18	17	40
400	25	18	18	48
1600	50	18	22	x
6400	100	18	30	x
14400	150	17	38	x

(a) $H = 4h$

Subdoms	k	$\mathrm{AS}_{8h}/\mathrm{BCS}_{2h}$	$\mathrm{AS}_{8h}/\mathrm{BCS}_{4h}$	$\mathrm{AS}_{8h}/\mathrm{BCS}_{8h}$
16	10	19	16	21
36	15	19	20	29
64	20	21	22	36
100	25	23	24	44
400	50	27	32	x
1600	100	27	39	x
3600	150	27	46	x

(b) $H = 8h$

Table 7.2: NOI results for MP-2 using the $\mathrm{AS}_H/\mathrm{BCS}_{CH}$ preconditioner with varying CH and $H = 4h$ or $H = 8h$. Additionally, $\kappa_h \approx 0.3125$ and $\delta = 0$.

In Table 7.2 the NOI results are given for the $\mathrm{AS}_{4h}/\mathrm{BCS}_{CH}$ and $\mathrm{AS}_{8h}/\mathrm{BCS}_{CH}$ preconditioner with varying CH and $\delta = 0$ with an increasing wave number and $\kappa_h \approx 0.3125$. In Table C.3 in the appendix κ_h is lowered to $\kappa_h \approx 0.15625$. From Table 7.2 we see that, increasing CH for a $\mathrm{AS}_H/\mathrm{BCS}_{CH}$ preconditioner with a fixed overlap, fixed n^l and fixed κ_h deteriorates the performance of the preconditioner. Specifically, the results of the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner are wave number independent, but this property is lost when CH increases. For the $\mathrm{AS}_{8h}/\mathrm{BCS}_{2h}$ preconditioner with $\kappa_h \approx 0.3125$ (Figure Table 7.2 (**b**)) it is not clear if the NOI results are wave number independent, since there is an increase in the NOI when the wave number increases, but it also

seems that an upper bound for the NOI is reached at 27 iterations.
From Table C.3 it can be seen that lowering κ_h to $\kappa_h \approx 0.15625$ results in the above mentioned consequences. In general, lowering κ_h results in the $\mathrm{AS}_H/\mathrm{BCS}_{CH}$ preconditioner with $CH = 4h$ showing NOI results that are close to but not wave number independent. Hence, there is little benefit to lowering κ_h from $\kappa_h \approx 0.3125$ to $\kappa_h \approx 0.15625$, while increasing $CH = 2h$ to $CH = 4h$, since n^c stays the same, but we lose wave number independence in the NOI results. The only benefit of having $\kappa_h \approx 0.15625$ and $CH = 4h$ looks to be that the NOI results depend less on the choice of H, which can be seen in Table C.3.

To investigate the results of using the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner with $\kappa_h \approx 0.3125$ further Table C.4 is constructed, which shows the NOI results for a slowly increasing wave number and a slowly increasing number of subdomains. Something to note about Table C.4 is that there appears to be a value of κ_h for which the NOI does not increase. For a fixed wave number and and fixed n^l, n increases if the number of subdomains increases and κ_h decreases. From Table C.4 it can be seen that increasing the number of subdomains a certain amount every time the wave number is increased, results in the NOI required to reach convergence to not increase. In other words, there seems to be a κ_h for which the preconditioner requires a NOI which is wave number independent. Additionally, when κ_h is small enough such that the NOI results are wave number independent, then the NOI needed to reach convergence is around 17 and 18. Later in this section we investigate if κ_h has an upper bound such that the NOI results are wave number independent.

Next, we investigate increasing the overlap of $\mathrm{AS}_{4h}/\mathrm{BCS}_{CH}$ to $\delta = 1$. In Table 7.3 numerical results of the $\mathrm{AS}_{4h}/\mathrm{BCS}_{CH}$ preconditioner with $\delta = 1$ are given for a fixed n^l and an increasing wave number. From the results we conclude that increasing the overlap to $\delta = 1$ worsens the performance of the preconditioner with $CH = 2h$ and $CH = 4h$, and improves the performance of the preconditioner with $CH = 8h$ slightly. This influence of the overlap is the same as was seen with the $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{CH}$ preconditioner. Most importantly, increasing the overlap of the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner results in the NOI to reach convergence to be wave number dependent. As a final note on Table 7.3, comparing the results with Table 7.2 it appears that the NOI results are less dependent on CH when $\delta = 1$. This is because the results with $CH = 2h$ and $CH = 4h$ in Table 7.3 are all more similar to each other. For the $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{CH}$ preconditioner, the same behaviour was seen on MP-1.

Subdoms	k	$\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$	$\mathrm{AS}_{4h}/\mathrm{BCS}_{4h}$	$\mathrm{AS}_{4h}/\mathrm{BCS}_{8h}$
64	10	18	18	21
144	15	22	21	28
256	20	25	24	35
400	25	29	27	41
1600	50	37	31	x
6400	100	38	33	x
14400	150	37	36	x

Table 7.3: NOI results for MP-2 using the $\mathrm{AS}_{4h}/\mathrm{BCS}_{CH}$ preconditioner with varying CH. Additionally, $\kappa_h \approx 0.3125$ and $\delta = 1$.

The final benchmark for wave number independence is the model problem MP-1. From the MP-2 results it follows that some setups of the $\mathrm{AS}_H/\mathbb{P}_1\text{-}\mathrm{CS}_{CH}$ preconditioner have NOI results that grow faster with the wave number then other. These poor performing preconditioner setups are omitted from the numerical results of MP-1.

Subdoms	k	$\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$
64	10	14
144	15	17
256	20	17
400	25	17
1600	50	16
6400	100	12
14400	150	15
57600	300	14

(a) $H = 4h$

Subdoms	k	$\mathrm{AS}_{8h}/\mathrm{BCS}_{2h}$
16	10	13
36	15	17
64	20	20
100	25	22
400	50	25
1600	100	18
3600	150	22
14400	300	20

(b) $H = 8h$

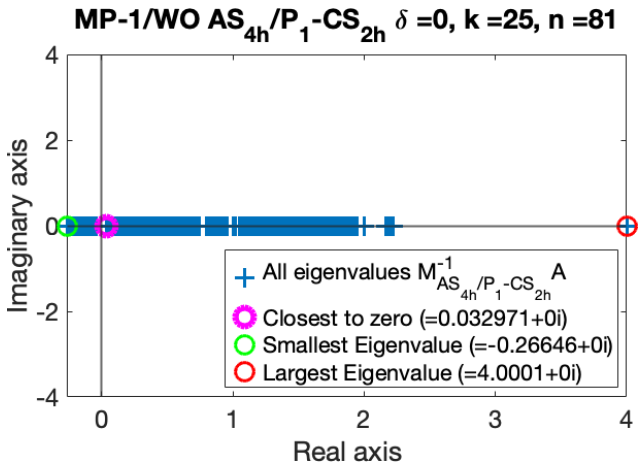
Table 7.4: NOI results for MP-1 using the $\mathrm{AS}_H/\mathrm{BCS}_{CH}$ preconditioner with $CH = 2h$ and $H = 4h$ or $H = 8h$. Additionally, $\kappa_h \approx 0.3125$ and $\delta = 0$.

In Table 7.4 numerical results are given of the most promising setups of the $\mathrm{AS}_H/\mathrm{BCS}_{CH}$ preconditioner with $\kappa_h \approx 0.3125$. Table 7.4 shows wave number independence in the NOI results for both the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ and $\mathrm{AS}_{8h}/\mathrm{BCS}_{2h}$ preconditioner setup up to at least $k = 300$. The results show that, the CH size plays an important role in achieving NOI results that are wave number independent for a fixed κ_h.
In general, the two-level AS Schwarz theory from Chapter 4 does not hold for the Helmholtz problem, but it might be that parts of the theory also hold for the Helmholtz problem. Looking at Equation (4.18) it is not surprising that preconditioner with $H = 4h$ in Table 7.4 (**a**) has lower NOI results then the preconditioner with

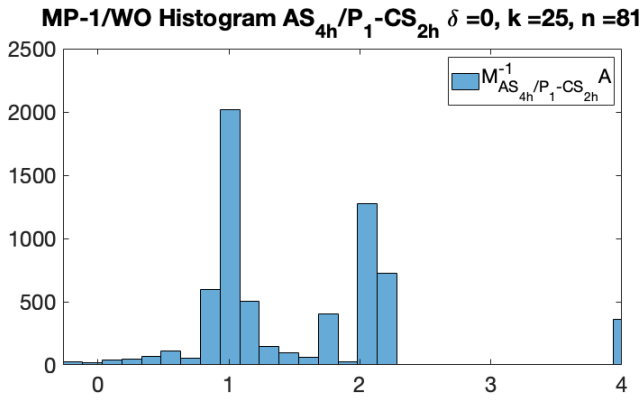
$H = 8h$ in Table 7.4 (**b**). Having a smaller n^c does appear to improve the performance of the preconditioner for the Helmholtz problem. Also, the NOI required to reach convergence does vary more when increasing the wave number with the $\mathrm{AS}_{8h}/\mathrm{BCS}_{2h}$ preconditioner then with the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner.
In Table C.5 NOI results are given for the $\mathrm{AS}_H/\mathrm{BCS}_{4h}$ preconditioner with $\kappa_h \approx 0.15625$ and $CH = 4h$. We see from these results the NOI results are also wave number independent.

From the numerical results of MP-2 with the $\mathrm{AS}_H/\mathrm{BCS}_{CH}$ preconditioner, it was concluded that for $CH = 2h$ the performance of the preconditioner only deteriorates when the overlap is increased. Hence, $\delta > 0$ is not investigated for the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ and $\mathrm{AS}_{8h}/\mathrm{BCS}_{2h}$ preconditioner setup on MP-1.

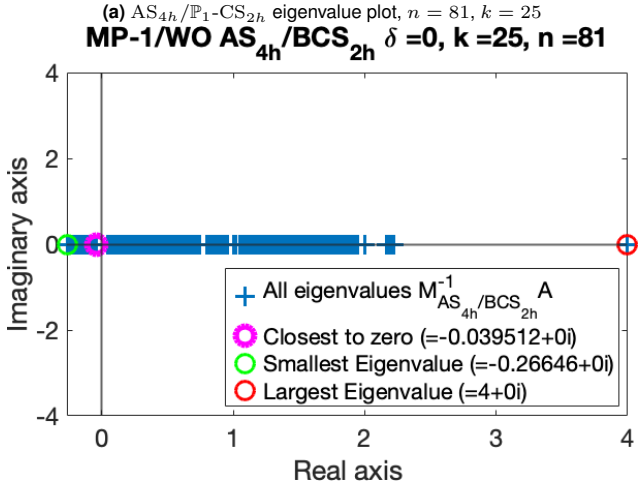
Figure 7.1 provides the eigenvalues distributions and histograms of the eigenvalues for both the $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$ and the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner. Additional eigenvalue distribution plots are found in Appendix C in Figure C.1. From these eigenvalue plot in Figure 7.1 (**c**) and Figure C.1, many similarities can be seen with Figure 7.1 (**a**). In Figure C.1 (**c**) the smallest eigenvalue is a lot smaller than the rest of the eigenvalues. The reason for this outlier eigenvalue is not clear and when looking at Table 7.4 it does not appear to deteriorate the performance of the solver. Therefore, it is assumed that the smallest eigenvalue in C.1 (**c**) is an anomaly. This anomaly also appears in some other eigenvalue plots for in the next section.
Up until now, n^l has not changed much. Often, we have taken $H = 4h$ or $H = 8h$. Table C.6 in the Appendix looks at the influence of n^l on the NOI results for MP-1. This is done by fixing n, CH and k while changing n^l. The results in Table C.6 show that n^l greatly influence the NOI results and that sticking to a subdomain size H of $4h$ and $8h$ is good.



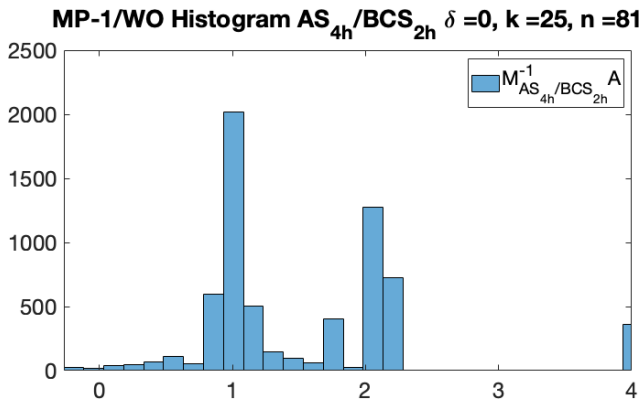
(a) $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$ eigenvalue plot, $n = 81$, $k = 25$



(b) $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$ histogram of eigenvalues, $n = 81$, $k = 25$



(c) $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ eigenvalue plot, $n = 81$, $k = 25$



(d) $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ histogram of eigenvalues, $n = 81$, $k = 25$

Figure 7.1: Eigenvalue plots and histograms with $k = 25$ with the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ and $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$ Helmholtz solver for MP-1. Note that $\delta = 0$.

7.1.2 An Upper Bound for κ_{CH}

In Section 3.1, requirements were introduced for the relationship between the number of grid points and the wave number. When there are enough grid points per wave length, the pollution error can be mitigated

($\kappa_h \leq 0.625$) or the pollution error can be removed all together ($k^3h^2 \leq 1$).
$k^3h^2 \leq 1$ is a harsh requirement which causes the number of grid nodes to increase rapidly when the wave number increases. Therefore, the weaker requirement $\kappa_h \approx kh \leq 0.625$ is used when trying to construct a Helmholtz solver that is wave number independent and efficient.

BCS_{CH} with $CH = 2h$ was used in the numerical experiments on MP-1. In the previous chapter, κ_{CH} was defined. For some setups of the $\mathrm{AS}_H/\mathrm{BCS}_{2h}$ preconditioner we will investigate the influence of κ_{CH} and see if there is a general relationship between the number of coarse grid nodes in one dimension and the wave number. The investigation is done using model problem MP-1 and increasing the number of subdomains and the wave number.

We are interested in finding out how the behavior of the preconditioner changes depending on κ_{CH}. Figure 7.2 shows the influence of κ_{CH} on the performance of the $\mathrm{AS}_{8h}/\mathrm{BCS}_{2h}$ preconditioner for MP-1. From this, we can see that κ_{CH} is ideally lower than $\kappa_{CH} \approx 0.625$ with $CH = 2h$. This explains the behavior of the $\mathrm{AS}_{8h}/\mathrm{BCS}_{2h}$ preconditioner in Table 7.4 and suggest that the $\mathrm{AS}_{8h}/\mathrm{BCS}_{2h}$ preconditioner should have a κ_h slightly lower then $\kappa_h \approx 0.3125$. In general, it appears that there is an upper bound for κ_{CH}, but it is not yet clear if this upper bound depends on CH and/or H.
In Appendix C in Figure C.2 a similar plot for κ_{CH}, but then for the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner is given. This shows that $\kappa_{CH} = 0.625$ is satisfactory for wave number independent NOI results. Comparing Figure 7.2 and Figure C.2 also suggests that a upper bound for κ_{CH} depends on H, since the upper bound moved as a consequence of changing n^l.

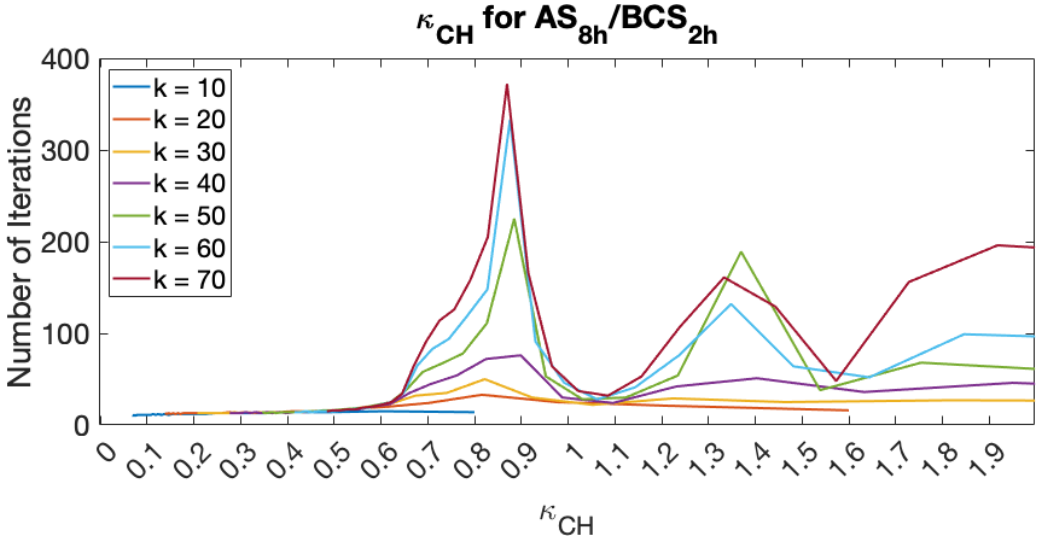


Figure 7.2: $\mathrm{AS}_{8h}/\mathrm{BCS}_{2h}$ on MP-1

7.1.3 Numerical Experiment: Engineering-like Problems

Now that the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ Helmholtz preconditioner is shown to be wave number independent and efficient with $\kappa_h \approx 0.3125$, the preconditioner can also be applied to numerical experiments more real model problems. Since the $\mathrm{AS}_{8h}/\mathrm{BCS}_{2h}$ Helmholtz preconditioner requires $\kappa_h < 0.3125$ for wave number independent convergence, the preconditioner is left out of the following numerical results. Table 7.5 shows wave number independent NOI results for the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner with $\kappa_h \approx 0.3125$ for the wedge model problem. In Table 7.6 the numerical results are given for the 3D MP-1 model problem with the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner. The NOI results of Table 7.6 are also wave number independent. Note that the number of subdomains becomes very large very quickly, which is a consequence of the problem being three dimensional, having small subdomains and having $\kappa_h \approx 0.3125$. From the previous numerical results in this section it was concluded that increasing H results in

Subdoms	k	$\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$
64	10	18
400	25	18
1600	50	18
6400	100	18
14400	150	18

Table 7.5: NOI results with the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner when $\kappa_h \approx 0.3125$ for the wedge Helmholtz problem. Additionally, $\delta = 0$.

a loss of wave number independence and worse NOI results. Hence, there is no way of lowering the number of subdomains required in Table 7.6 without deteriorating the results.
The computational time of the 3D MP-1 numerical experiment is large, which results in the wave number only going up to $k = 25$. This high computational cost is mostly due to n^c becoming too large, which results in the solving the coarse problem requiring a lot of time.

If we summarize the findings from the numerical results of this section, the following can be said. First, from the MP-2 numerical results we find that the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner with $\kappa_h \approx 0.3125$ shows NOI results that are independent of the wave number. Additionally, making the coarse grid twice as coarse and κ_h twice as low does also results in wave number independence, but the coarse problem stays the same size. Therefore, using $CH > 2h$ does not have any benefits to the preconditioner. The $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner with $\kappa_h \approx 0.3125$ is wave number independent for the MP-1 model problem, the wedge model problem and the 3D MP-1 model problem. Next, comparing the eigenvalue plots of the $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$ preconditioner and the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner shows no information that can explain the wave number independent performance of the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner. Finally, the number of grid nodes per standard subdomain, n^l, has an influence on whether the $\mathrm{AS}_H/\mathrm{BCS}_{2h}$ preconditioner requires a NOI to reach convergence which is independent of the wave number.

Subdoms	k	$\mathrm{AS}_{4h}/\text{3D-}\mathrm{BCS}_{2h}$
4096	5	17
32768	10	19
110592	15	20
512000	25	19

Table 7.6: results with the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner when $\kappa_h \approx 0.3125$ for the 3D MP-1 problem. Additionally, $\delta = 0$.

7.2 Numerical Results with Different Schwarz Methods

For the final research question, different variants of Schwarz preconditioners are tested to see the impact on the performance of the solver. The preconditioners using a two-level SAS, HS and SHS in combination with BCS are respectively abbreviated by $\mathrm{SAS}_H/\mathrm{BCS}_{CH}$, $\mathrm{HS}_H/\mathrm{BCS}_{CH}$, $\mathrm{SHS}_H/\mathrm{BCS}_{CH}$.
To prevent this section from being too lengthy the numerical experiments of the solvers with new Schwarz preconditioners are only performed on MP-1 and the wedge problem. Each subsection in this section covers one of the variants of the different Schwarz preconditioners. In the subsection, the numerical results for that preconditioner are given. After that, eigenvalue analysis is performed on the preconditioner. Finally, each subsection ends with an investigation of the influence of κ_{CH} on the wave number independence.

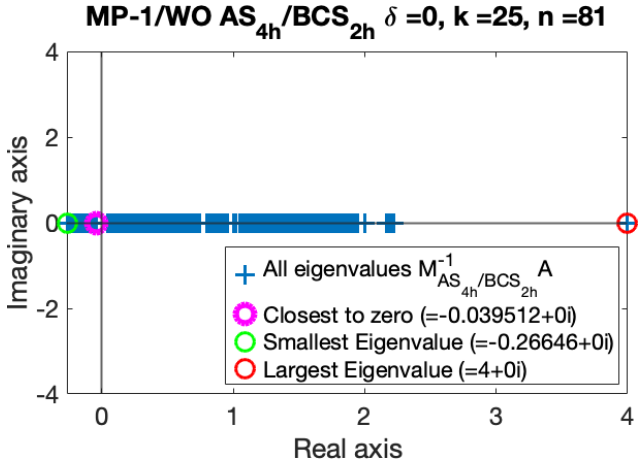
7.2.1 Two-Level Scaled Additive Schwarz Preconditioner

The SAS domain decomposition preconditioner is introduced in Chapter 4 in equation (4.19). From the numerical results of MP-1 in Table 7.7 it can be seen that the $\mathrm{SAS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner requires a NOI to reach convergence that is wave number independent. Besides that, the NOI results are respectively lower compared to the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner results.
Figure 7.3 has the eigenvalue plots and the eigenvalue histograms of both the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ and $\mathrm{SAS}_{4h}/\mathrm{BCS}_{2h}$ preconditioners, so that they can be compared. In Figure 7.3 (**c**) it can be seen that the eigenvalues of the preconditioner linear system using the $\mathrm{SAS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner are more compactly clustered. Specifically, the high frequency eigenvalues have moved towards 1 with some remaining between 1 and 2. The low frequency eigenvalues in Figure 7.3 (**c**) seem similar as in Figure 7.3 (**a**). This more compact clustering of the eigenvalues is a possible explanation for the lower NOI required to reach convergence of the $\mathrm{SAS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner compared to the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner. As a final note on Figures 7.3 (**a**) and (**c**), we see that the closest to zero eigenvalue is not influenced by the changing the Schwarz preconditioner from AS to SAS.

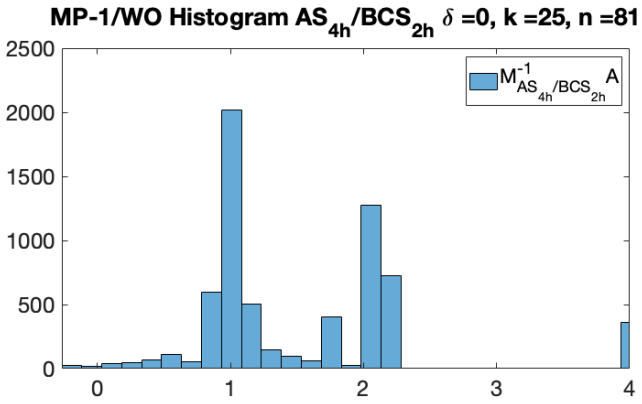
Subdoms	k	$\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$	$\mathrm{SAS}_{4h}/\mathrm{BCS}_{2h}$
64	10	14	10
400	25	17	14
1600	50	16	13
6400	100	12	10
14400	150	15	12

Table 7.7: NOI results of $\mathrm{SAS}_{4h}/\mathrm{BCS}_{2h}$ with $\kappa_h \approx 0.3125$ for the MP-1 compared to previous results. Additionally, $\delta = 0$.

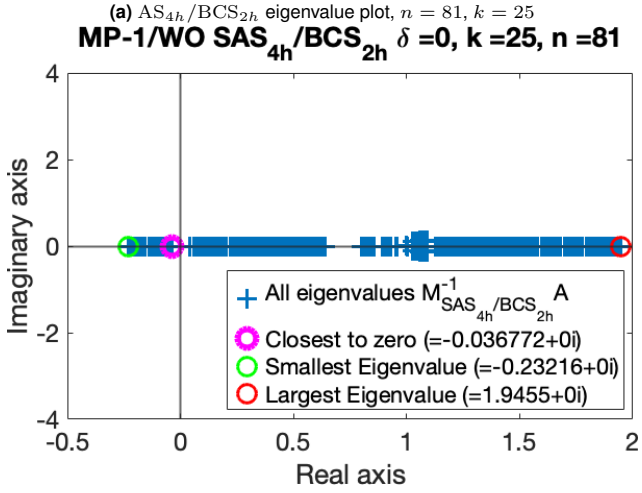
In the Figure C.3 in the appendix, a plot to investigate κ_{CH} is given for the $\mathrm{SAS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner. From this figure it can be concluded that $\kappa_{CH} \approx 0.625$ is satisfactory for the $\mathrm{SAS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner to have NOI results are wave number independent, which aligns with our numerical results. Moreover, κ_h could even be increased slightly, resulting in the problem size becoming less large, and the number of subdomains increasing slower, as the wave number increases. For the $\mathrm{SAS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner the numerical results in Table 7.8 are wave number independent on the wedge model problem. Again, comparing the results with the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner results, the NOI required to reach convergence is lower for the $\mathrm{SAS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner then for the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner.



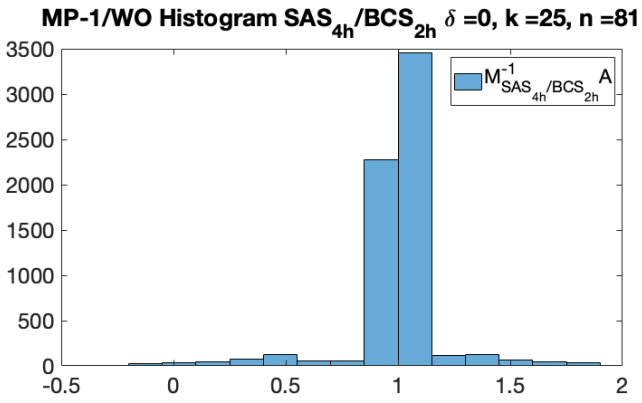
(a) $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ eigenvalue plot, $n = 81$, $k = 25$



(b) $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ histogram of eigenvalues, $n = 81$, $k = 25$



(c) $\mathrm{SAS}_{4h}/\mathrm{BCS}_{2h}$ eigenvalue plot, $n = 81$, $k = 25$



(d) $\mathrm{SAS}_{4h}/\mathrm{BCS}_{2h}$ histogram of eigenvalues, $n = 81$, $k = 25$

Figure 7.3: Eigenvalue plots and histograms of the $\mathrm{SAS}_{4h}/\mathrm{BCS}_{2h}$ and $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioners for MP-1 with $k = 25$ and $\kappa_h \approx 0.3125$. Note that $\delta = 0$

Subdoms	k	$\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$	$\mathrm{SAS}_{4h}/\mathrm{BCS}_{2h}$
64	10	18	15
400	25	18	15
1600	50	18	15
6400	100	18	15
14400	150	18	15

Table 7.8: NOI results for the $\mathrm{SAS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner on the wedge model problem with $\kappa_h \approx 0.3125$ compared to previous results. Additionally, $\delta = 0$.

7.2.2 Two-level Hybrid Schwarz Preconditioner

Subdoms	k	$\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$	$\mathrm{HS}_{4h}/\mathrm{BCS}_{2h}$
64	10	14	10
400	25	17	11
1600	50	16	10
6400	100	12	9
14400	150	15	10

(a) $H = 4h$

Subdoms	k	$\mathrm{AS}_{8h}/\mathrm{BCS}_{2h}$	$\mathrm{HS}_{8h}/\mathrm{BCS}_{2h}$
16	10	13	8
100	25	22	9
400	50	25	9
1600	100	18	8
3600	150	22	9

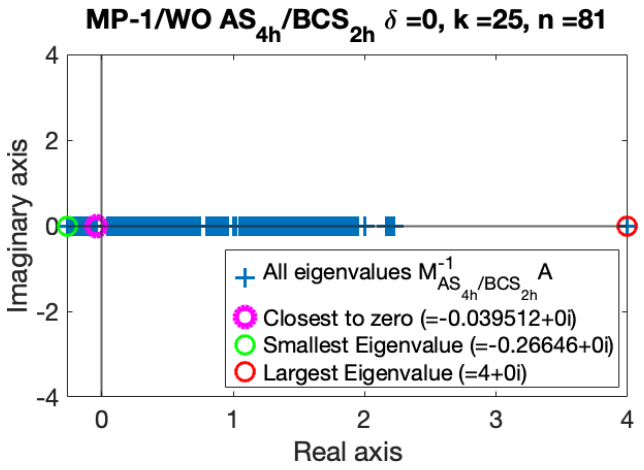
(b) $H = 8h$

Subdoms	k	$\mathrm{AS}_{16h}/\mathrm{BCS}_{2h}$	$\mathrm{HS}_{16h}/\mathrm{BCS}_{2h}$
4	10	10	7
25	25	16	8
100	50	21	8
400	100	16	7
900	150	20	8

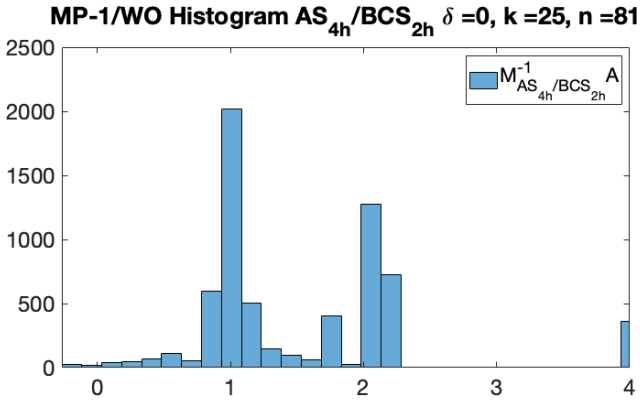
(c) $H = 16h$

Table 7.9: NOI results for MP-1 using the $\mathrm{HS}_H/\mathrm{BCS}_{2h}$ preconditioner with $H = 4h$, $H = 8h$ or $H = 16h$, compared to previous results. Additionally, $\kappa_h \approx 0.3125$ and $\delta = 0$.

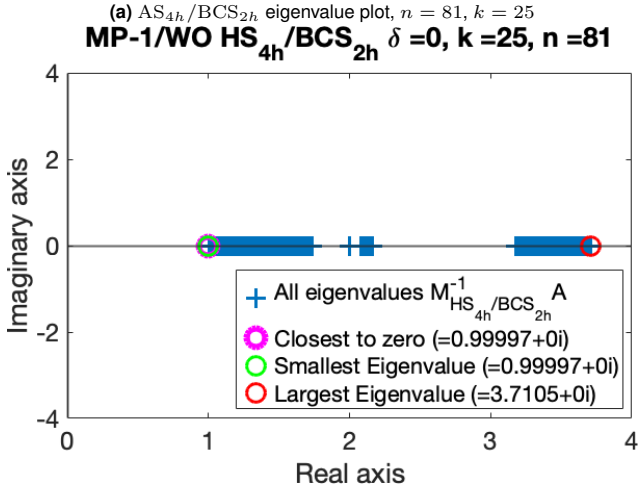
In the final research question given in the introduction, only SAS and SHS are mentioned in the sub-questions. In this subsection the $\mathrm{HS}_H/\mathrm{BCS}_{CH}$ is analysed, in order to understand the impact of the $\mathrm{HS}_H/\mathrm{BCS}_{CH}$ preconditioner and therefore acquire a better understand of the SHS preconditioner. With the scaling not being applied to the preconditioner, the eigenvalue plots should nicely show the deflation of the eigenvalues.
The two-level HS preconditioner from Equation (4.20) has a higher computational cost than the two-level AS preconditioner. This is a consequence of the additional large matrix-vector operations in the deflation part of the preconditioner. The numerical results in Table 7.9 are wave number independent for $CH = 2$ with $\kappa_h \approx 0.3125$. Additionally, contrary to the results in Section 7.1, increasing the subdomain size H lowers the NOI required to reach convergence. Being able to change n^l, while remaining to get NOI results that are wave number independent, is a big deal for making use of parallel computing, since the amount of available processors often decides the amount of subdomains.
Increasing the overlap still only deteriorates the performance of the preconditioner in the case of $CH = 2h$. This can be seen from the numerical results in Table C.7 in the appendix. All the respective NOI results are worse or the same with $\delta = 1$.
The eigenvalue plot and the eigenvalue histogram in Figure 7.4 (**c**) and (**d**) give a nice illustration of deflation. When comparing Figure 7.4 (**a**) and Figure 7.4 (**c**) it can be seen that the low frequency eigenvalues (in this case $\{\lambda_i \in \lambda \mid 1 \leq i \leq (n^2-1)/2\}$, due to the BCS_{2h} coarse problem) are deflated towards 1. The remaining high frequency eigenvalues seem mostly untouched. A slightly different eigenvalue clustering around 4 can be seen when comparing the eigenvalue plots and histograms of the eigenvalues of $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ and $\mathrm{HS}_{4h}/\mathrm{BCS}_{2h}$.
The clustering due to deflation is favourable and could be an explanation for the improved performance in Table 7.9, since the literature about eigenvalue clustering for Helmholtz solvers suggests that eigenvalues close to 0 could deteriorate the performance of the Helmholtz solver [42].



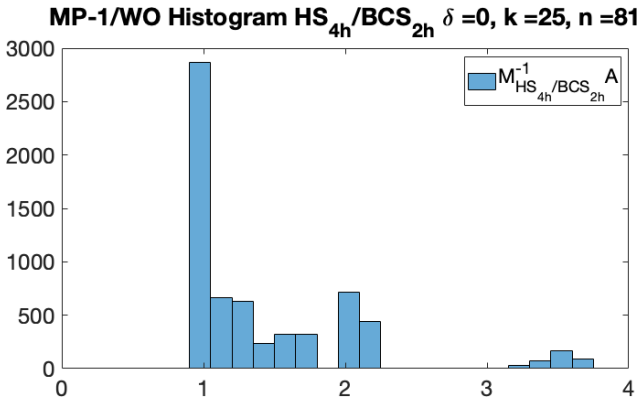
(a) $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ eigenvalue plot, $n = 81$, $k = 25$



(b) $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ histogram of eigenvalues, $n = 81$, $k = 25$



(c) $\mathrm{HS}_{4h}/\mathrm{BCS}_{2h}$ eigenvalue plot, $n = 81$, $k = 25$



(d) $\mathrm{HS}_{4h}/\mathrm{BCS}_{2h}$ histogram of eigenvalues, $n = 81$, $k = 25$

Figure 7.4: Eigenvalue plots and histograms of the $\mathrm{HS}_{4h}/\mathrm{BCS}_{2h}$ and $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioners for MP-1 with $k = 25$ and $\kappa_h \approx 0.3125$. Note that $\delta = 0$

Figures 7.5 and C.4 investigate κ_{CH} of the $\mathrm{HS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner and the $\mathrm{HS}_{8h}/\mathrm{BCS}_{2h}$ preconditioner, respectively. From Figure 7.5 and C.4 it can be seen that both $\mathrm{HS}_{4h}/\mathrm{BCS}_{2h}$ and $\mathrm{HS}_{8h}/\mathrm{BCS}_{2h}$ should perform well with $\kappa_{CH} \approx 0.625$. Figure 7.5 also shows that κ_h can even be increased for the $\mathrm{HS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner. The results of the wedge model problem are given below in Table 7.10. The Helmholtz solver using $\mathrm{HS}_H/\mathrm{BCS}_{2h}$ solves in NOI required to reach convergence that is independent of the wave number.

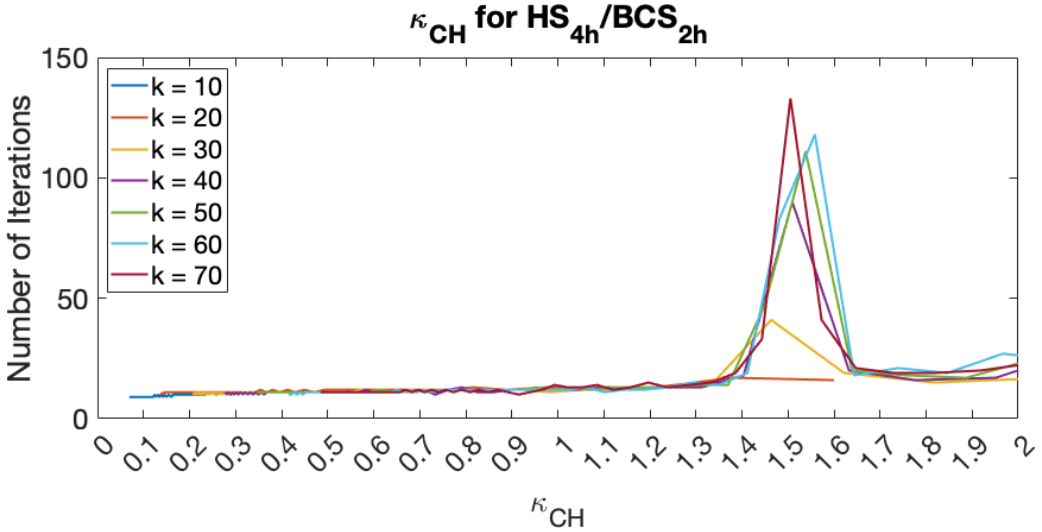


Figure 7.5: $\mathrm{HS}_{4h}/\mathrm{BCS}_{2h}$ on MP-1

Subdoms	k	$\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$	$\mathrm{HS}_{4h}/\mathrm{BCS}_{2h}$
64	10	18	13
400	25	18	13
1600	50	18	12
6400	100	18	12
14400	150	18	12

(a) $H = 4h$

Subdoms	k	$\mathrm{AS}_{8h}/\mathrm{BCS}_{2h}$	$\mathrm{HS}_{8h}/\mathrm{BCS}_{2h}$
16	10	13	11
100	25	22	11
400	50	25	11
1600	100	18	11
3600	150	22	11

(b) $H = 8h$

Table 7.10: NOI results for the $\mathrm{HS}_{4h}/\mathrm{BCS}_{2h}$ and $\mathrm{HS}_{8h}/\mathrm{BCS}_{2h}$ preconditioner on the wedge model problem with $\kappa_h \approx 0.3125$, compared to previous results. Additionally, $\delta = 0$.

7.2.3 Two-level Scaled Hybrid Schwarz Preconditioner

The eigenvalue analysis of using a two-level SAS preconditioner showed that such a preconditioner results in an eigenvalue clustering with an upper bound of 2. The low frequency eigenvalues are clustered in a similar way as with the AS preconditioner. The eigenvalue analysis of the HS preconditioner showed the deflation of the low frequency eigenvalues (See Figure 7.7 (**b**)). In this subsection, the SAS and HS preconditioners are combined into the two-level SHS preconditioner.

Using the $\mathrm{SHS}_H/\mathrm{BCS}_{2h}$ preconditioner on MP-1 gives results that are wave number independent for $\kappa_h \approx 0.3125$, presented in Table 7.11. Additionally, the NOI required to reach convergence is the lowest up to this point, which allows the numerical results to go up to $k = 500$, the highest wave number up to this point. Similarly as was seen with the $\mathrm{HS}_H/\mathrm{BCS}_{2h}$ preconditioner, changing the number of nodes per standard subdomain does not result in the preconditioner deteriorating in performance.

Subdoms	k	$\mathrm{SHS}_{4h}/\mathrm{BCS}_{2h}$
64	10	5
400	25	6
1600	50	5
6400	100	4
14400	150	5
40000	250	5
160000	500	5

(a) $H = 4h$

Subdoms	k	$\mathrm{SHS}_{8h}/\mathrm{BCS}_{2h}$
16	10	5
100	25	6
400	50	5
1600	100	5
3600	150	6
10000	250	6
40000	500	5

(b) $H = 8h$

Table 7.11: NOI results for the $\mathrm{SHS}_{4h}/\mathrm{BCS}_{2h}$ and $\mathrm{SHS}_{8h}/\mathrm{BCS}_{2h}$ preconditioner on MP-1 with $\kappa_h \approx 0.3125$. Additionally, $\delta = 0$.

While the $\mathrm{SHS}_H/\mathrm{BCS}_{2h}$ preconditioner does yield NOI results that are wave number independent and low, the preconditioner does run into a computational bottleneck. This computational bottleneck is the size of the coarse problem. For $k = 500$ in Table 7.11 we have $n^* = 1601$, resulting in $n = 1601^2$ and $n^c = 801^2$. The coarse problem is too large in this case. To be more specific, when solving for $k = 500$, the LU-decomposition uses more than 90% of the total computational time. Figure 7.6 gives runtime results for the $\mathrm{SHS}_H/\mathrm{BCS}_{2h}$ preconditioner, which illustrate the computational time problem. The eigenvalue distribution of Figure 7.7 shows what we would expect from the SHS preconditioner. The clustering of the eigenvalues of the preconditioned system is favourable and coincides with the good performance of the $\mathrm{SHS}_H/\mathrm{BCS}_{CH}$ preconditioner in Table 7.11. In Figure 7.7 (**a**) and (**c**) we see that around 1 the eigenvalues become slightly imaginary. This is a consequence of using the scaled versions on the preconditioners. The preconditioning matrix of the SHS and SAS preconditioner are not symmetric anymore due to the scaling.

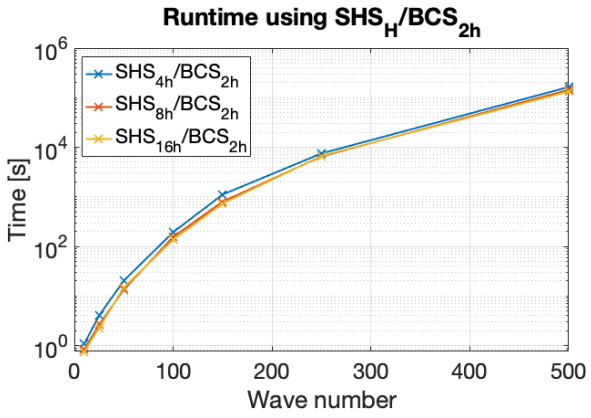
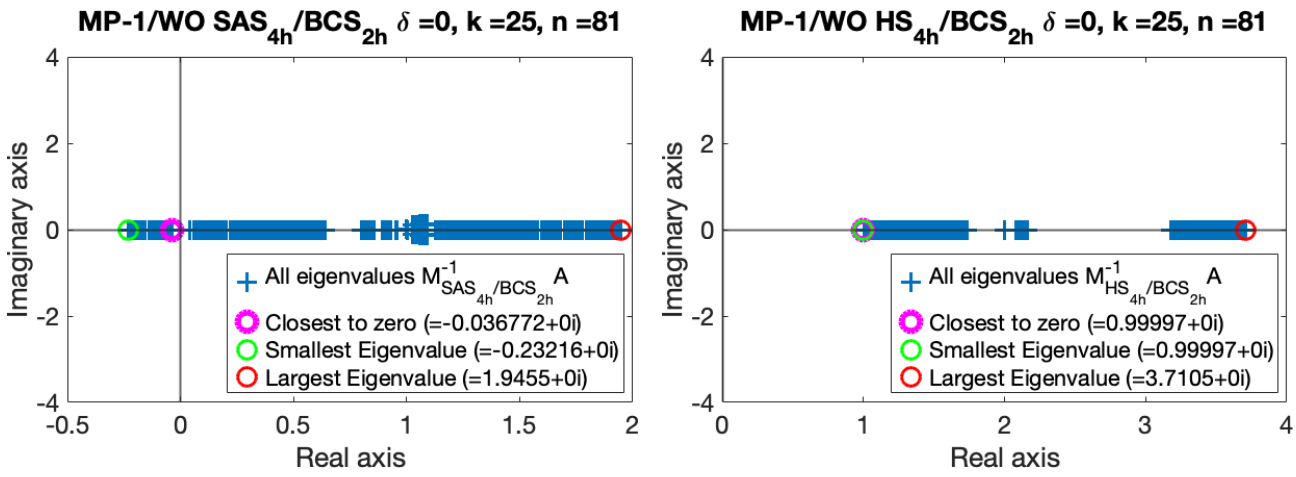
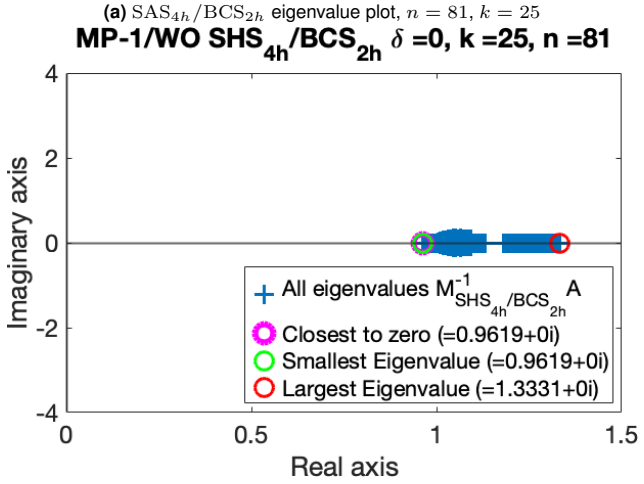


Figure 7.6: Runtime plot for the $\mathrm{SHS}_H/\mathrm{BCS}_{2h}$ preconditioner on the wedge model problem.

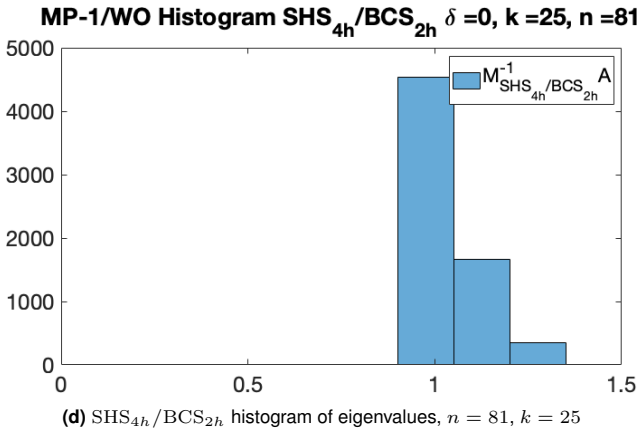


(a) $\mathrm{SAS}_{4h}/\mathrm{BCS}_{2h}$ eigenvalue plot, $n = 81$, $k = 25$

(b) $\mathrm{HS}_{4h}/\mathrm{BCS}_{2h}$ eigenvalue plot, $n = 81$, $k = 25$



(c) $\mathrm{SHS}_{4h}/\mathrm{BCS}_{2h}$ eigenvalue plot, $n = 81$, $k = 25$



(d) $\mathrm{SHS}_{4h}/\mathrm{BCS}_{2h}$ histogram of eigenvalues, $n = 81$, $k = 25$

Figure 7.7: Eigenvalue plots of the $\mathrm{SAS}_{4h}/\mathrm{BCS}_{2h}$, $\mathrm{HS}_{4h}/\mathrm{BCS}_{2h}$ and $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioners for MP-1 with $k = 25$ and $\kappa_h \approx 0.3125$. Note that $\delta = 0$

Figures 7.8 and C.5 give κ_{CH} plots of the $\mathrm{SHS}_H/\mathrm{BCS}_{CH}$ preconditioner. These plots show a similar behavior as seen with the $\mathrm{HS}_H/\mathrm{BCS}_{CH}$ preconditioner, since the figures show that κ_h can be increased, while the NOI results stay wave number independent as the number of subdomains increase with the wave number. Figures 7.8 and C.5 also show that κ_{CH} depends on n^l. As a final note on Figures 7.8, increasing CH to $CH = 4h$ does not seem possible for the $\mathrm{SHS}_{4h}/\mathrm{BCS}_{CH}$ preconditioner, since Figures 7.8 suggests that this would make the solver have wave number dependent convergence.

As is almost expected at this moment, both the $\mathrm{SHS}_{4h}/\mathrm{BCS}_{2h}$ and $\mathrm{SHS}_{8h}/\mathrm{BCS}_{2h}$ perform well on the wedge problem. In Table 7.12 the NOI results are wave number independent and go all the way up to $k = 500$ with $\kappa_h \approx 0.3125$.

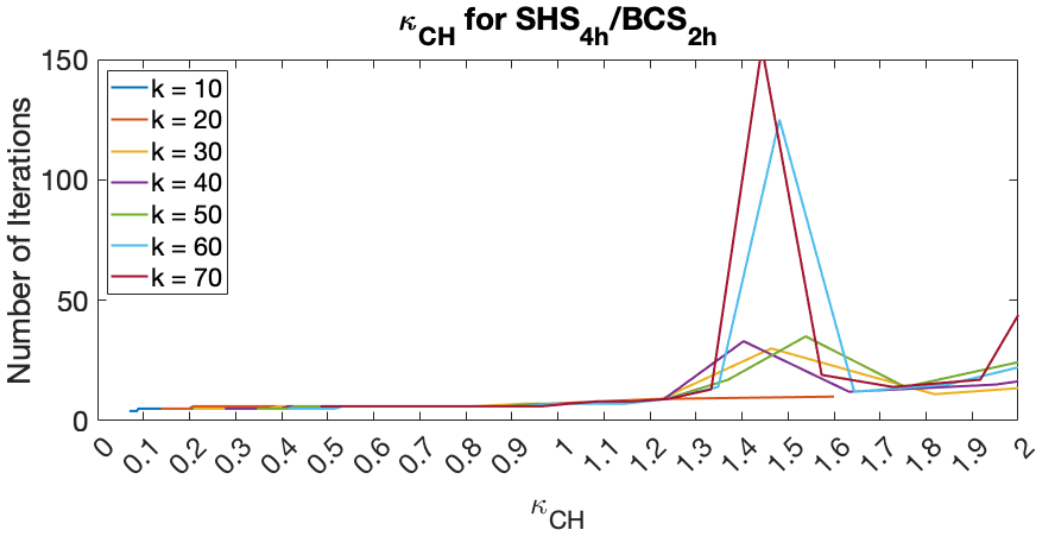


Figure 7.8: $\mathrm{SHS}_{4h}/\mathrm{BCS}_{2h}$ on MP-1

Subdoms	k	$\mathrm{SHS}_{4h}/\mathrm{BCS}_{2h}$
64	10	7
400	25	7
1600	50	7
6400	100	7
14400	150	7
40000	250	6
160000	500	6

(a) $H = 4h$

Subdoms	k	$\mathrm{SHS}_{8h}/\mathrm{BCS}_{2h}$
16	10	7
100	25	7
400	50	7
1600	100	7
3600	150	7
10000	250	7
40000	500	7

(b) $H = 8h$

Subdoms	k	$\mathrm{SHS}_{8h}/\mathrm{BCS}_{2h}$
4	10	6
25	25	8
100	50	9
400	100	7
900	150	9
25000	250	10
10000	500	7

(c) $H = 8h$

Table 7.12: NOI results for the $\mathrm{SHS}_{4h}/\mathrm{BCS}_{2h}$, $\mathrm{SHS}_{8h}/\mathrm{BCS}_{2h}$ and $\mathrm{SHS}_{8h}/\mathrm{BCS}_{2h}$ preconditioner on the wedge model problem with $\kappa_h \approx 0.3125$. Additionally, $\delta = 0$.

7.3 Convergence Plot

To get a little more information out of the developed and existing preconditioners it is valuable to look at the convergence plot of the preconditioners for MP-1. The relative residuals convergence plot is given in Figure 7.9. The figure also shows the convergence rate of two one-level Schwarz preconditioners. These are the one-level AS preconditioner, indicated in the legend of Figure 7.9 by $\mathrm{AS}\,1_H$, and the one-level SAS preconditioner, indicated in the legend of Figure 7.9 by $\mathrm{SAS}\,1_H$. These do not make use of any coarse problem by definition. A clear difference can be seen between the two-level and one-level preconditioners in the figure on the right.

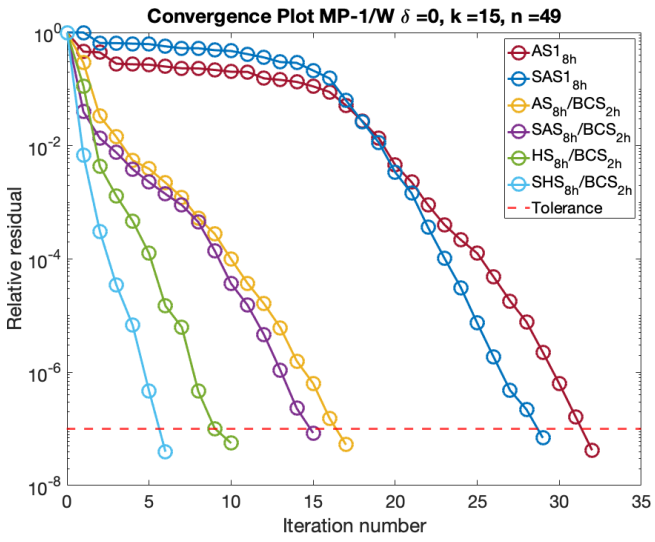


Figure 7.9: Convergence plot for different preconditioners on MP-1 with $\kappa_h \approx 0.3125$, $H = 8h$ and $\delta = 0$.

7.4 Concluding Remarks and Summary

This section has the goal of capturing the main findings from the numerical experiments and additional analysis of this chapter.

- The $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner is wave number independent for $\kappa_h \approx 0.3125$ on MP-1. Also, the $\mathrm{AS}_{H}/\mathrm{BCS}_{2h}$ preconditioner is wave number independent for but then with $\kappa_h \approx 0.15625$ on MP-1. Both preconditioner have the same n^c.
- For MP-1, comparing the eigenvalue plots of the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ and $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$ preconditioners show no conclusive results that would allow the explanation of the better performance of the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner for MP-1. The upper bound for the eigenvalues on MP-1 is 4.
- the $\mathrm{SAS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner has a lower NOI then the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner, while still having wave number independent convergence with $\kappa_h \approx 0.3125$.
- The eigenvalue plot of the $\mathrm{SAS}_{4h}/\mathrm{BCS}_{2h}$ preconditioned system shows a clustering of eigenvalues with an upper bound of 2.
- The $\mathrm{HS}_{H}/\mathrm{BCS}_{2h}$ preconditioner gives NOI results that are wave number independent for $\kappa_h \approx 0.3125$.
- The eigenvalue plot of the $\mathrm{HS}_{4h}/\mathrm{BCS}_{2h}$ preconditioned system shows a clustering of eigenvalue with a lower bound around 1, due to the deflation of the low frequency eigenvalues. Additionally, the eigenvalues have an upper bound slightly below 4.
- The lowest number of iterations in combination with wave number independence is found when using the $\mathrm{SHS}_{H}/\mathrm{BCS}_{2h}$ preconditioner with $\kappa_h \approx 0.3125$. Additionally, the low iteration number allows the numerical results to go up to $k = 500$.
- Using the $\mathrm{SHS}_{H}/\mathrm{BCS}_{2h}$ preconditioner results in eigenvalues clustered between approximately 1 and 1.34, which is favourable for GMRES.
- The convergence rate of the $\mathrm{SHS}_{H}/\mathrm{BCS}_{2h}$ solver is almost linear in the semilog relative residual plot.
- A limitation of all the mentioned preconditioners is the quickly increasing computational cost when the wave number increases. This increasing computational cost is mainly caused by solving a large coarse problem with a direct method when the wave number becomes large. The coarse problem has a coefficient matrix which is also indefinite.
  Currently, the coarse problem is becoming large when the wave number becomes large. In this thesis we looked at ways to lower the number of coarse grid nodes, while keeping the wave number independence in the NOI results. Unfortunately, finding a way to prevent the coarse problem from becoming large has been unsuccessful. Using BCS_{CH} or $\mathbb{P}_1\text{-}\mathrm{CS}_{CH}$, for a fixed number of subdomains, fixed CH and fixed n^l, changes nothing about the number of coarse grid nodes. The difference in BCS_{CH} and $\mathbb{P}_1\text{-}\mathrm{CS}_{CH}$ is not in n^c, but in the density of R_0^T and R_0. Therefore, the computational cost of direct solving of the BCS_{CH} coarse problem and the $\mathbb{P}_1\text{-}\mathrm{CS}_{CH}$ coarse problem are roughly the same.

8 | Conclusion & Discussion

In this chapter, the conclusions of this thesis are presented and discussed. Firstly, a recap of the research goal, questions and method are given. Next, conclusions are drawn about domain decomposition preconditioner Helmholtz solvers using a first-order coarse problem and a higher-order Bézier coarse problem in Subsection 8.1.1 and 8.1.2. In Section methods used in this research and their results are discussed. Additionally, recommendations for future research are given in this section.

8.1 Conclusion

While it has been possible to solve Helmholtz problems for medium sized wave numbers, it has not been possible to simulate the Helmholtz equation with a large wave numbers on more realistic types of problems that appear in diverse scientific and engineering disciplines. The reason for this is that all current numerical Helmholtz solvers are not numerically scalable and efficient at the same time. The goal of this thesis was to develop a two-level Schwarz preconditioner Helmholtz solver that is both efficient and wave number independent. The constructed Helmholtz solvers all consists of three main parts. The first main part is the use of GMRES as the iterative method. Another part of the solvers are the different variants of two-level Schwarz domain decomposition preconditioners to accelerate the convergence of the iterative method. And the final part of the solvers is the grid coarse problem, which is constructed using either first-order or a higher-order Bézier interpolation.
Using paralellizable domain decomposition methods in a Helmholtz solver has the potential to be computationally fast when parallelization is implemented. This research only makes use of sequential computing, but parallel computing can be essential for linear time complexity.
In the introduction of this thesis research questions and research sub-questions were given. These questions were:

1. Does a Helmholtz solver constructed using a two-level additive Schwarz preconditioner combined with a **first-order** grid coarse problem show **wave number independent convergence** and **efficiency**?
    - How can the performance of the Helmholtz solver using a first-order grid coarse problem be explained?
2. Does a Helmholtz solver, constructed using a two-level additive Schwarz preconditioner and a **higher-order Bézier** coarse problem, show **wave number independent convergence** and **efficiency**?
    - What causes the solver to be inscalable if this is the case?
3. Does a Helmholtz solver, constructed using different two-level Schwarz preconditioners and a **higher-order Bézier** coarse problem, show **wave number independent convergence** and **efficiency**?
    - For the scaled additive Schwarz preconditioner, does the solver show wave number independent convergence and efficiency and if so, why?
    - For the scaled hybrid Schwarz preconditioner which uses deflation, does the solver show wave number independent convergence and efficiency and if so, why?

One of the main conclusions of this thesis is that we developed a two-level additive Schwarz preconditioner which leads to wave number independent convergence for wave numbers in 2D and 3D, while remaining to have accurate numerical solutions. The preconditioner uses a coarse problem which is constructed using higher-order interpolation with quadratic rational Bézier curves. The other main conclusion of this thesis is that with the different two-level Schwarz preconditioner and a higher-order Bézier grid coarse problem the number of iterations required for the iterative solver to reach convergence becomes even lower, while the number of iterations remain independent of the wave number when increasing the number of subdomains with the wave number.

8.1.1 Using a First-Order Coarse Problem

The goal of chapter 6 was to answer research question 1 and its sub-question. From this chapter, it can be concluded that a Helmholtz solver constructed using a two-level additive Schwarz preconditioner combined with first-order grid coarse problem does not show wave number independence and efficiency for MP-1, the

Helmholtz benchmark problem. The chapter also shows that κ_h influences the rate at which the NOI results increase as the number of subdomains and wave number increases.
Analyzing the clustering of the eigenvalues gives some insight into the performance of the solver using a first-order coarse problem. It also allows for the comparison of eigenvalue distributions of the preconditioners using a higher-order Bézier coarse problem.

8.1.2 Using a Higher-Order Bézier Coarse Problem

Chapter 7, answers research questions 2 and 3 and their sub-questions. From the findings in this chapter, the main conclusions from above are drawn. The additional conclusions of this chapter are the following. Firstly, for the two-level additive Schwarz preconditioner wave number independent convergence is found. The lowest number of iterations required to reach convergence is around 15. Additionally, the eigenvalue distributions give no clear reason that explains wave number independent convergence when comparing the eigenvalue distribution to the eigenvalue distribution when using preconditioner with a first-order coarse problem.
Thirdly, when using two-level hybrid Schwarz or scaled hybrid Schwarz preconditioner on Helmholtz problems we can be more flexible in the choice of subdomain size, which is not the case when using the two-level additive Schwarz preconditioner. This is important for the practicality of the preconditioners when parallel computing is used.
One of the main drawbacks of the preconditioners with wave number independent convergence in this research is that their coarse problem size becomes large, resulting in high computational costs. However, there seems to be no good coarsening alternative to the coarse grid size of $CH = 2h$. Hence, for wave number independent convergence $CH = 2h$ is chosen. For most of the preconditioners from this thesis, the choice of $CH = 2h$ is also required to be combined with $\kappa_h \approx 0.3125$ in order to have wave number independent convergence. The κ_{CH} plots of the $\mathrm{SHS}_H/\mathrm{BCS}_{CH}$ preconditioner and the $\mathrm{HS}_H/\mathrm{BCS}_{CH}$ preconditioner showed that $\kappa_h \approx 0.3125$ can be increased up to a certain upper bound. However, this upper bound does dependent on n^l. Increasing κ_h results in the number of subdomains and the coarse problem size increasing less rapidly when the wave number increases, and thus lowers the overall computational cost of the Helmholtz solver. While this does lower the computational cost, we will run into the same computational cost problem when the wave number is increased even more. This is why linear time complexity is important.

As with the sub-question of research question 1, it is difficult to be conclusive when answering the sub-questions of research questions 1 and 2. Comparing the eigenvalue distributions of preconditioners with different two-level Schwarz preconditioners provides use with insight its behaviour and performance. Using the scaled additive Schwarz preconditioner results in the high frequency eigenvalues of the preconditioned linear system being lower compared to the high frequency eigenvalues of the additive Schwarz preconditioned linear system. The eigenvalue clustering, as a consequence of the scaled additive Schwarz preconditioner, have an upper bound around 2. The Helmholtz problem numerical results of the scaled additive Schwarz preconditioner show a lower number of iterations required to reach convergence, which is likely due to this more favourable clustering.
As a consequence of using the hybrid Schwarz preconditioner, the low frequency eigenvalues of the preconditioned linear system are no longer close to 0, but close to 1. This is a consequence of the deflation in the hybrid Schwarz preconditioner. This results in an even better performance of the iterative method when it comes to the number of iterations required to reach convergence. Again, this is likely due to the clustering of the eigenvalues, specifically, the eigenvalues are now clustered further away from 0. The literature on Helmholtz solvers tells us that these close to zero eigenvalues are often the reason for numerical inscalability and poor performance.
Combining both the scaled and hybrid methods into the scaled hybrid Schwarz method results in a clustering of the eigenvalues roughly between 1 and 1.4. This eigenvalue clustering is the reason the solver using the scaled hybrid Schwarz preconditioner requires the least amount of iterations to reach convergence, while remaining wave number independent with $\kappa_h \approx 0.3125$.

8.2 Discussion

This section focuses on discussing the research method, results and conclusion while also providing directions for future research.

Time Complexity

The ultimate goal in the development of a Helmholtz solver is to develop a numerical solver that is both numerically scalable and efficient. As mentioned in the introduction, for this research, the choice was made to ignore time complexity. Future research could therefore focus on investigating the time complexity of the solvers from this research and investigate how to improve the solver's computational cost in order to reach numerical scalability.
The choice of using parallelizable Schwarz domain decomposition methods was made because it allows the leveraging of parallel computing in order eventually get a numerically scalable and efficient Helmholtz solver. It would be interesting to see how the solver performs in parallel and where computational gains are with parallel computing. Solving the local subdomain problems sequentially when this can be done in parallel is inefficient and computationally costly. However, it is not the biggest computational bottle neck in the solvers developed in this research, and therefore it is not yet necessary to go to high performance computing and parallelization yet. Note that, the computational cost of the local subproblems does depend on the size of the subdomains.
Currently, we solve the coarse problem by using LU-decomposition and Gaussian elimination. The coarse problem increases in size as the wave number and fine grid increase, eventually resulting in the direct solution method becoming very slow. The current biggest bottleneck of the preconditioners is the large coarse problem size. Solving the MP-1 problem using the $\mathrm{SHS}_{4h}/\mathrm{BCS}_{2h}$ preconditioner with $k = 500$ and $\kappa_h = 0.3125$ takes a very long time, of which more than 90% is spend on the LU-decomposition of the coarse problem coefficient matrix. Ideally, we would coarsen the coarse grid, while remaining to get NOI results that are wave number independent, but this seems impossible. Even when the coarse grid is coarsened by a factor 2 and κ_h is lowered by a factor of 2, the number of iterations results are not wave number independent anymore, while the coarse problem has actually not changed in total number of grid nodes.
The large and quickly increasing computational cost of the preconditioner when the wave number increases influences the numerical experiments performed in this research. It would be beneficial to be able to perform numerical experiments for larger wave number in order to make sure that the wave number independence convergence also holds for $k > 500$.
In the future, the direct solution method for the coarse problem could be replaced by a flexible iterative solution method or a multigrid method. The coarse problem matrix that results from the higher-order Bézier interpolation operator is also indefinite for large wave numbers. Therefore, if a flexible iterative method is used for the coarse problem, a wave number independent and efficient Helmholtz preconditioner needs to be used to accelerate the flexible iterative method of the coarse problem. The other method that can replace the direct method is the multigrid method. A multigrid method is used for approximating the coarse space in [42, 43]. Something similar could be implemented in the solvers developed in this research.
For future research it might also be interesting to look into constructing a multi-level Schwarz domain decomposition method, similar to how a multi-level Helmholtz solver was built in [43].
As a final suggestion for future research on the topic of time complexity, using an inexact solution method for the subdomains can be investigated and implemented [45]. If the local subproblems become too large a direct solution method becomes too computationally costly. Specifically for 3D high frequency Helmholtz problems using a two-level deflation preconditioner with FGMRES for the subdomains shows promising results and computational cost benefits, even for a low relative residual tolerance.

Finite Difference Method and Complex Geometries

Since this research is about developing a Helmholtz numerical solver, the choice was made to use a finite difference method to discretize the Helmholtz boundary value problem. One of the big limitations of the finite difference method is that it is difficult to use when the problem involves complex geometries. Ultimately, Helmholtz solvers need to be used on real scientific and industrial problems, which could include more complex geometries.
Hence, for a numerically scalable and efficient Helmholtz solver this research can be extended to finite element method. In order to do this, it should also be investigated if and how the higher-order Bézier coarse interpolations can still be used in a finite element method setting.

Schwarz Preconditioner Analysis for the Helmholtz Problem

When comparing the two-level additive Schwarz preconditioner for both a first- and a higher-order coarse problem on MP-1, we see a difference in performance of the solvers in the numerical results. On the other hand, looking at the eigenvalue distributions no clear and conclusive differences are found that would explain this difference in performance of the preconditioners. This is most likely due to the eigenvalues of the preconditioned linear system not solely being descriptive for the performance of GMRES for the Helmholtz problem. Another possible explanation is that the close to zero eigenvalues, which are not closely investigated in this research,

are the reason for the varying performance depending on the coarse space.

Contrary to the first- and higher-order coarse problem, the various Schwarz preconditioners show clear differences in eigenvalue distribution. We can therefore empirically learn more about the spectral influence on the convergence of GMRES. While this is only empirically, it appears from the results that clustering the eigenvalues more compactly and away from 0 improves the convergence of GMRES for the Helmholtz problem. In order to get a better understanding of the performance of the preconditioners in this research, spectral analysis of the Schwarz preconditioners has to be performed. This would lead to a better understanding of the desired spectral properties and therefore a more focused development of a two-level Schwarz Helmholtz solvers.
The eigenvalue plots in this research are limited to the Helmholtz problem with a homogeneous Dirichlet boundary condition. This problem results in a coefficient matrix of the linear system which is real, symmetric and indefinite. For the Helmholtz problem with a Sommerfeld radiation condition the coefficient matrix becomes indefinite, complex, symmetric, but non-Hermitian. The Schwarz preconditioner analysis could change for these different types of matrices. For future research it is therefore also interesting to investigate the eigenvalues of the Helmholtz problem with a Sommerfeld radiation condition.
For the Laplace problem, Schwarz domain decomposition convergence theory has been developed. Laplace Schwarz domain decomposition convergence theory might have some content that also holds true for the Helmholtz problem.

High Performance Computing

As already mentioned, in order to bring the solvers forward in engineering usability, it is important to look at parallel implementation of the solver to allow for scalability. For parallel implementation, it would make sense to make use of existing frameworks. One possibility is to make use of the FROSch package [51], which allows for parallel scalability up to more than 100.000 cores. This way we do not have to implement a parallel Helmholtz solver from scratch.

Additional Applications

The time-harmonic Maxwell equation and the time-harmonic elastic wave equation are similar to the Helmholtz equation. Often, we see effective Helmholtz solvers being used or extended to these other time-harmonic wave propagation problems. It would be interesting to see how the solvers from this research perform on time-harmonic Maxwell equation and the time-harmonic elastic wave equation problems and see how effective the Helmholtz solvers are.

A | Higher-Order Bézier Coarse Grid Interpolation

One dimensional higher-order Bézier coarse grid interpolation operators for larger CH can be introduced similarly as for the $CH = 2h$ case shown in Chapter 5. The details are again given for a one dimensional domain, since the interpolation operators for higher dimensional domains follow in a tensor product way.

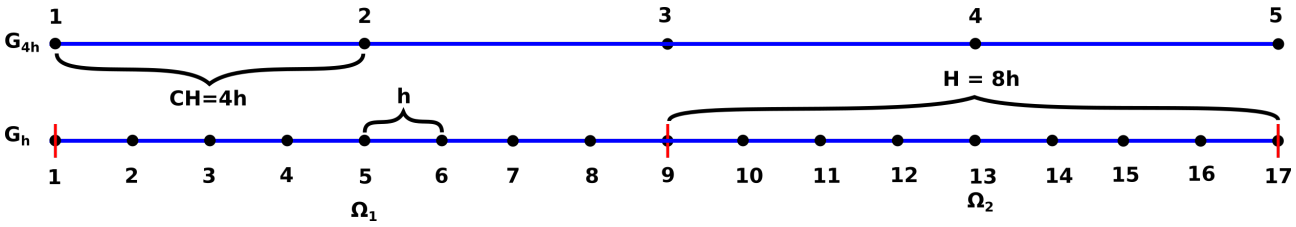


Figure A.1: Fine and coarse 1D grid layout with $H = 8h$ and $CH = 4h$. The fine grid is partitioned into two subdomains indicated by Ω_1 and Ω_2 with grid node 9 being shared by both subdomains.

The first step of constructing the higher-order Bézier coarse grid interpolation operators is to construct the first-order interpolation. In the case of $CH = 4h$, the following first-order interpolation extension operators for G_{4h} is given by

$$I_{4h}^{h}\left[u_{4h}\right]_i = \left\{ \begin{array}{ll} \left[u_{4h}\right]_{(i+3)/4} & \text{if } i = 1,5,9,...,n \\ \frac{3}{4}\left[u_{4h}\right]_{(i+2)/4} + \frac{1}{4}\left[u_{4h}\right]_{((i+2)/4)+1} & \text{if } i = 2,6,10,...,n-3 \\ \frac{1}{2}\left[u_{4h}\right]_{(i+1)/4} + \frac{1}{2}\left[u_{4h}\right]_{((i+1)/4)+1} & \text{if } i = 3,7,11,...,n-2 \\ \frac{1}{4}\left[u_{4h}\right]_{(i)/4} + \frac{3}{4}\left[u_{4h}\right]_{((i)/4)+1} & \text{if } i = 4,8,12,...,n-1 \end{array} \right\} \tag{A.1}$$

for $i = 1, ..., n$ and with $m = 4i - 3$ the first-order interpolation restriction operator is given by

$$I_{h}^{4h}\left[u_h\right]_i = \frac{1}{4}\left[u_h\right]_{(m-3)} + \frac{1}{2}\left[u_h\right]_{(m-2)} + \frac{3}{4}\left[u_h\right]_{(m-1)} + \left[u_h\right]_m + \frac{3}{4}\left[u_h\right]_{(m+1)} + \frac{1}{2}\left[u_h\right]_{(m+2)} + \frac{1}{4}\left[u_h\right]_{(m+3)} \tag{A.2}$$

for $i = 1, ..., \frac{(n+3)}{4}$

Next, the direct under that coarse grid node is replaced the higher-order Bézier approximation scheme from Chapter 5. This results in the higher-order Bézier extension operator for G_{4h} being The first step of constructing the higher-order Bézier coarse grid interpolation operators is to construct the first-order interpolation. In the case of $CH = 4h$, the following first-order interpolation extension operators for G_{4h} is given by

$$I_{4h}^{h}\left[u_{4h}\right]_i = \left\{ \begin{array}{ll} \left(\frac{1}{8}\left[u_{4h}\right]_{((i+3)/4)-1} + \left(\frac{3}{4}\right)\left[u_{4h}\right]_{(i+3)/4} + \frac{1}{8}\left[u_{4h}\right]_{((i+3)/4)+1}\right) & \text{if } i = 1,5,9,...,n \\ \frac{3}{4}\left[u_{4h}\right]_{(i+2)/4} + \frac{1}{4}\left[u_{4h}\right]_{((i+2)/4)+1} & \text{if } i = 2,6,10,...,n-3 \\ \frac{1}{2}\left[u_{4h}\right]_{(i+1)/4} + \frac{1}{2}\left[u_{4h}\right]_{((i+1)/4)+1} & \text{if } i = 3,7,11,...,n-2 \\ \frac{1}{4}\left[u_{4h}\right]_{(i)/4} + \frac{3}{4}\left[u_{4h}\right]_{((i)/4)+1} & \text{if } i = 4,8,12,...,n-1 \end{array} \right\} \tag{A.3}$$

for $i = 1, ..., n$ and with $m = 4i - 3$ the first-order interpolation restriction operator is given by

$$\begin{aligned} I_{h}^{4h}\left[u_h\right]_i = \frac{1}{8}\left[u_h\right]_{(m-4)} + \frac{1}{4}\left[u_h\right]_{(m-3)} + \frac{1}{2}\left[u_h\right]_{(m-2)} + \frac{3}{4}\left[u_h\right]_{(m-1)} + \frac{3}{4}\left[u_h\right]_m + \frac{3}{4}\left[u_h\right]_{(m+1)} \\ + \frac{1}{2}\left[u_h\right]_{(m+2)} + \frac{1}{4}\left[u_h\right]_{(m+3)} + \frac{1}{8}\left[u_h\right]_{(m+4)} \end{aligned} \tag{A.4}$$

for $i = 1, ..., \frac{(n+3)}{4}$ with the corresponding matrices being R_0^T and R_0,respectively.
For larger CH, these steps can followed in similar manner.

B | More Numerical Experiment Results Using a First-Order Grid Coarse Problem

B.1 MP-3

Subdoms	$\mathrm{AS}_{8h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$	$\mathrm{AS}_{8h}/\mathbb{P}_1\text{-}\mathrm{CS}_{4h}$	$\mathrm{AS}_{8h}/\mathbb{P}_1\text{-}\mathrm{CS}_{8h}$
16	13	13	17
64	13	13	17
144	13	13	16
256	13	13	16
400	13	13	16

Table B.1: NOI results for MP-3 using the $\mathrm{AS}_{8h}/\mathbb{P}_1\text{-}\mathrm{CS}_{CH}$ preconditioner with varying CH. Additionally, $\delta = 0$.

Subdoms	$\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$	$\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{4h}$	$\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{8h}$
64	14	15	16
256	15	15	17
576	15	15	17
1024	15	15	16
1600	15	14	16

Table B.2: NOI results for MP-3 using the $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{CH}$ preconditioner with varying CH. Additionally, $\delta = 1$.

In Figure B.1 the eigenvalues of the preconditioned linear systems $\lambda(M^{-1}_{\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}}A)$ are shown to for $\delta = 1$ and $\delta = 2$. From these plots it can be seen that when the overlap becomes too large, the largest eigenvalue changes and the all eigenvalues become larger.

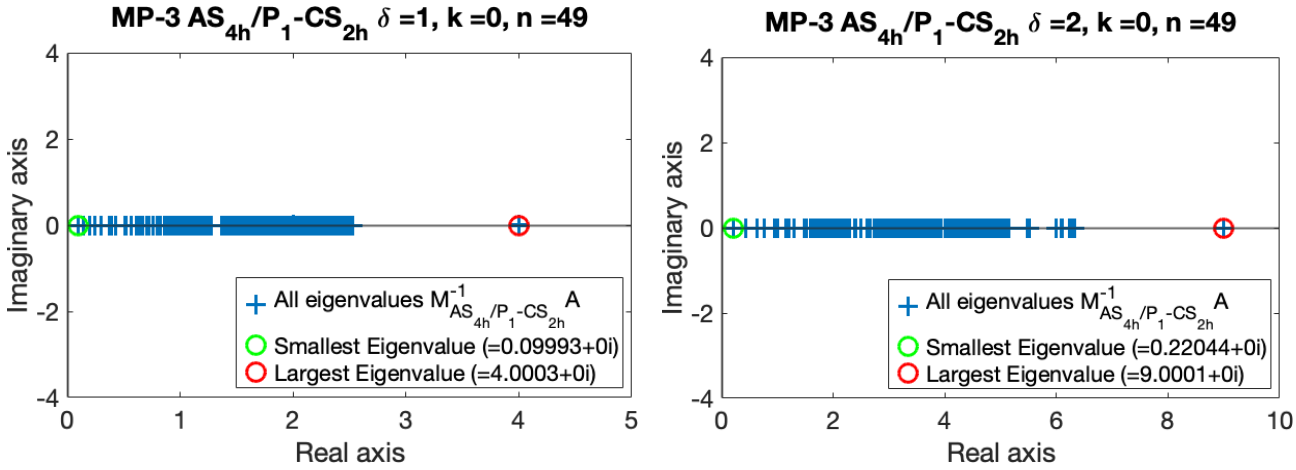
n	$\delta = 0$	$\delta = 1$	$\delta = 2$
64	21	16	17
256	23	17	18
576	22	17	19
1024	22	16	19
1600	22	16	19

(a) $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{8h}$

Subdoms	$\delta = 0$	$\delta = 1$	$\delta = 2$	$\delta = 4$
16	17	15	15	16
64	17	15	15	17
144	16	15	15	18
256	16	15	15	18
400	16	15	15	19

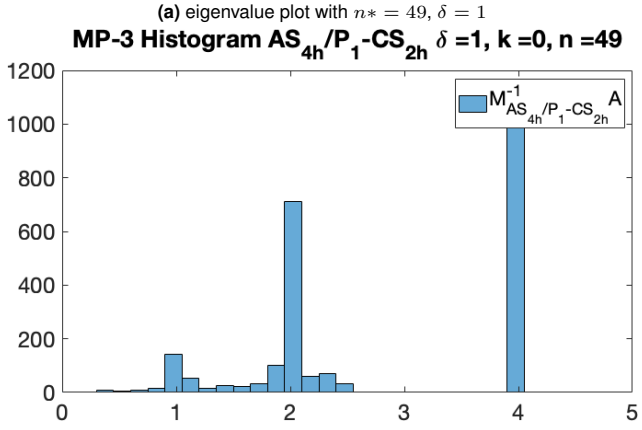
(b) $\mathrm{AS}_{8h}/\mathbb{P}_1\text{-}\mathrm{CS}_{8h}$

Table B.3: NOI for MP-3 for $\mathrm{AS}_H/\mathbb{P}_1\text{-}\mathrm{CS}_{8h}$ preconditioners with different degrees of overlap.

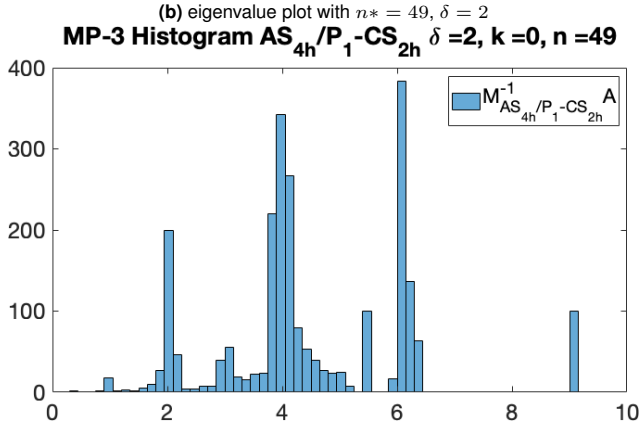


(a) eigenvalue plot with $n* = 49$, $\delta = 1$

(b) eigenvalue plot with $n* = 49$, $\delta = 2$



(c) Histogram for $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$ preconditioner with $n* = 49$, $\delta = 1$



(d) Histogram for $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$ preconditioner with $n* = 49$, $\delta = 2$

Figure B.1: Eigenvalue plots and histograms for $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$ preconditioner on MP-3 with different degrees of overlap.

B.2 MP-1

Subdoms	k	$\mathrm{AS}_{8h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$	$\mathrm{AS}_{8h}/\mathbb{P}_1\text{-}\mathrm{CS}_{4h}$	$\mathrm{AS}_{8h}/\mathbb{P}_1\text{-}\mathrm{CS}_{8h}$
1	5	2	2	2
9	15	16	17	17
25	25	25	29	32
49	35	42	57	69
81	45	62	92	101
100	50	94	128	141
169	65	170	x	x

(a) $\kappa_h \approx 0.625$

Subdoms	k	$\mathrm{AS}_{8h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$	$\mathrm{AS}_{8h}/\mathbb{P}_1\text{-}\mathrm{CS}_{4h}$	$\mathrm{AS}_{8h}/\mathbb{P}_1\text{-}\mathrm{CS}_{8h}$
4	5	12	12	12
36	15	18	21	27
100	25	22	24	45
196	35	22	44	80
324	45	25	47	108
400	50	24	54	138
676	65	35	75	224

(b) $\kappa_h \approx 0.3125$

Table B.4: NOI results for MP-1 using the AS_{8h} preconditioner with $\mathbb{P}_1\text{-}\mathrm{CS}_{CH}$ with varying CH. The wave number and the number of subdomains increase such that $\kappa_h \approx 0.625$ or $\kappa_h \approx 0.3125$ remain true. Additionally, $\delta = 0$.

Subdoms	k	$\mathrm{AS}_{4h}/\mathbb{P}_1\text{-CS}_{2h}$	$\mathrm{AS}_{4h}/\mathbb{P}_1\text{-CS}_{4h}$	$\mathrm{AS}_{4h}/\mathbb{P}_1\text{-CS}_{8h}$
4	5	11	12	8
36	15	18	21	19
100	25	35	36	36
196	35	106	93	90
324	45	214	139	141
400	50	x	199	178
676	65	x	x	x

(a) $\kappa_h \approx 0.625$

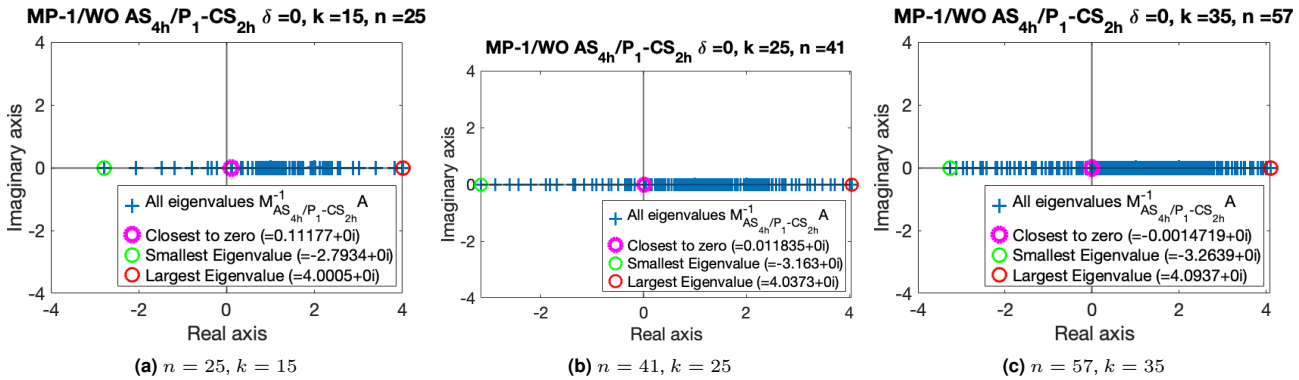
Subdoms	k	$\mathrm{AS}_{4h}/\mathbb{P}_1\text{-CS}_{2h}$	$\mathrm{AS}_{4h}/\mathbb{P}_1\text{-CS}_{4h}$	$\mathrm{AS}_{4h}/\mathbb{P}_1\text{-CS}_{8h}$
16	5	13	14	13
144	15	20	23	24
400	25	27	28	40
784	35	31	47	69
1296	45	35	47	95
1600	50	33	60	124
2704	65	52	72	200

(b) $\kappa_h \approx 0.3125$

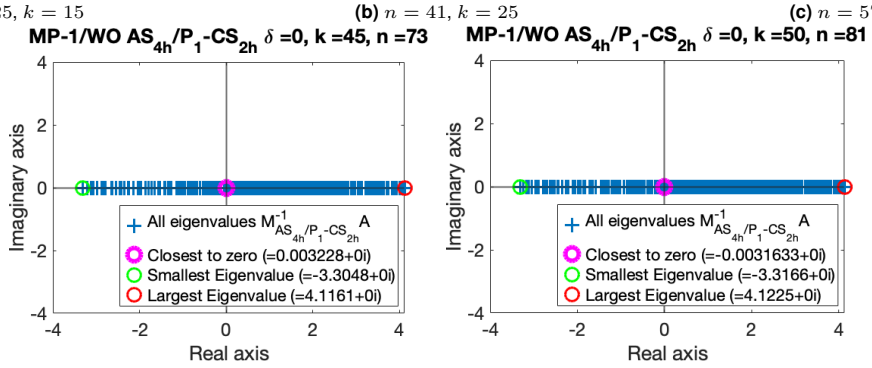
Table B.5: NOI results for MP-1 using the AS_{4h} preconditioner with $\mathbb{P}_1\text{-CS}_{CH}$ with varying CH. The wave number and the number of subdomains increase such that $\kappa_h \approx 0.625$ or $\kappa_h \approx 0.3125$ remain true. Additionally, $\delta = 1$.

Subdoms	36	64	100	144	196	256	324	400	484	576	676	784	900	1024	1156	1296	1440	1600
k = 1	14	14	14	14	14	14	13	13	13	13	13	13	13	13	13	13	13	13
k = 5	14	13	13	13	13	12	12	12	12	12	12	12	12	12	12	12	11	11
k = 10	15	12	13	13	13	13	12	12	12	12	12	12	12	12	11	11	11	11
k = 15	24	21	18	16	16	16	15	15	15	14	14	14	14	14	14	14	14	14
k = 20	29	29	27	20	18	19	16	16	16	16	15	15	15	15	15	14	14	14
k = 25	29	38	39	33	24	20	19	17	17	17	17	16	16	16	16	15	15	15
k = 30	43	38	81	46	48	31	22	20	18	16	17	17	17	17	16	16	16	16
k = 35	58	78	64	146	72	56	40	26	23	22	19	19	19	19	19	18	18	18
k = 40	82	60	88	117	177	97	71	65	34	29	29	28	27	28	25	22	22	21
k = 45	76	104	210	70	134	182	100	92	70	36	25	25	22	22	21	21	20	20
k = 50	72	210	145	106	92	212	187	119	104	87	49	33	28	25	25	24	24	22

Table B.6: Number of iterations with the $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-CS}_{2h}$ preconditioner for MP-1. Additionally, $\delta = 0$.

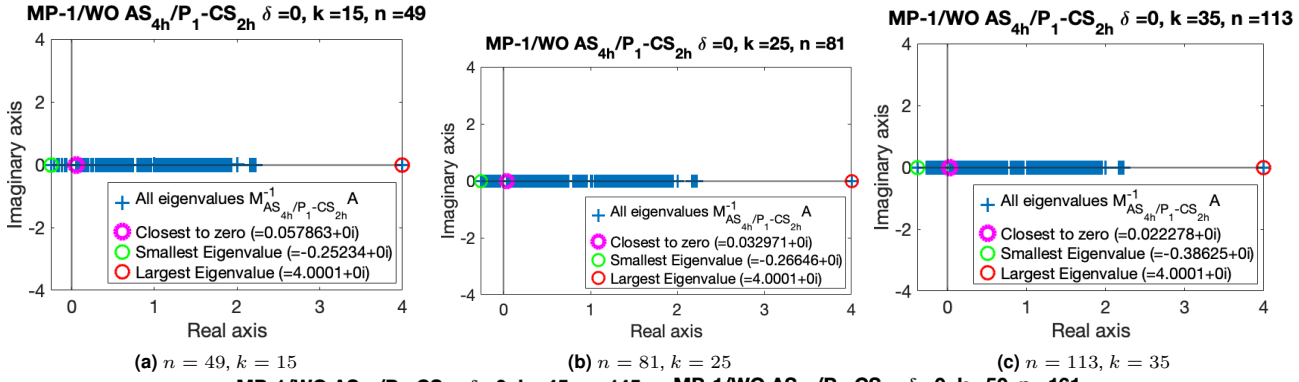


(a) $n = 25, k = 15$ **(b)** $n = 41, k = 25$ **(c)** $n = 57, k = 35$

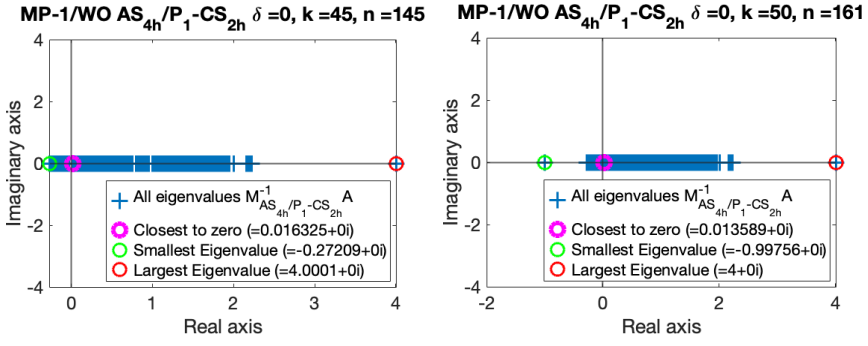


(d) $n = 73, k = 45$ **(e)** $n = 81, k = 50$

Figure B.2: eigenvalue plots for $k = 15, 25, 35, 45, 50$ and $\kappa_h \approx 0.625$ with the $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$ Helmholtz solver. Note that $\delta = 0$



(a) $n = 49, k = 15$ **(b)** $n = 81, k = 25$ **(c)** $n = 113, k = 35$



(d) $n = 145, k = 45$ **(e)** $n = 161, k = 50$

Figure B.3: Numerical eigenvalue plots for $k = 15, 25, 35, 45, 50$ and $\kappa_h \approx 0.3125$ with the $\mathrm{AS}_{4h}/\mathbb{P}_1\text{-}\mathrm{CS}_{2h}$ Helmholtz solver. Note that $\delta = 0$

C | More Numerical Experiment Results Using a Higher-Order Bézier Grid Coarse Problem

C.1 Two-level Additive Schwarz Solver

Subdoms	$\mathrm{AS}_{8h}/\mathrm{BCS}_{2h}$	$\mathrm{AS}_{8h}/\mathrm{BCS}_{4h}$	$\mathrm{AS}_{8h}/\mathrm{BCS}_{8h}$
16	13	14	15
64	13	14	16
144	12	14	15
256	12	14	15
400	12	14	15

Table C.1: NOI results for MP-3 using the $\mathrm{AS}_{8h}/\mathrm{BCS}_{CH}$ preconditioner with varying CH. Additionally, $\delta = 0$ and $\varepsilon = 0$.

Subdoms	$\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$	$\mathrm{AS}_{4h}/\mathrm{BCS}_{4h}$	$\mathrm{AS}_{4h}/\mathrm{BCS}_{8h}$
64	15	16	16
256	15	16	17
576	15	16	16
1024	15	16	16
1600	15	16	16

Table C.2: NOI results for MP-3 using the $\mathrm{AS}_{4h}/\mathrm{BCS}_{CH}$ preconditioner with varying CH. Additionally, $\delta = 1$.

Subdoms	k	$\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$	$\mathrm{AS}_{4h}/\mathrm{BCS}_{4h}$	$\mathrm{AS}_{4h}/\mathrm{BCS}_{8h}$
256	10	17	16	25
1024	20	17	17	29
2304	30	18	18	34
4096	40	18	19	40
6400	50	18	20	48
25600	100	18	24	x

(a) $H = 4h$**,** $\delta = 0$

Subdoms	k	$\mathrm{AS}_{8h}/\mathrm{BCS}_{2h}$	$\mathrm{AS}_{8h}/\mathrm{BCS}_{4h}$	$\mathrm{AS}_{8h}/\mathrm{BCS}_{8h}$
64	10	16	16	21
256	20	16	17	25
576	30	17	18	31
1024	40	17	19	37
1600	50	16	19	45
6400	100	17	24	98

(b) $H = 8h$**,** $\delta = 0$

Subdoms	k	$\mathrm{AS}_{16h}/\mathrm{BCS}_{2h}$	$\mathrm{AS}_{16h}/\mathrm{BCS}_{4h}$	$\mathrm{AS}_{16h}/\mathrm{BCS}_{8h}$
16	10	15	15	19
64	20	16	17	26
144	30	17	19	32
256	40	17	19	38
400	50	17	20	46
1600	100	17	24	99

(c) $H = 16h$**,** $\delta = 0$

Table C.3: NOI results for MP-2 using the $\mathrm{AS}_H/\mathrm{BCS}_{CH}$ **preconditioner with varying** CH **and** $H = 4h$**,** $H = 8h$ **or** $H = 16h$**. Additionally,** $\kappa_h \approx 0.15625$**.**

The following table gives an overview of how the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ solver for MP-2

Subdoms	36	64	100	144	196	256	324	400	484	576	676	784	900	1024	1156	1296	1440	1600
k = 1	15	16	16	16	16	16	16	16	16	17	17	17	17	17	17	18	18	18
k = 5	16	16	16	16	16	17	17	17	17	17	17	17	17	17	17	17	17	17
k = 10	17	17	17	17	17	17	17	17	17	17	18	18	18	18	18	18	18	18
k = 15	21	20	19	17	17	17	17	17	17	17	17	17	17	17	17	18	18	18
k = 20	22	25	26	20	19	18	17	17	17	17	17	17	17	17	17	17	17	17
k = 25	27	29	31	31	24	20	18	18	17	17	17	17	17	17	17	17	17	17
k = 30	32	30	55	36	36	28	22	19	18	18	18	18	18	18	18	18	18	18
k = 35	52	54	34	115	44	43	33	24	21	19	18	18	18	17	17	17	17	17
k = 40	53	45	42	41	98	54	50	44	27	22	20	19	18	18	17	17	17	17
k = 45	60	85	164	44	59	103	66	59	51	31	24	21	19	18	18	18	17	17
k = 50	56	134	70	65	46	88	109	76	66	63	36	26	22	20	19	18	18	18

Table C.4: Number of iterations with the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ **solver for MP-2 for small** k**. Additionally,** $\delta = 0$**.**

Subdoms	k	$\mathrm{AS}_{8h}/\mathrm{BCS}_{4h}$
64	10	14
144	15	17
256	20	17
400	25	17
1600	50	16
6400	100	12
14400	150	15
57600	300	14

(a) $H = 8h$

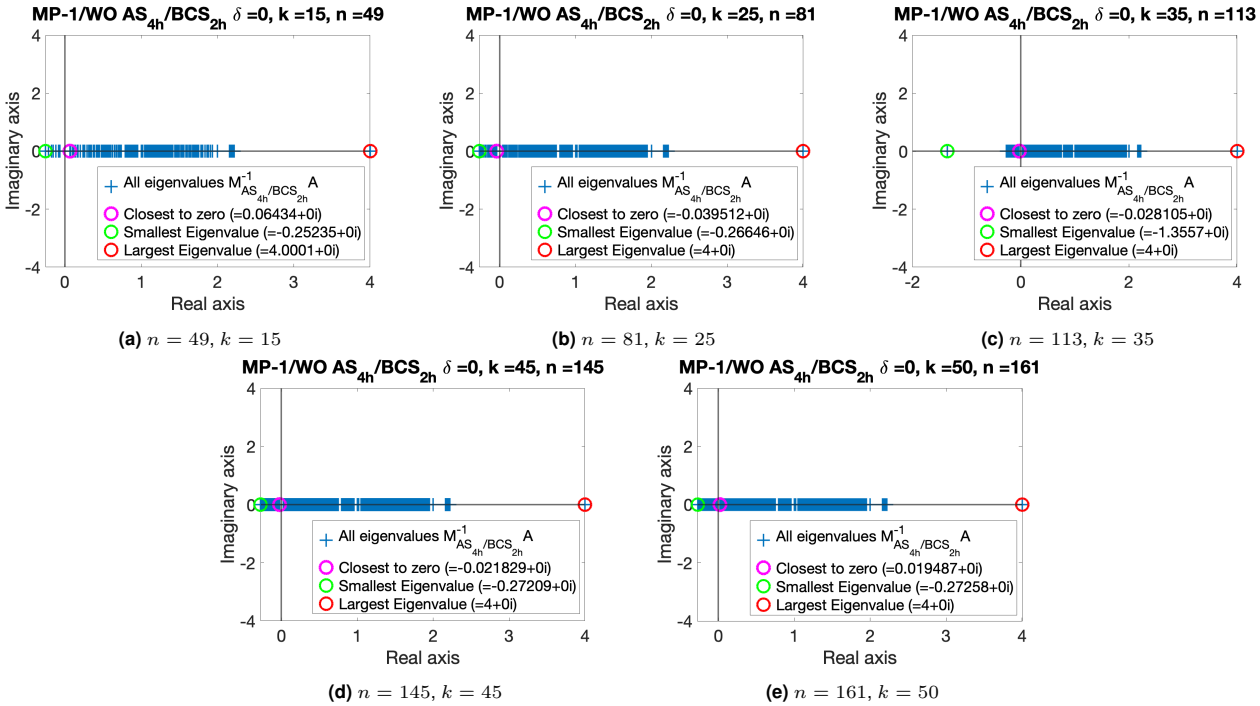
Subdoms	k	$\mathrm{AS}_{16h}/\mathrm{BCS}_{4h}$
16	10	13
36	15	17
64	20	20
100	25	22
400	50	25
1600	100	18
3600	150	22
14400	300	20

(b) $H = 16h$

Subdoms	k	$\mathrm{AS}_{32h}/\mathrm{BCS}_{4h}$
4	10	13
9	15	17
16	20	20
25	25	22
100	50	25
400	100	18
1600	150	22
3600	300	20

(c) $H = 32h$

Table C.5: NOI results for MP-1 using the $\mathrm{AS}_H/\mathrm{BCS}_{CH}$ preconditioner with $CH = 4h$ and $H = 8h$, $H = 16h$ or $H = 32h$. Additionally, $\kappa_h \approx 0.15625$ and $\delta = 0$.



(a) $n = 49, k = 15$ **(b)** $n = 81, k = 25$ **(c)** $n = 113, k = 35$

(d) $n = 145, k = 45$ **(e)** $n = 161, k = 50$

Figure C.1: Eigenvalue plots for $k = 15, 25, 35, 45, 50$ and $\kappa_h = 0.3125$ with the $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ Helmholtz solver. Note that $\delta = 0$

SubDoms (H)	$\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$
16 (80h)	36
25 (64h)	32
64 (40h)	x
100 (32h)	53
256 (20h)	x
400 (16h)	16
1600 (8h)	18
6400 (4h)	12

Table C.6: NOI for $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ for MP-1/WO problem with $n = 321$ and $k = 100$ ($\kappa_h = 0.3125$) when the number of subdomains is increased, meaning n^l is decreased. Additionally, $\delta = 0$ and x denotes that the maximum number of iterations is reached.

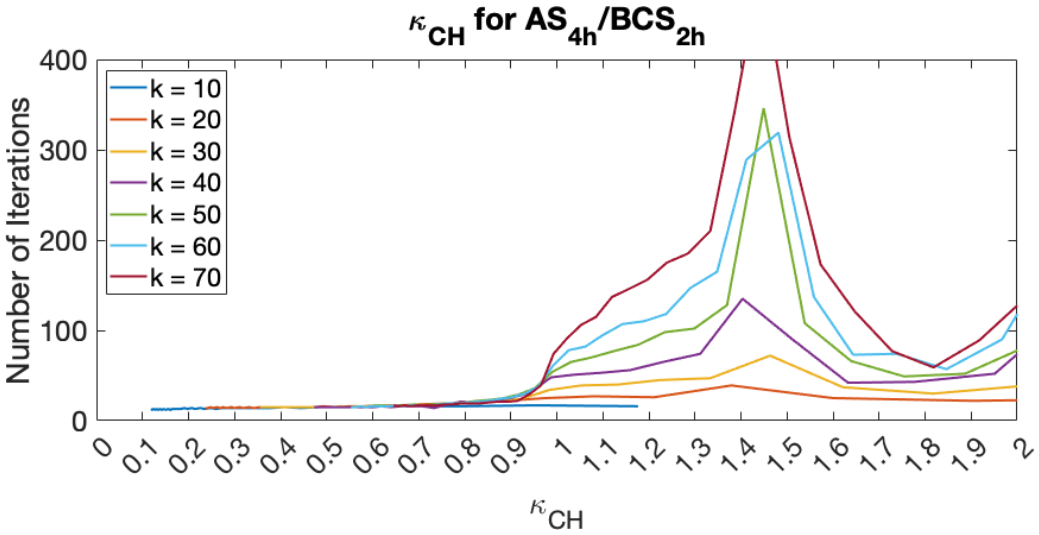


Figure C.2: $4h$ subdomain size $\mathrm{AS}_{4h}/\mathrm{BCS}_{2h}$ on MP-1

C.2 Two-level Scaled Additive Schwarz Solver

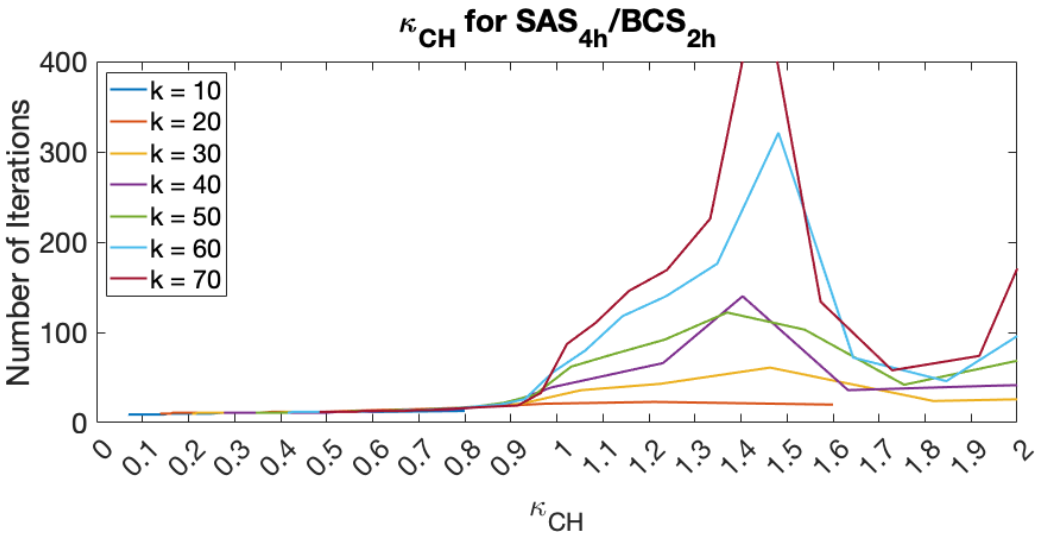


Figure C.3: $4h$ subdomain size $\mathrm{SAS}_{4h}/\mathrm{BCS}_{2h}$ on MP-1

C.3 Two-level Hybrid Schwarz Solver

Subdoms	k	$\mathrm{HS}_{4h}/\mathrm{BCS}_{2h}$
64	10	9
400	25	8
1600	50	8
6400	100	7
14400	150	8

(a) $H = 4h$

Subdoms	k	$\mathrm{HS}_{8h}/\mathrm{BCS}_{2h}$
16	10	9
100	25	10
400	50	10
1600	100	8
3600	150	9

(b) $H = 8h$

Subdoms	k	$\mathrm{HS}_{16h}/\mathrm{BCS}_{2h}$
4	10	9
25	25	11
100	50	10
400	100	10
900	150	8

(c) $H = 16h$

Table C.7: NOI results for MP-1 using the $\mathrm{HS}_H/\mathrm{BCS}_{2h}$ preconditioner with $H = 4h$, $H = 8h$ or $H = 16h$. Additionally, $\kappa_h \approx 0.3125$ and $\delta = 1$.

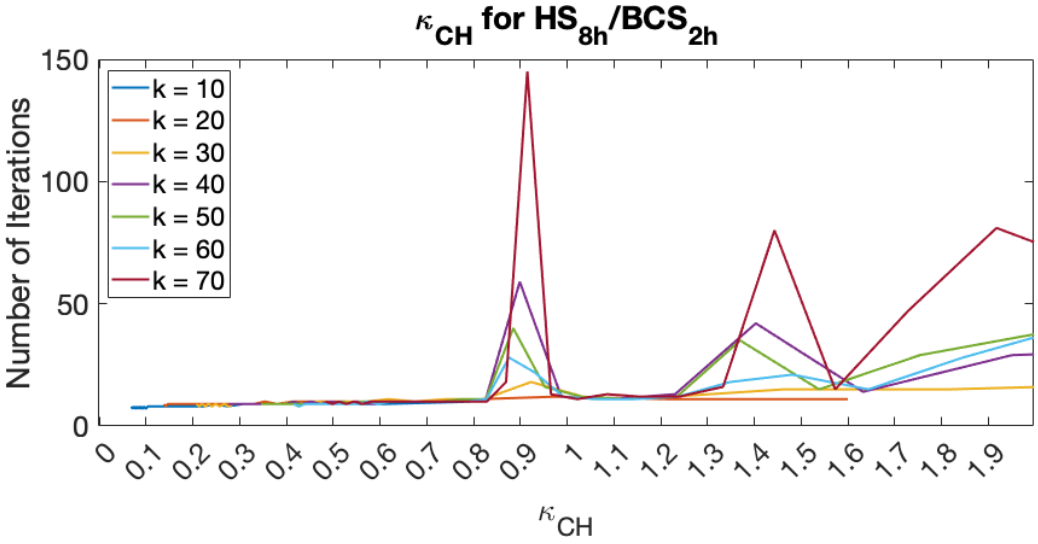


Figure C.4: $\mathrm{HS}_{8h}/\mathrm{BCS}_{2h}$ on MP-1

C.4 Two-level Scaled Hybrid Schwarz Solver

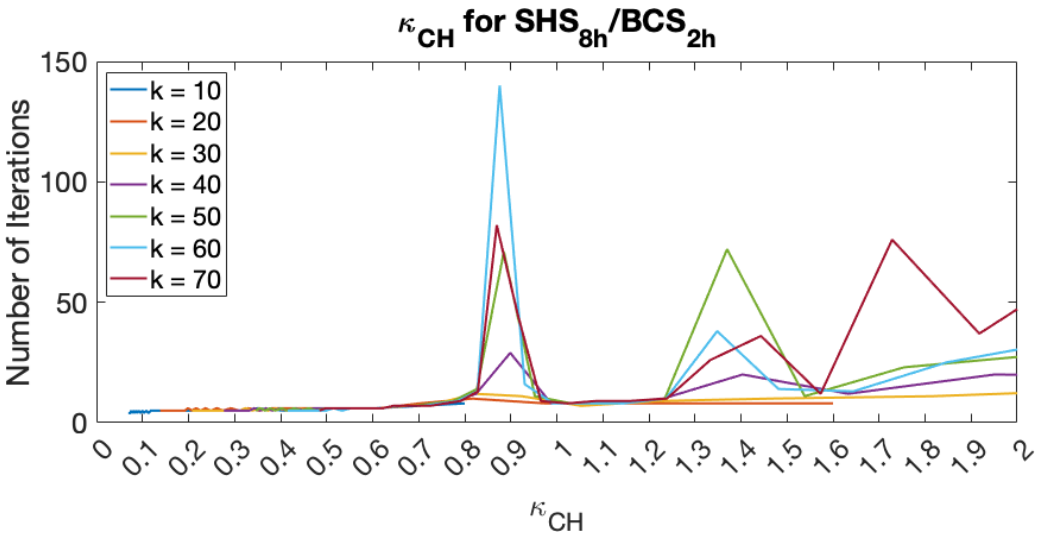


Figure C.5: $8h$ subdomain size $\mathrm{SHS}_{8h}/\mathrm{BCS}_{2h}$ on MP-1

Bibliography

[1] Hermann Amandus Schwarz. *Gesammelte mathematische abhandlungen*. Vol. 260. American Mathematical Soc., 1972. ISBN: 0828402604.

[2] Aleksandr Mikhailovich Matsokin and Sergei Vladimirovich Nepomnyaschikh. “The Schwarz alternation method in a subspace”. In: *Izvestiya Vysshikh Uchebnykh Zavedenii. Matematika* 10 (1985), pp. 61–66. ISSN: 0021-3446.

[3] Sergey V Nepomnyaschikh. “Domain decomposition and Schwarz methods in a subspace for the approximate solution of elliptic boundary value problems”. Thesis. 1986.

[4] Roy A Nicolaides. “Deflation of conjugate gradients with applications to boundary value problems”. In: *SIAM Journal on Numerical Analysis* 24.2 (1987), pp. 355–365.

[5] Olof Widlund and Maksymilian Dryja. “An additive variant of the Schwarz alternating method for the case of many subregions”. In: (1987).

[6] Maksymilian Dryja. “An additive Schwarz algorithm for two-and three-dimensional finite element elliptic problems”. In: *Domain decomposition methods* (1989), pp. 168–172.

[7] Pierre-Louis Lions. “On the Schwarz alternating method. II”. In: *Domain decomposition methods* 628 (1989), pp. 47–70.

[8] Maksymilian Dryja and Olof B. Widlund. “Chapter 16 - Some Domain Decomposition Algorithms for Elliptic Problems”. In: *Iterative Methods for Large Linear Systems*. Ed. by David R. Kincaid and Linda J. Hayes. Academic Press, 1990, pp. 273–291.

[9] H. A. van der Vorst. “Bi-CGSTAB: A Fast and Smoothly Converging Variant of Bi-CG for the Solution of Nonsymmetric Linear Systems”. In: *SIAM Journal on Scientific and Statistical Computing* 13.2 (1992), pp. 631–644.

[10] Maksymilian Dryja and Olof B Widlund. “Domain decomposition algorithms with small overlap”. In: *SIAM Journal on Scientific Computing* 15.3 (1994), pp. 604–620. ISSN: 1064-8275.

[11] William D. Gropp Barry F. Smith Petter E. Bjorstad. *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations*. Cambridge: Cambridge University Press, 1996. ISBN: 0521602866.

[12] Xiao-Chuan Cai and Marcus Sarkis. “A Restricted Additive Schwarz Preconditioner for General Sparse Linear Systems; CU-CS-843-97”. In: (1997).

[13] F. Ihlenburg and I. Babuska. “Finite element solution of the Helmholtz equation with high wave number .2. The h-p version of the FEM”. In: *SIAM JOURNAL ON NUMERICAL ANALYSIS* 34.1 (1997), pp. 315–358.

[14] David E Keyes. “Parallel numerical algorithms: An introduction”. In: *Parallel Numerical Algorithms*. Springer, 1997, pp. 1–15.

[15] Barry F Smith. “Domain decomposition methods for partial differential equations”. In: *Parallel Numerical Algorithms*. Springer, 1997, pp. 225–243.

[16] A. Deraemaeker, I. Babuska, and P. Bouillard. “Dispersion and pollution of the FEM solution for the Helmholtz equation in one, two and three dimensions”. In: *INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING* 46.4 (1999), pp. 471–499.

[17] K. Gerdes and F. Ihlenburg. “On the pollution effect in FE solutions of the 3D-Helmholtz equation”. In: *COMPUTER METHODS IN APPLIED MECHANICS AND ENGINEERING* 170.1-2 (1999), pp. 155–172.

[18] Alfio Maria Quarteroni and Alberto Valli. *Domain decomposition methods for partial differential equations*. Oxford University Press, 1999. ISBN: 0198501781.

[19] Yousef Saad. *Iterative methods for sparse linear systems*. SIAM, 2003. ISBN: 0898715342.

[20] Yogi A Erlangga, Cornelis Vuik, and Cornelis Willebrordus Oosterlee. “On a class of preconditioners for solving the Helmholtz equation”. In: *Applied Numerical Mathematics* 50.3-4 (2004), pp. 409–425. ISSN: 0168-9274.

[21] Andrea Toselli and Olof Widlund. *Domain decomposition methods-algorithms and theory*. Vol. 34. Springer Science & Business Media, 2004. ISBN: 3540206965.

[22] Yogi A Erlangga, Cornelis W Oosterlee, and Cornelis Vuik. “A novel multigrid based preconditioner for heterogeneous Helmholtz problems”. In: *SIAM Journal on Scientific Computing* 27.4 (2006), pp. 1471–1492. ISSN: 1064-8275.

[23] Yogi A Erlangga, Cornelis Vuik, and Cornelis W Oosterlee. “Comparison of multigrid and incomplete LU shifted-Laplace preconditioners for the inhomogeneous Helmholtz equation”. In: *Applied numerical mathematics* 56.5 (2006), pp. 648–666. ISSN: 0168-9274.

[24] Martin B van Gijzen, Yogi A Erlangga, and Cornelis Vuik. “Spectral analysis of the discrete Helmholtz operator preconditioned with a shifted Laplacian”. In: *SIAM Journal on Scientific Computing* 29.5 (2007), pp. 1942–1958. ISSN: 1064-8275.

[25] Clark R Dohrmann, Axel Klawonn, and Olof B Widlund. “A family of energy minimizing coarse spaces for overlapping Schwarz preconditioners”. In: *Domain decomposition methods in science and engineering XVII*. Springer, 2008, pp. 247–254.

[26] Clark R Dohrmann, Axel Klawonn, and Olof B Widlund. “Domain decomposition for less regular subdomains: Overlapping Schwarz in two dimensions”. In: *SIAM journal on numerical analysis* 46.4 (2008), pp. 2153–2168. ISSN: 0036-1429.

[27] Victor Eijkhout. *Introduction to high performance scientific computing*. Lulu. com, 2010. ISBN: 1257992546.

[28] O. G. Ernst and M. J. Gander. “Why it is difficult to solve Helmholtz problems with classical iterative methods”. In: ed. by T. Y. Hou et al. Vol. 83. Springer Verlag, 2012, pp. 325–363.

[29] Lea Conen et al. “A coarse space for heterogeneous Helmholtz problems based on the Dirichlet-to-Neumann operator”. In: *Journal of Computational and Applied Mathematics* 271 (2014), pp. 83–99. ISSN: 0377-0427.

[30] Martin J Gander and Gerhard Wanner. “The origins of the alternating Schwarz method”. In: *Domain decomposition methods in science and engineering XXI*. Springer, 2014, pp. 487–495.

[31] André Gaul. “Recycling Krylov subspace methods for sequences of linear systems”. In: (2014).

[32] William Gropp, Ewing Lusk, and Anthony Skjellum. *Using MPI: portable parallel programming with the Message-Passing-Interface*. Third edition. Cambridge, MA: The MIT Press, 2014.

[33] A.H. Sheikh. “Development Of The Helmholtz Solver Based On A Shifted Laplace Preconditioner And A Multigrid Deflation Technique”. Thesis. 2014.

[34] C. Vuik and D. J. P. Lahaye. *Scientific computing : WI4201*. Delft: TU Delft, 2014.

[35] Victorita Dolean, Pierre Jolivet, and Frédéric Nataf. *An introduction to domain decomposition methods: algorithms, theory, and parallel implementation*. SIAM, 2015. ISBN: 1611974054.

[36] Martin J Gander, Ivan G Graham, and Euan A Spence. “Applying GMRES to the Helmholtz equation with shifted Laplacian preconditioning: what is the largest shift for which wavenumber-independent convergence is guaranteed?” In: *Numerische Mathematik* 131.3 (2015), pp. 567–614. ISSN: 0945-3245.

[37] Alexander Heinlein, Axel Klawonn, and Oliver Rheinbach. “A Parallel Implementation of a Two-Level Overlapping Schwarz Method with Energy-Minimizing Coarse Space Based on Trilinos”. In: *SIAM Journal on Scientific Computing* 38.6 (2016), pp. C713–C747. ISSN: 1064-8275. DOI: 10.1137/16m1062843. URL: %3CGo%20to%20ISI%3E://WOS:000391853100005.

[38] Pierre-Henri Cocquet and Martin J Gander. “How large a shift is needed in the shifted Helmholtz preconditioner for its effective inversion by multigrid?” In: *SIAM Journal on Scientific Computing* 39.2 (2017), A438–A478. ISSN: 1064-8275.

[39] Clark R Dohrmann and Olof B Widlund. “On the design of small coarse spaces for domain decomposition algorithms”. In: *SIAM Journal on Scientific Computing* 39.4 (2017), A1466–A1488. ISSN: 1064-8275.

[40] M. Bonazzoli et al. “Two-level Preconditioners for the Helmholtz Equation”. In: *Lecture Notes in Computational Science and Engineering*. Vol. 125. Springer Verlag, 2018, pp. 139–147.

[41] Y. Wang, P. F. Van de Moortele, and B. He. “Automated gradient-based electrical properties tomography in the human brain using 7 Tesla MRI”. In: *Magnetic Resonance Imaging* 63 (2019), pp. 258–266. DOI: 10.1016/j.mri.2019.08.003. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85071396351&doi=10.1016%2fj.mri.2019.08.003&partnerID=40&md5=3ffdca382f572ba991a00e4dbc48e99a.

[42] Vandana Dwarka and Cornelis Vuik. “Scalable convergence using two-level deflation preconditioning for the helmholtz equation”. In: *SIAM Journal on Scientific Computing* 42.2 (2020), A901–A928. ISSN: 1064-8275.

[43] Vandana Dwarka and Cornelis Vuik. “Scalable multi-level deflation preconditioning for the highly indefinite Helmholtz equation”. In: (2020). ISSN: 1389-6520.

[44] N. Bootland et al. “A comparison of coarse spaces for Helmholtz problems in the high frequency regime”. In: *Computers and Mathematics with Applications* 98 (2021), pp. 239–253. ISSN: 08981221 (ISSN).

[45] Niall Bootland et al. “Inexact subdomain solves using deflated GMRES for Helmholtz problems”. In: *arXiv preprint arXiv:2103.17081* (2021).

[46] Vandana Dwarka et al. “Towards accuracy and scalability: Combining Isogeometric Analysis with deflation to obtain scalable convergence for the Helmholtz equation”. In: *Computer Methods in Applied Mechanics and Engineering* 377 (2021), p. 113694. ISSN: 0045-7825.

[47] X. Huang and S. Greenhalgh. “A finite-difference iterative solver of the Helmholtz equation for frequency-domain seismic wave modeling and full-waveform inversion”. In: *Geophysics* 86.2 (2021), T107–T116. DOI: 10.1190/geo2020-0411.1. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85105052013&doi=10.1190%2fgeo2020-0411.1&partnerID=40&md5=f419ef0a08879611c247f9c9a8945a23.

[48] John Jumper et al. “Highly accurate protein structure prediction with AlphaFold”. In: *Nature* 596.7873 (2021), pp. 583–589. ISSN: 1476-4687. DOI: 10.1038/s41586-021-03819-2. URL: https://doi.org/10.1038/s41586-021-03819-2.

[49] Daniel Appelö et al. “El-WaveHoltz: A time-domain iterative solver for time-harmonic elastic waves”. In: *Computer Methods in Applied Mechanics and Engineering* 401 (2022), p. 115603. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2022.115603. URL: https://www.sciencedirect.com/science/article/pii/S0045782522005655.

[50] Maksymilian Dryja and Olof B Widlund. “Towards a unified theory of domain decomposition algorithms for elliptic problems”. In: *Third international symposium on domain decomposition methods for partial differential equations*, pp. 3–21.

[51] A. Heinlein et al. “FROSch: A Fast And Robust Overlapping Schwarz Domain Decomposition Preconditioner Based on Xpetra in Trilinos”. In: *Lecture Notes in Computational Science and Engineering*. Ed. by R. Haynes et al. Vol. 138. Springer Science and Business Media Deutschland GmbH, pp. 176–184.

[52] Pierre-Louis Lions. “On the Schwarz alternating method. I”. In: *First international symposium on domain decomposition methods for partial differential equations*. Vol. 1. Paris, France, p. 42.

[53] Pierre-Louis Lions. “On the Schwarz alternating method. III: a variant for nonoverlapping subdomains”. In: *Third international symposium on domain decomposition methods for partial differential equations*. Vol. 6. SIAM Philadelphia, pp. 202–223.