Delft University of Technology
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft Institute of Applied Mathematics

Fourier Analysis of Iterative Methods for the Helmholtz problem

A thesis submitted to
Delft Institute of Applied Mathematics
in partial fulfillment of the requirements

for the degree

MASTER OF SCIENCE
in
APPLIED MATHEMATICS

by

HANZHI DIAO

Delft, the Netherlands
December, 2012

TU Delft

MSc THESIS APPLIED MATHEMATICS

"Fourier Analysis of Iterative Methods for the Helmholtz problem"

Hanzhi Diao

Delft University of Technology

Daily Supervisor

Prof. dr. ir. C. Vuik

Responsible Professor

Prof. dr. ir. A. W. Heemink

Other thesis committee member

Dr. ir. M. B. van Gijzen

December, 2012

Delft, the Netherlands

Contents

Acknowledgement

This thesis is the project that finalizes my study on the joint master program COSSE [20].

First and foremost, I would like to express the utmost gratitude to my daily supervisor Prof. dr. ir. Kees Vuik. Without his solid support I would never finish the thesis work. I deeply appreciate the opportunity that he offered me, the patience that he shared with me and the trust that he put in me. It is a great pleasure to have the weekly discussion with him that advanced my thesis work towards the final completion.

Besides that I thank Prof. dr. ir. Arnold W. Heemink for being my responsible professor and Dr. ir. Martin van Gijzen for being the member of the committee board.

Finally, my heartfelt thanks go to Associate Professor Dr. Michael Hanke in KTH Sweden. As the coordinator of COSSE program, he keeps giving me firm support and valuable suggestion both of which help me to complete the program.

Hanzhi Diao
December, 2012
Delft, the Netherlands

Abstract

Helmholtz problem has a various application in real world. The discretization of Helmholtz equation results in a sparse linear system that is hard to solve. The main difficulty lies in the high indefiniteness of the matrix. In order to improve the spectral properties of the original matrix, the shifted Laplacian preconditioner and deflation operator are used to precondition the system so that the Krylov convergence accelerates.

This thesis is devoted to the analysis on the spectrum distribution of the preconditioned systems together with the resulting convergence behaviour. Fourier analysis is the tool that finds out the eigenvalues of the matrices and therefore the spectrum distribution. The preconditioning effect is investigated with respect to various wavenumbers and different wave resolution. The study is also done on the action of the shift in the shifted Laplacian preconditioner. Based on the multigrid method, the investigation is extended to the approximated preconditioning where the shifted Laplacian preconditioner is inverted not exactly but by several multigrid iterations. Besides that the convergence behaviour of the multigrid method is studied and some observation is obtained. The calculation is done for the optimal shift which is expected to result in the fastest Krylov convergence of the preconditioned system.

The theoretical analysis is substantiated by the numerical experiments. The numerical observation matches the conclusion by Fourier analysis. In addition to that the numerical experiment reveals the influence of orthogonalization method on the Krylov convergence. The experiment on the multilevel Krylov multigrid method shows how the internal iteration on different levels affects the external Krylov convergence. Finally, the optimal shift for the Krylov convergence is obtained by the numerical experiment.

Key Words Helmholtz problem, Krylov subspace methods, multigrid method, multilevel Krylov multigrid method, shifted Laplacian preconditioner, deflation operator, Fourier analysis

Chapter 1

Introduction

The Helmholtz equation, named for the German physicist Hermann von Helmholtz, is a partial differential equation that governs the scattering of plane wave in acoustics and electromagnetism [17].

$$-\Delta u(\mathbf{x}) - k^2 u(\mathbf{x}) = f(\mathbf{x}), \quad \text{in } \Omega \in \mathbb{R}^3$$

In many applications the problem is modelled by a high wavenumber propagation. For the sake of accuracy, the discretization of the Helmholtz equation generates a very large-scale coefficient matrix, especially in case of the 3D modelling. Due to the sparsity of the large matrix, iterative methods [26] are usually employed as the solver. And these solvers require more attention to the choice of preconditioner because the matrix becomes highly indefinite as the wavenumber increases. In the early 1980's Goldstein & Turkel [2] started the work on the iterative methods for Helmholtz problem. From then on, various methods have been developed and applied to Helmholtz problem, which makes it nowadays still an active research topic. A survey is done in [7] for the recent advances in solving Helmholtz problem.

1.1 Helmholtz problem

1.1.1 Derivation of Helmholtz equation

Helmholtz equation is the time-independent form of time harmonic wave propagation. It often appears in the study of physical problem that involves PDE in both time and space. The technique of separation of variables reduces the complicated analysis into the simpler form concerning only spatial derivatives.

The derivation starts from the common wave equations

$$\left(\nabla^2 - \frac{1}{c^2}\frac{\partial^2}{\partial t^2}\right) u(\mathbf{x}, t) = 0, \tag{1.1}$$

where c is the wave speed and $\mathbf{x}$ represents the space position in $\mathbb{R}^3$. Applying separation of variables, the harmonic wave function $u(\mathbf{x}, t)$ can be decomposed into

$$u(\mathbf{x}, t) = \phi(\mathbf{x})T(t). \tag{1.2}$$

Substitution (1.2) into (1.1) results in

$$\frac{\nabla^2\phi}{\phi} = \frac{1}{c^2 T}\frac{\partial^2 T}{\partial t^2}, \tag{1.3}$$

where the left-hand expression is only dependent of $\mathbf{x}$ and the right-hand side depends only on t. In order to have the equality (1.3) valid, both left-hand and right-hand sides are supposed

to equal an identical constant value. This observation brings out two equations for $\phi(\mathbf{x})$ and $T(t)$ respectively.

$$\frac{\nabla^2\phi}{\phi} = -k^2 \quad \text{and} \quad \frac{1}{c^2T}\frac{\partial^2 T}{\partial t^2} = -k^2. \tag{1.4}$$

Here the expression $-k^2$ is chosen as the constant value. Physically, k stands for the wavenumber. Thus, the Helmholtz equation can be obtained by rearranging the first equation in (1.4).

$$-\nabla^2\phi - k^2\phi = -(\nabla^2 + k^2)\phi = 0 \tag{1.5}$$

In the above derivation ϕ is an abstract quantity which can be more meaningful in a specific problem, i.e. pressure, amplitude, velocity et al.

1.1.2 Boundary Conditions

As an elliptic equation, the Helmholtz equation requires a proper boundary condition in order to construct a well-posed physical problem. Particular physical laws are satisfied on the boundary of the domain where the solution is computed. The domain is either finite or infinite, which refers to the interior problem or exterior problem respectively. When the solution is computed numerically, an infinite domain needs a truncation.

Generally there are Dirichlet condition, Neumann condition and Sommerfeld conditon [6]. Here, the Sommerfeld condition in the form of first order is given by.

$$\frac{\partial\phi}{\partial\mathbf{n}} - \iota k\phi = 0, \quad \text{on } \Gamma = \partial\Omega, \tag{1.6}$$

where $\mathbf{n}$ is the outward direction normal to the boundary and $\iota = \sqrt{-1}$ is the imaginary number[1].

Using the Sommerfeld condition, the Helmholtz problem for wave propagation can be defined in the following way.

Definition 1.1. *Find the field ϕ such that*

$$\begin{aligned} -\nabla^2\phi - k^2\phi &= f, \quad \textit{in } \Omega \in \mathbb{R}^3 \\ \frac{\partial\phi}{\partial\mathbf{n}} - \iota k\phi &= 0, \quad \textit{on } \Gamma = \partial\Omega \end{aligned}$$

The function f is the source term or the driving force.

1.1.3 Dimensionless Helmholtz problem

For the sake of generalization, the Helmholtz problem is now scaled to a dimensionless problem on a unit domain $\tilde{\Omega} = [0,1]^3$. The wavenumber will therefore be adapted properly in order to make the dimensionless problem spectrally equivalent to the original one.

The Helmholtz equation can be written in the following scalar form

$$-\frac{\partial^2\phi}{\partial x^2} - \frac{\partial^2\phi}{\partial y^2} - \frac{\partial^2\phi}{\partial z^2} - k^2\phi = f(x,y,z) \quad \text{on} \quad \Omega = [0,L]^3.$$

The new scaled variables are introduced as

$$\tilde{x} = \frac{x}{L},\ \tilde{y} = \frac{y}{L},\ \tilde{z} = \frac{z}{L},$$

which leads to the derivative relation

$$\frac{d\tilde{x}}{dx} = \frac{d\tilde{y}}{dy} = \frac{d\tilde{z}}{dz} = \frac{1}{L}.$$

[1] Throughout the thesis, the Greek letter ι is employed as the unit for imaginary number.

Then, the original Helmholtz equation is reformulated into

$$-\frac{\partial^2\phi}{\partial\tilde{x}^2}-\frac{\partial^2\phi}{\partial\tilde{y}^2}-\frac{\partial^2\phi}{\partial\tilde{z}^2}-\tilde{k}^2\phi=\tilde{f}(\tilde{x},\tilde{y},\tilde{z}), \tag{1.7}$$

where $(\tilde{x},\tilde{y},\tilde{z})\in\tilde{\Omega}=[0,1]^3$ and $\tilde{k}=L\cdot k, \tilde{f}=L^2\cdot f$. Although the domain is transferred from $[0,L]^3$ to $[0,1]^3$, the solution is still the same due to the new $\tilde{k}$ and $\tilde{f}$. And the scaled wavenumber also guarantees the spectral equivalence between two problems in different domains. For the sake of brevity, the tilde notation will be left out in the following section.

1.1.4 Discretization of Helmholtz equation

Both finite difference method and finite elements method are suitable for the discretization of the Helmholtz problem. In this thesis the finite difference method is used while the application of finite element method to Helmholtz problem can be found in [18].

Here, a second order accurate finite difference method is applied to the dimensionless Helmholtz equation (1.7). A high order finite difference application can be found in [29].

The 3D domain of interest $\Omega=[0,1]^3$ is discretized on an equidistant grid which has L subintervals in x-direction, M subintervals in y-direction and N subintervals in z-direction

$$\begin{aligned} x_l &= l\ \Delta x, \quad l=0,1,\cdots,L\\ y_m &= m\Delta y, \quad m=0,1,\cdots,M\\ z_n &= n\ \Delta z, \quad n=0,1,\cdots,N. \end{aligned}$$

A standard central difference scheme is applied to the spatial derivatives and results in a second order accuracy.

$$\left.\frac{\partial^2\phi}{\partial x^2}\right|_l=-\frac{1}{\Delta x^2}(\phi_{l-1}-2\phi_l+\phi_{l+1})-\mathcal{O}(\Delta x^2)$$

It is similar in y-direction and z-direction. The complete set of discretization becomes

$$-\frac{1}{\Delta x^2}(\phi_{l-1}-2\phi_l+\phi_{l+1})-\frac{1}{\Delta y^2}(\phi_{m-1}-2\phi_m+\phi_{m+1})+ \\ -\frac{1}{\Delta z^2}(\phi_{n-1}-2\phi_n+\phi_{n+1})-k^2\phi_{l,m,n}=f_{l,m,n} \tag{1.8}$$

A first order forward/backward scheme is applied to (1.6) on left/right conditions. The result is a set of extra terms added to equation (1.8).

$$\frac{p_{1,m,n}-p_{0,m,n}}{\Delta x}-\iota k p_{0,m,n}=0 \quad \text{and} \quad \frac{p_{L,m,n}-p_{L-1,m,n}}{\Delta x}-\iota k p_{L,m,n}=0 \tag{1.9}$$

The discretized equations in y and z directions have the similar form to (1.9) and all the unknowns on the boundaries can now be written as the expression of their neighboring internal grid points.

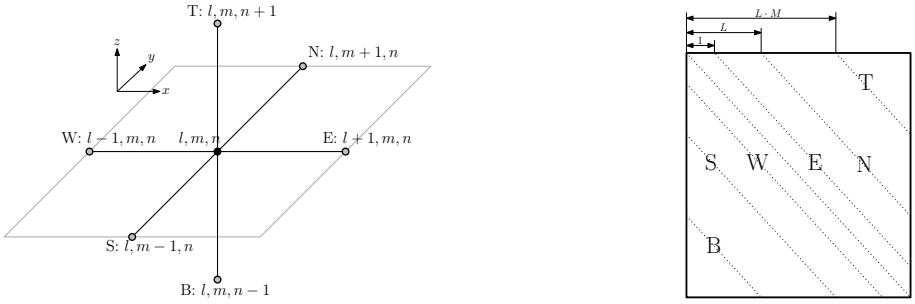
$$\begin{aligned} p_{0,m,n}&=\frac{p_{1,m,n}}{1+\iota k\Delta x} \quad \text{and} \quad p_{L,m,n}=\frac{p_{L-1,m,n}}{1+\iota k\Delta x}\\ p_{l,0,n}&=\frac{p_{l,1,n}}{1+\iota k\Delta y} \quad \text{and} \quad p_{l,M,n}=\frac{p_{l,M-1,n}}{1+\iota k\Delta y}\\ p_{l,m,0}&=\frac{p_{l,m,1}}{1+\iota k\Delta z} \quad \text{and} \quad p_{l,m,N}=\frac{p_{l,m,N-1}}{1+\iota k\Delta z} \end{aligned} \tag{1.10}$$

Inserting equation (1.10) into equation (1.8) for all l,m,n in the domain yields a set of linear equations

$$A\mathbf{x}=b \quad \text{for } A\in\mathbb{C}^{n\times n}.$$

The vector $\mathbf{x}$ contains all unknowns in the domain and its total amount is $n=(L-1)\cdot(M-1)\cdot(N-1)$.

Linear System The resulting linear system is a symmetric complex sparse matrix. The seven-point stencil discretization (see Figure 1.1a) generates a highly sparse matrix which only has seven non-zero diagonals (see Figure 1.1b). Although the coefficient matrix for the Helmholtz equation is real, the complex number is introduced to the linear system by the Sommerfeld condition. For a large k, the matrix becomes very indefinite which has a bad impact on the convergence of iteration.



(a) seven-point stencil (b) sparsity pattern

Figure 1.1: The illustration of discretization stencil and the linear system

1.1.5 Model Problem

Throughout this thesis, the model problem is based on the one-dimensional Helmholtz problem with homogeneous Dirichlet boundary condition. The resulting linear system is given by

$$Ax = b \quad \text{where } A = \frac{1}{h^2}\begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix} - k^2 I. \tag{1.11}$$

Here, I denotes the identity matrix. The matrix size is $n \times n$ and the grid size is given by $h = \frac{1}{n+1}$.

Wave Resolution In the numerical solution of wave problems, the wave resolution determines the accuracy of the solution [3]. It refers to the amount of subintervals per wavelength, which is denoted by gw. So the wave resolution gw will be an important parameter that represents the discretization accuracy of the Helmholtz problem in this thesis. The relation among grid size, wavenumber and wave resolution is given by

$$\frac{2\pi}{k} = gw \cdot h.$$

Eigenvalues The matrix A in (1.11) is a linear combination of the discrete Laplacian operator. Its eigenvalues are given by

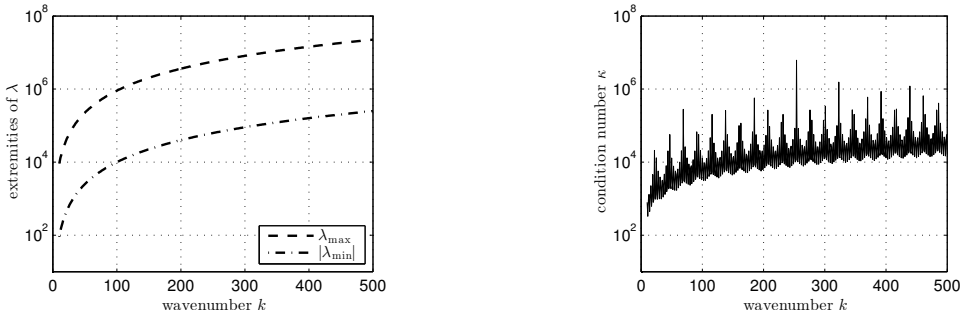
$$\lambda_l = \frac{4}{h^2}\sin^2(l\pi h/2) - k^2 \quad \text{for } l = 1, 2, \cdots, n.$$

The corresponding eigenvectors are in the form of

$$v_l = [\sin(l\pi h), \sin(2 \cdot l\pi h), \cdots, \sin(n \cdot l\pi h)]^T \quad \text{for } l = 1, 2, \cdots, n.$$

1.2 Difficulties in Solving Helmholtz problem

The main difficulty in solving Helmholtz problem by Krylov subspace methods lies in the high indefiniteness of the linear system. Consider the model problem on page 4. The eigenvalues are spread over a large range which varies from the negative to the positive. Such an unfavourable spectrum distribution leads to a very slow Krylov convergence. The indefiniteness is demonstrated by the comparison between the largest eigenvalue $\lambda_{\max}$ and the smallest eigenvalue $\lambda_{\min}$ as in Figure 1.2a. Due to the indefiniteness, the smallest eigenvalue is negative so it is shown in the absolute value.



(a) largest and smallest eigenvalues (b) condition number

Figure 1.2: The spectral properties of the model problem on page 4

Besides the indefiniteness, the condition number $\kappa(A)$ is also very large as plotted in Figure 1.2b. The smallest eigenvalue in magnitude is located somewhere near zero, the distance of which is neither regular nor monotonic as the wavenumber increases. This explains the oscillation in the Figure 1.2b.

Multigrid Method It might be reasonable to consider solving Helmholtz problem by multigrid method. However, the application of multigrid method would definitely fail. The first reason is that the smoothing iteration is not convergent since some of the eigenvalues of the ω-Jacobi operator are always larger than one. Secondly, the coarse grid is incapable of tackling the high wavenumber problems. So the information will be lost or distorted during the intergrid operation. See [12] for more details.

1.3 Outline of the Thesis

The Helmholtz problem is solved by the Krylov subspace method. This thesis attempts to build a bridge between the convergence behaviour and the spectral properties. The main task focuses on the application of Fourier analysis to investigating spectral properties.

All the analysis and computation are applied to the simple model problem on page 4. There are two reasons.

- There is no essential difference in principle between one-dimensional and higher dimensional problems. The conclusion on one-dimensional problem provides a good reference and can be extended to higher dimensional problems.
- The Dirichlet boundary condition is mathematically simple but results in the most unfavourable spectral properties. So the research is focused on the problem with the poorest convergence.

The formulation of the Helmholtz problem has already been discussed in the previous two sections. Chapter 2 discusses the iterative methods that are used in the solution of Helmholtz

problem. The preconditioning techniques for the Helmholtz problem are introduced in Chapter 3. In Chapter 4 the spectral properties of the matrix A are investigated in detail by using Fourier analysis. The numerical experiments are presented in Chapter 5, which corresponds to the analytical result in the previous chapter. The thesis is finalized by Chapter 6.

Chapter 2

Iterative Methods

The Helmholtz problem is discretized with either finite difference method or finite element method, which results in the linear system

$$Ax = b.$$

The linear system can be solved by a direct method if its size is small. However, the computational cost will increase superlinearly in accordance to the problem size. Furthermore, the numerical solution by a direct method will be plagued by the rounding error as the matrix size goes up. So the iterative methods are employed to tackle the large scale problems.

The computational procedure of an iterative method starts with an initial guess for the solution and stops when the approximation has met the termination criterion. The convergence behaviour of the iterative methods is closely related to the spectrum of the matrix A, i.e. $\sigma(A)$. Provide it is convergent, each iteration step improves the approximation. For large problems, a well-chosen iterative method manages to approximate the solution to the expected precision at a much cheaper cost than the direct method.

Three kinds of iterative methods are working for the solution of Helmholtz problem. The Krylov subspace method is the main solver that solves the entire preconditioned system. The inversion of shifted Laplacian preconditioner is approximated by the multigrid method in which the ω-Jacobi iteration is employed as the smoother.

2.1 Basic Iterative Methods

The class of basic iterative methods annihilates some components of the residual. The iteration process consecutively updates the iterate via a fixed iteration operator. The construction of the iteration operator is based on the decomposition $A = M - N$. In case of an invertible M, the updating formula is given by

$$x_{m+1} = M^{-1}N\, x_m + M^{-1}b. \tag{2.1}$$

Denote $M^{-1}N$ by G as the iteration operator. Then, the error e_m is updated by the relation

$$e_{m+1} = Ge_m.$$

The sufficient condition of convergence requires the spectrum radius be bounded by one, namely $\rho(G) \leqslant 1$.

2.1.1 ω-Jacobi Iteration

The type of iteration depends on the decomposition of the matrix A. One of the most popular type is the ω-Jacobi iteration which splits the diagonal away from A. The splitting is given by

$$M = D \quad \text{and} \quad N = D - A,$$

where D is a diagonal matrix whose diagonal is identical to that of A. Then, the updating formula of the Jacobi iteration is given by

$$x_{m+1} = (I - D^{-1}A)\, x_m + D^{-1}b.$$

The convergence behaviour can be improved by introducing a weight ω, which motivates the ω-Jacobi iteration

$$x_{m+1} = \big((1-\omega)I + \omega(I - D^{-1}A)\big)\; x_m + \omega D^{-1}b.$$

The weight determines how much old information will be used to update the approximation. With a suitable ω, each iteration will be able to reduce the residual by a maximal fraction.

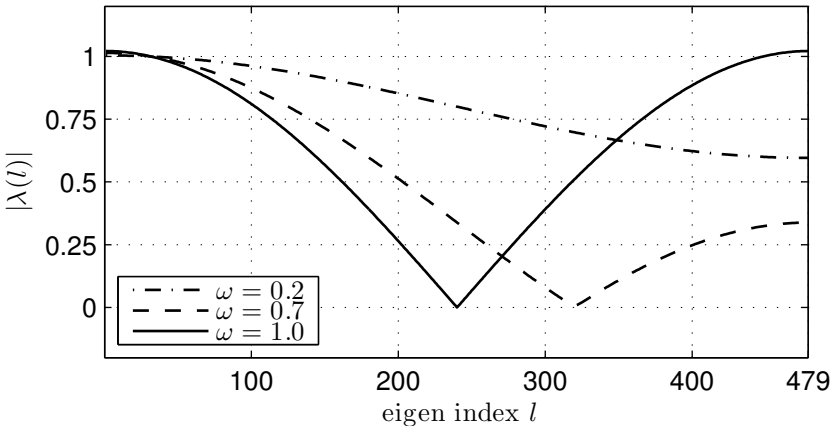


Figure 2.1: The magnitude of eigenvalues of the ω-Jacobi iteration operator for $M = -\Delta_h - \beta k^2 I$ where $k = 100$ and $\beta = 1 - 1\iota$

The detailed analysis of the optimal ω is presented in appendix A for the application to approximating the inversion M^{-1}.

2.2 Multigrid Method

The basic iterative methods are incapable of approximating the inversion of shifted Laplacian preconditioner. The multigrid method [32, 4] is employed as the iterative solver for the linear system

$$Mx = b.$$

2.2.1 Motivation

Assume the system size is n and it is an odd number. The error is spanned by the basis of n eigenvectors of the ω-Jacobi iteration operator, namely n components. The convergence factor of each component in the error is determined by the corresponding eigenvalue.

The eigen index l can be considered as a representation of wave frequency of the component. As shown in Figure 2.1, the eigenvalues of the low frequency (small l) are always far away from zero but close to one while those of the high frequency part (large l) can be adjusted close to zero by an appropriate choice of ω. So the high frequency components ($(n+1)/2 \leqslant l \leqslant n$) in the error are reduced faster than the low ones ($1 \leqslant l \leqslant (n-1)/2$). After some iteration steps, the high frequency components will be almost wiped out while the low frequency part is hardly reduced. In order to accelerate the convergence, the idea is to solve a smaller problem in the coarser grid.

In the coarse grid, the matrix size of the small problem is $(n-1)/2$. Now the eigen index varies from 1 to $(n-1)/2$. Part of the low frequency components ($(n+1)/4 \leqslant l \leqslant (n-1)/2$) in the fine grid becomes high frequency part in the coarse grid. And the amount of low frequency components has been reduced to $(n-3)/4$ in the coarse grid. Figure 2.2 gives an illustration to this idea.

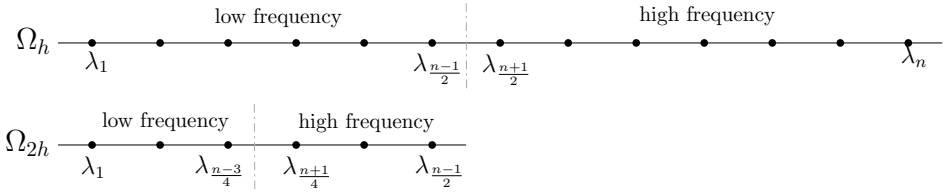


Figure 2.2: The illustration of the eigen relation between fine ad coarse grids

A large problem in the fine grid is transferred into a small problem in the coarse grid. The error will be corrected in the coarse grid. If the small problem is sufficiently small, then it is solved by a direct method. Otherwise the small problem is transferred into an even smaller problem in an even coarser grid. Such a process will not stop until the problem in the coarsest grid is small enough for the direct method. In the process from fine to coarse grids, different components of the error will be gradually reduced level by level until the lowest frequency components is thoroughly removed by a direct solution.

2.2.2 Multigrid Components

The multigrid method consists of several components. They collaborate to complete the multigrid iteration.

Multilevel Grid Assume an m-level grid. For the sake of simplicity, the grid on each level is coarsened by a factor of two. On the first level is the finest grid denoted by Ω_h. And on the lowest level m is the coarsest grid denoted by Ω_{mh}. The subscript is the grid size on each level.

Intergrid Operator Information on different levels is transferred by the intergrid operators. The restriction operator R_k^{k+1} projects quantity from the fine grid on the k-th level onto the coarse grid on the $(k+1)$-th level. When the information is turned back to the fine grid, the prolongation operator P_{k+1}^k interpolates the information of the coarse grid.

There are many choices of the restriction and prolongation operators. Throughput the thesis, the full weighting operator realizes the effect of restriction and the linear interpolation is used for the prolongation operator. In the matrix notation, they reads

$$R_k^{k+1} = \frac{1}{4}\begin{bmatrix} 1 & 2 & 1 & & & & & & \\ & & 1 & 2 & 1 & & & & \\ & & & & \cdots & \cdots & \cdots & & \\ & & & & & & 1 & 2 & 1 \end{bmatrix} \quad \text{and} \quad P_{k+1}^k = \frac{1}{2}\begin{bmatrix} 1 & & & \\ 2 & & & \\ 1 & 1 & & \\ & 2 & & \\ & 1 & \vdots & \\ & & \vdots & \\ & & \vdots & 1 \\ & & & 2 \\ & & & 1 \end{bmatrix}$$

Suppose the size of R_k^{k+1} and P_{k+1}^k are $a \times b$ and $b \times a$ respectively. Then, a and b should satisfy $a = \frac{b-1}{2}$.

Smoother There are two types of smoothing in one multigrid iteration. The smoothing operator is denoted by S. The pre-smoothing happens before the problem is restricted to the coarse grid. The post-smoothing happens after the problem has been prolongated onto the fine grid. The amount of pre- and post- smoothing steps are denoted by μ_1 and μ_2 respectively. Throughout this thesis, the smoothing is implemented by the ω-Jacobi iteration.

2.2.3 Multigrid Cycle

Consider the two-grid problem. The approximation x_m is obtained by m steps of iteration. The pre-smoothing by μ_1 ω-Jacobi iterations results in the approximation $\bar{x}_{m,1}$, which defines the residual

$$\bar{d}_{m,1} := b - M\bar{x}_{m,1}.$$

The residual on the first level is restricted to the second level by the restriction operator as

$$\bar{d}_{m,2} := R_1^2\, \bar{d}_{m,1}.$$

The coarse grid operator M_2 is obtained by the Galerkin projection as

$$M_2 := R_1^2\, M_1\, P_2^1 \quad \text{where } M_1 = M.$$

The error is corrected by a direct method, which yields

$$\hat{y}_{m,2} := M_2^{-1}\, \bar{d}_{m,2}.$$

The error correction $\hat{y}_{m,2}$ is then projected onto the first level by prolongation operator as

$$\hat{y}_{m,1} := P_2^1\, \hat{y}_{m,2}.$$

The approximation is now corrected by

$$\bar{x}_{m,1} = \bar{x}_{m,1} + \hat{y}_{m,1}.$$

The application of μ_2 ω-Jacobi iterations to the updated $\bar{x}_{m,1}$ yields the new approximation x_{m+1}. The iteration process can be expressed in the matrix form. The two-grid operator is given by

$$T_1^2 := S^{\mu_2}\, (I - P_2^1 M_2^{-1} R_1^2 M_1)\, S^{\mu_2}.$$

Here, the coarse grid correction operator can be defined as

$$K_1^2 := I - P_2^1 M_2^{-1} R_1^2 M_1.$$

The process of two-grid correction can be extended to the multigrid correction by recursively applying the two-grid correction to solving the inversion on the second level. The multigrid iteration operator is given by the recursion

$$T_k = S_k^{\nu_2}(I - P_{k+1}^k\, (I - T_{k+1}) A_{k+1}^{-1}\, R_k^{k+1} A_k) S_k^{\nu_1} \quad \text{with } T_m = 0, \tag{2.2}$$

Here, the number γ is the cycle index. On the k-th level, the problem on the $(k+1)$-th level is solved by γ multigrid iterations. Three typical types are illustrated in Figure 2.3. The F-cycle refers to an index $\gamma = \gamma_k$ that depends on the level.

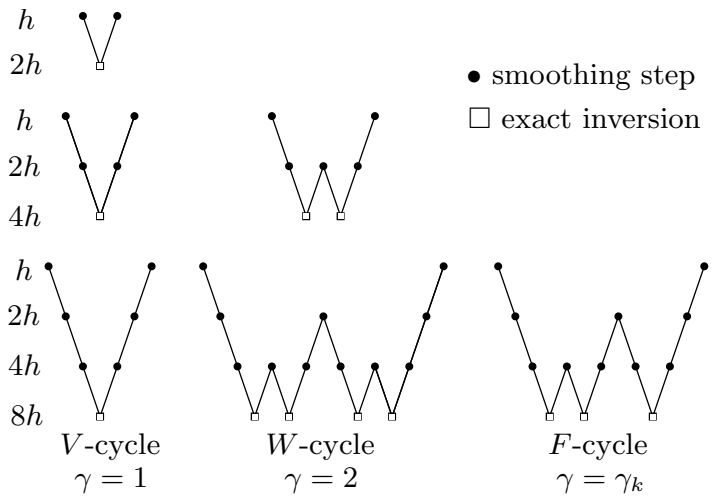


Figure 2.3: The illustration of multigrid cycles

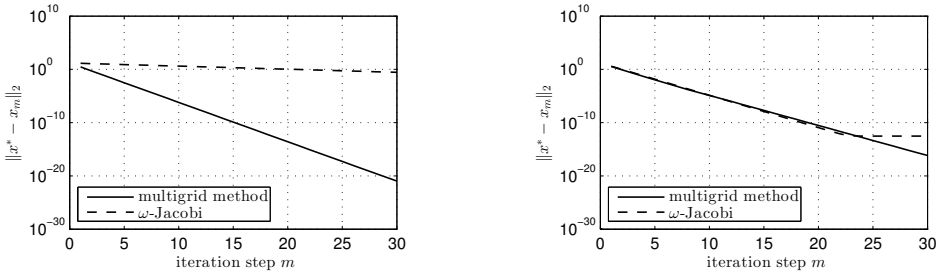
2.2.4 An Illustrative Example

An experiment is conducted to show the advantage of multigrid method over basic iterative methods. Both multigrid method and ω-Jacobi iteration are used to solve the linear system $Mx = b$ which results from the model problem

$$-u'' - \beta k^2 u = 0 \quad \text{where } k = 100 \text{ and } \beta = 1 - 1\iota.$$

The model problem has a zero source term (right hand side) so the exact solution is zero. For ω-Jacobi iteration, by choosing the initial guess as one of the eigenvector of the ω-Jacobi operator, the error reduction only depends on the magnitude of the corresponding eigenvalue.

The system size is $n = 479$. The initial guess of eigenvector w_{100} represents the low frequency component while that of w_{400} represents the high frequency part. The convergence behaviour is shown in Figure 2.4. It is clear that ω-Jacobi iteration can hardly reduce the error of low frequency but multigrid method has successfully reduced the error in both cases.



(a) low frequency initial guess

(b) high frequency initial guess

Figure 2.4: The comparison of convergence behaviour between ω-Jacobi iteration and multigrid method

2.3 Krylov Subspace Methods

Krylov subspace methods are a set of iterative methods that can efficiently solve large sparse linear systems. The Krylov solution can be considered as the projection processes onto Krylov subspaces. Different specific algorithms are developed to fit different kinds of problems. They

are distinguished by the way how the approximation is built, which therefore determines the type of error minimization and presents corresponding convergence behaviour.

The fundamental idea of Krylov iteration will be explained in the next section, which are followed by the introduction to several typical Krylov methods.

2.3.1 Krylov Subspace

For a linear system $Ax = b$, the projection method takes an approximation x_m from the m-dimensional search subspace $\mathcal{K}$ by imposing the Petrov-Galerkin condition

$$b - Ax_m \perp \mathcal{L}_m.$$

Here, $\mathcal{L}_m$ is the subspace of constraints. With an arbitrary initial guess x_0 of the solution, the search subspace can be denoted by $\mathcal{K} = x_0 + \mathcal{K}_m$.

A Krylov subspace method is the one that uses Krylov subspace to build the search subspace with

$$\mathcal{K}_m = \text{span}\{r_0, Ar_0, A^2r_0, \cdots, A^{m-1}r_0\}, \tag{2.3}$$

where $r_0 = b - Ax_0$ is the initial residual. Different Krylov methods vary in the choice of the subspace of constraints. A simple choice is $\mathcal{L}_m = \mathcal{K}_m$ or $\mathcal{L}_m = A\mathcal{K}_m$ for the orthogonalization methods. Another class of choice uses A^* to construct the subspace as

$$\mathcal{L}_m = \text{span}\{r_0, A^*r_0, (A^*)^2r_0, \cdots, (A^*)^{m-1}r_0\}. \tag{2.4}$$

This forms the origin of the biorthogonalization methods.

Orthonormality In practical implementation an orthonormal basis of Krylov subspace is used instead of the straightforward form (2.3) so that the numerical calculation is stable and insensitive to the rounding error.

The orthonormal basis can be generated by Arnoldi iteration [1] which actually implements the modified Gram-Schmidt orthogonalization. At the m-th step, the following relations hold:

$$AV_m = V_{m+1}\tilde{H}_m \quad \text{and} \quad V_m^*AV_m = H_m, \tag{2.5}$$

where the columns of V_m forms the orthonormal basis of Krylov subspace and H_m is obtained by deleting the last line of the $(m+1) \times m$ Hessenberg matrix $\tilde{H}_m$.

Anther approach to construct the orthonormal basis is to apply the Householder algorithm. It computes the QR factorization of the subspace (2.3). From a numerical viewpoint, Householder algorithm is more robust at the cost of more computational resources.

Symmetry When the matrix A is Hermitian, the Arnoldi's algorithm can be simplified to the Lanczos algorithm. The Hessenberg matrix H_m turns out to be Hermitian and tridiagonal. Fast algorithm is based on the three-term recurrences instead of $(m+1)$-term at step m. The relation (2.5) becomes

$$AV_m = V_{m+1}\tilde{T}_m \quad \text{and} \quad V_m^*AV_m = T_m. \tag{2.6}$$

Biorthogonalization Lanczos algorithm can be extended to the non-Hermitian matrix by building a pair of biorthogonal bases

$$\mathcal{K}_m(A, v_1) = \text{span}\{v_1, Av_1, \cdots, A^{m-1}v_1\}$$

and

$$\mathcal{K}_m(A, w_1) = \text{span}\{w_1, A^*w_1, \cdots, (A^*)^{m-1}w_1\}.$$

The two subspaces should satisfy the biorthogonal condition $W_m^* V_m = I_m$. The computation is then doubled since the orthogonalization happens in both subspaces at each step. Similar to the application of Lanczos process to Hermitian matrix, the relations for biorthogonalization methods hold in terms of corresponding Hessenberg matrices as

$$\begin{aligned} AV_m =& V_{m+1}\tilde{T}_m \\ A^*W_m =& W_{m+1}\tilde{S}_m \\ W_m^*AV_m =& \tilde{T}_m = \tilde{S}_m^*. \end{aligned}$$

2.3.2 GMRES

The generalized minimal residual method (GMRES) [27] is based on choosing $\mathcal{L}_m = A\mathcal{K}_m$ as the subspace of constraints. Among all the vectors in the subspace $x_0+\mathcal{K}_m$, the approximation x_m is supposed to have the minimal residual. GMRES has no additional requirement on the matrix A and can be applied to any type of matrices.

The derivation of GMRES exploits the optimality property. At the m-th iteration step, the vector in the search subspace $x_0 + \mathcal{K}_m$ is written as

$$x_m = x_0 + V_m y,$$

where y is an $m \times 1$ vector. With the help of orthogonal relation (2.6), the residual can be simplified as

$$\begin{aligned} r_m =& b - Ax_m \\ =& r_0 - AV_m y \\ =& \beta v_1 - V_{m+1}\tilde{H}_m y \\ =& V_{m+1}(\beta e_1 - \tilde{H}_m y). \end{aligned}$$

As a function of the vector y, the residual norm can be written as

$$\begin{aligned} J(y) =& \|r_m\|_2 \\ =& \|V_{m+1}(\beta e_1 - \tilde{H}_m y)\|_2 \\ =& \|\beta e_1 - \tilde{H}_m y)|_2 \end{aligned} \tag{2.7}$$

So the GMRES algorithm finds a unique vector y so that the approximation $x_m = x_0 + V_m y \in x_0 + \mathcal{K}_m$ minimizes the residual function $J(y) = \|\beta e_1 - \tilde{H}_m y)|_2$. The task is then transformed into solving an $(m+1) \times m$ least-square problem. Although the original problem size n is very large, m is typically small enough for a direct solution. The sketch of GMRES is given in algorithm 1 on the following page.

Breakdown Algorithm 1 will stop if the condition on line 10 is not satisfied. In this situation the next Arnoldi vector cannot be generated and the residual vector is zero, which means the iteration has reached the exact solution.

Restarting As the iteration goes on and the dimension of Krylov subspace increases, the growing storage requirement makes the algorithm impractical. One remedy is to restart the Arnoldi orthogonalization. The approximate solution of the previous loop and its residual is used as the initial guess for the next loop. However, this approach sometime will stagnate if the matrix is not positive definite.

Algorithm 1 GMRES with modified Gram-Schmidt orthogonalization

1: Choose an initial guess x_0
2: Compute $r_0 = b - Ax_0$, $\beta = \|r_0\|_2$ and $v_1 := r_0/\beta$
3: **for** $j = 1, 2, \cdots$ until convergence **do**
4: Compute $w_j := Av_j$
5: **for** $i = 1, 2, \cdots, j$ **do**
6: $h_{ij} := (w_j, v_i)$
7: $w_j := w_j - h_{ij}v_i$
8: **end for**
9: $h_{j+1,j} := \|w_j|_2$
10: **if** $h_{j+1,j} \neq 0$ **then**
11: $v_{j+1} := w_j/h_{j+1,j}$
12: **else**
13: go to line 16
14: **end if**
15: **end for**
16: Build the $(m+1) \times m$ Hessenberg matrix $\tilde{H}_m = \{h_{i,j}\}$
17: Find $y =: \text{argmin}_y \|\beta e_1 - \tilde{H}_m y\|_2$
18: Build $V_j := [v_1, v_2, \cdots, v_j]$
19: Compute $x_j = x_0 + V_j y$ as the resulting approximation.

Convergence Let $p \in P_m$ where P_m is the set of polynomials whose degree are not higher than m and satisfy $p(0) = 1$. For a diagonalizable matrix which can be decomposed as $A = X^{-1}\Lambda X$, the convergence rate of GMRES reads

$$\frac{\|r_m\|_2}{\|r_0\|_2} \leqslant \kappa_2(X) \inf_{p \in P_m} \max_{\lambda \in \Lambda(A)} |p(\lambda)|. \tag{2.8}$$

where $\kappa_2 = \|X^{-1}\|_2 \|X\|_2$. Theoretically, GMRES iteration for an $n \times n$ system converges to the exact solution in at most n steps or when $r_m = 0$. But this has little practical content since a useful GMRES must converge to the desirable precision in $m \ll n$ steps.

2.3.3 CG

The conjugate gradient method (CG) is among the most important iterative techniques for solving sparse linear systems which are Hermitian and positive definite [19]. CG algorithm takes $\mathcal{L}_m = \mathcal{K}_m$ as the subspace of constraints. The Hermitian property of A makes it feasible to apply Lanczos iteration. The approximate solution x_m minimizes the A-norm of the error

$$\|e\|_A = \sqrt{e^* A e}, \tag{2.9}$$

which is made meaningful by the positive definite property.

The CG algorithm, originally as a direct method for optimality, can be derived by imposing the orthogonality condition to the residual

$$(r_i, r_j) = 0 \quad \text{for } i \neq j,$$

and conjugacy condition to the search direction

$$(Ap_i, p_j) = 0 \quad \text{for } i \neq j.$$

The approximation x_{m+1} can be expressed in a recursive form as

$$x_{m+1} = x_m + \alpha_m p_m.$$

and the new search direction is a linear combination of the old direction and the residual

$$p_{m+1} = r_{m+1} + \beta_m p_m.$$

The orthogonality necessitates

$$\alpha_m = \frac{(r_m, r_m)}{(Ap_m, p_m)}.$$

and the consequence of the conjugacy follows as

$$\beta_m = \frac{(r_{m+1}, r_{m+1})}{(r_m, r_m)}.$$

Combining these relations together forms the whole algorithm.

Algorithm 2 CG

1: Choose an initial guess x_0
2: Compute $r_0 = b - Ax_0$ and take $p_0 := r_0$
3: **for** $j = 1, 2, \cdots$ until convergence **do**
4: $\alpha_j := (r_j, r_j)/(Ap_j, p_j)$
5: $x_{j+1} := x_j + \alpha_j p_j$
6: $r_{j+1} := r_j - \alpha_j A p_j$
7: $\beta_{j+1} := (r_{j+1}, r_{j+1})/(r_j, r_j)$
8: $p_{j+1} := r_{j+1} + \beta_j p_j$
9: **end for**

Convergence The convergence rate of CG can be expressed with respect to the error in A-norm

$$\frac{\|e_m\|_A}{\|e_0\|_A} \leqslant \inf_{p \in P_m} \max_{\lambda \in \Lambda(A)} |p(\lambda)|. \tag{2.10}$$

With the help of Chebyshev polynomials, a slightly different formulation comes out as

$$\frac{\|e_m\|_A}{\|e_0\|_A} \leqslant 2 \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^m, \tag{2.11}$$

where κ is the spectral condition number $\kappa = \lambda_{\max}/\lambda_{\min}$.

CGNR The application of CG method is limited to Hermitian matrix of positive definiteness. For any non-singular but not necessarily Hermitian matrix, one of the simplest methods is to apply the CG algorithm to the normal equation (CGNR)

$$A^*Ax = A^*b.$$

The advantage of three-term recurrence is used but the computation will now requires two matrix-vector multiplications on line 4 in algorithm 2. The 2-norm residual is minimized over the Krylov subspace at every step as $\|e\|_{A^*A} = \|Ae\|_2 = \|r\|_2$. Furthermore, the condition number in equation (2.11) is squared, which is very likely to make the convergence far worse.

2.3.4 Bi-CGSTAB

The biconjugate gradient stabilized (Bi-CGSTAB) method [34] is a transpose-free biorthogonalization algorithm developed from the conjugate gradient squared (CGS) method [30]. It improves the irregular convergence of CGS by delivering a residual vector of the form

$$r'_j = \psi_j(A)\phi_j(A)r_0, \tag{2.12}$$

in which both $\psi(t)$ and $\phi(t)$ are polynomials of j-th degree. Additionally, $\psi(t)$ is defined recursively as

$$\psi(t)_{j+1} = (1 - \omega_j t)\psi(t)_j.$$

A natural choice for the parameter ω_j is to achieve a steepest descent step in the original residual direction $\phi_j(A)r_0$, which actually minimizes $\|(I - \omega_j A)\phi_j(A)r_0\|_2$. As a variant of CGS, Bi-CGSTAB compute other parameters for the iteration process in a similar way to CGS.

Algorithm 3 Bi-CGSTAB

1: Choose an initial guess x_0
2: Compute $r_0 := b - Ax_0$ and choose a r_0^* such that $(r_0^*, r_0) \neq 0$
3: Take $p_0 := r_0$
4: **for** $j = 0, 1, \cdots$ until convergence **do**
5: $\quad \alpha_j := (r_j, r_0^*)/(Ap_j, r_0^*)$
6: $\quad s_j := r_j - \alpha_j Ap_j$
7: $\quad \omega_j := (As_j, s_j)/(As_j, As_j)$
8: $\quad x_{j+1} := x_j + \alpha_j p_j + \omega_j s_j$
9: $\quad r_{j+1} := s_j - \omega_j As_j$
10: $\quad \beta_j := \frac{(r_{j+1}, r_0^*)}{(r_j, r_0^*)} \times \frac{\alpha_j}{\omega_j}$
11: $\quad p_{j+1} := r_{j+1} + \beta_j(p_j - \omega_j Ap_j)$
12: **end for**

2.3.5 More Biorthogonalization Methods

Besides CGS and Bi-CGSTAB there are more Krylov subspace methods based on a biorthogonalization algorithm. The Arnoldi iteration is replaced with Lanczos iteration and the three-term recurrence process can be applied to the non-Hermitian matrices.

BCG In biconjugate gradient (BCG) method [13] the orthogonal relation is replaced with the biorthogonal relation as

$$(r_j, r_i^*) = 0 \quad \text{for } i \neq j \tag{2.13}$$
$$(Ap_j, p_i^*) = 0 \quad \text{for } i \neq j. \tag{2.14}$$

The total computational work is doubled due to the matrix-by-vector product with both A and A^*.

QMR Quasi-minimal residual (QMR) method [16] can be regarded as the biorthogonal version of GMRES. The QMR approximation minimizes the quasi-residual norm $\|\beta e_1 - \tilde{T}_m y\|_2$ rather than the complete residual $\|V_{m+1}(\beta e_1 - \tilde{T}_m y)\|_2$ since the columns of V_{m+1} is no longer orthogonal.

Transpose-free variants The transpose-free variants, for instance CGS [30], Bi-CGSTAB and TFQMR [15], can bypass the use of A^* in BCG and QMR. The idea is based on the fact that in biorthogonalization process the residual with respect to both A and A^* can be expressed in a polynomial form as

$$r_j = \phi_j(A)r_0 \quad \text{and} \quad r*_j = \phi_j(A^*)r_0^*.$$

Then, the inner product of the vectors in polynomial form can be given by

$$\left(\phi_j(A)r_0, \phi_j^*(A^*)r_0^*\right) = \left(\phi_j^2(A)r_0, r_0^*\right).$$

Hence, the idea is to find out a sequence of iterates whose residual have the form

$$r'_j = \phi_j^2(A)r_0.$$

Similar implementation is also applied to the conjugate direction. Thus, A^* is not explicitly used and faster convergence is obtained at a roughly same computational cost as BCG or QMR.

2.4 Modified System and Approximated Inversion

2.4.1 Iteration Operator

Without loss of generality an iterative method can be introduced by starting from the approximation of the residual (or defect) equation. For any approximation x_m of the exact solution x^* to the linear system $Ax = b$, the error is denoted by

$$e_m := x^* - x_m,$$

and the residual is denoted by

$$r_m := b - Ax_m.$$

Since $x^* = x_m + e_m$, to solve the original system is equivalent to solve the system of residual

$$Ae_m = r_m.$$

Numerically, the operator matrix in the residual equation is replaced with another simpler operator $\mathbb{A}$ such that the inversion of $\mathbb{A}$ can be computed in an easier way. Then, the solution $\hat{e}_m$ to

$$\mathbb{A}e_m = r_m$$

gives a new approximation

$$\begin{aligned} x_{m+1} :=& x_m + \hat{e}_m \\ =& x_m + \mathbb{A}^{-1}r_m \\ =& (I - \mathbb{A}^{-1}A)x_m + \mathbb{A}^{-1}b. \end{aligned}$$

The iterate process is now given by

$$x_{m+1} = Gx_m + f \quad \text{for } m = 0, 1, \cdots,$$

where $f = \mathbb{A}^{-1}b$ and the iteration operator is defined as

$$G := I - \mathbb{A}^{-1}A. \tag{2.15}$$

Then, the error is updated by

$$e_{m+1} = Ge_m = (I - \mathbb{A}^{-1}A)e_m, \tag{2.16}$$

and the residual is updated by

$$r_{m+1} = (I - A\mathbb{A}^{-1})r_m. \tag{2.17}$$

Each iteration can be considered as solving a modified linear system whose matrix $\mathbb{A}$ is exactly inverted. The approximation x_{m+1} is actually the exact solution to the *modified system*

$$\mathbb{A}x_{m+1} = \underbrace{(\mathbb{A} - A)x_m + b}_{\textit{modified RHS}}.$$

By rearranging (2.15), the expression of *approximated inversion*[1] can be given by

$$\mathbb{A}^{-1} = (I - G)A^{-1}. \tag{2.18}$$

The modified system is associated with the approximated inversion $\mathbb{A}^{-1}$, which results from one iteration for approximating the exact inversion A^{-1}.

The expression of $\mathbb{A}^{-1}$ is useless in practical computation and is impossible to obtain in matrix form since A^{-1} is never explicitly available. But $\mathbb{A}^{-1}$ is valuable in the theoretical analysis. The knowledge of $\mathbb{A}^{-1}$ gives an important insight into the convergence behaviour of the iteration operator. The iteration operator G plays a role of perturbation in the iteration process. An effective iteration operator, which yields a fast convergence, should be close to zero matrix and therefore have the approximated inversion $\mathbb{A}^{-1}$ less deviated away from the exact inversion A^{-1}.

2.4.2 More about Basic Iterative Methods

In the basic iterative methods the simpler operator $\mathbb{A}$ can be obtained from the operator of any stationary iterative method, for instance ω-Jacobi iteration. Given the initial guess x_0, the approximation x_m can be obtained by induction.

$$\begin{aligned} x_m =& G^m x_0 + \sum_{i=0}^{m-1} G^i \mathbb{A}^{-1} b \\ =& G^m x_0 + (I - G^m)(I - G)^{-1}\mathbb{A}^{-1} b \\ =& G^m x_0 + (I - G^m)A^{-1} b \\ \approx& (I - G^m)A^{-1} b \end{aligned}$$

The last approximate equality follows the fact that an efficient iteration should reduce $\|G^m\|$ to a very small and negligible quantity for large m. In case of $x_0 = 0$, it now turns out to be that x_m is the exact solution to the modified system

$$\left((I - G^m)A^{-1}\right)^{-1} x = b$$

and the approximated inversion after m iterations is denoted by

$$\mathbb{A}_m^{-1} = (I - G^m)A^{-1}. \tag{2.19}$$

2.4.3 More about Multigrid Method

The iteration operator for a two-grid method reads

$$T_1 = S_1^{\nu_2}(I - P_2^1 A_2^{-1} R_1^2 A_1) S_1^{\nu_1}.$$

When more than two levels of grids are used, the inversion A_2^{-1} is approximately obtained by the application of another two-grid method on the coarser grid. Using the relation (2.18), the approximation is given by

$$\mathbb{A}_2^{-1} = (I - T_2)A_2^{-1}.$$

Thus, by induction, the recursive definition of the iteration operator for a m-level multigrid method is

$$T_k = S_k^{\nu_2}(I - P_{k+1}^k (I - T_{k+1}) A_{k+1}^{-1} R_k^{k+1} A_k) S_k^{\nu_1} \quad \text{with } T_m = 0,$$

for $k = 1, 2 \cdots, m$. And the approximated inversion is given by

$$\mathbb{A}_{\mathrm{MG}}^{-1} = (I - T_1^m)A^{-1}, \tag{2.20}$$

[1] In the remaining part of the thesis the approximated inversion is denoted by the double-stroke symbol.

where T_1^m denotes the iteration operator in an m-level grid.

In case of W-cycle, or more generally the cycle index $\gamma > 1$, the inversion on the coarse grid is approximated by γ steps of multigrid iterations, which is implemented by

$$\mathbb{A}_{k+1}^{-1} = (I - M_{k+1}^{\gamma})A_{k+1}^{-1}.$$

2.4.4 Iteration Operator for Krylov Subspace Methods

Krylov subspace methods are not within the class of stationary iteration because the increasing Krylov subspace at every step generates a dynamic iteration. However, although no stationary iteration operator can be found, the error/residual after m iterations is related to the initial error/residual by a matrix-vector multiplication. At the m-th step the polynomial matrix that minimizes a certain norm of the error/residual plays the role of error/residual reduction operator and the entire effect of all the m iteration steps can be considered as one single iteration. Thus, the knowledge of reduction operator is sufficient for the derivation of iteration operator. Finally, the approximated inversion can be expressed in the same way as what is done for the stationary method.

Residual Reduction Operator of GMRES

In the GMRES iteration the residual after m iterations is given by

$$r_m = p_m(A)r_0,$$

where the polynomial p_m of degree m is chosen to minimize $\|r_m\|_2$. Now the polynomial matrix $p_m(A)$ is the residual reduction operator. Using equation (2.17), the approximated inversion by m steps of GMRES iterations is given by

$$\mathbb{A}^{-1} = A^{-1}\left(I - p_m(A)\right). \tag{2.21}$$

Error Reduction Operator of CG

After m steps of CG iterations, the error is given by

$$e_m = q_m(A)e_0,$$

which is minimized in A-norm by the polynomial q_m of degree m. The polynomial matrix $q_m(A)$ is the equivalent error reduction operator which plays the same role as G in equation (2.16). So the approximated inversion by m steps of CG iteration is given by

$$\mathbb{A}^{-1} = \left(I - q_m(A)\right)A^{-1}. \tag{2.22}$$

It is possible to extend the above approach to other Krylov subspace methods only if the expression of either error of residual can be obtained in terms of reduction operator. Once the corresponding polynomial is found, the approximated inversion can be obtained in the concrete form. Therefore, the investigation in spectral properties will reveal the Krylov convergence behaviour.

Chapter 3

Preconditioning Techniques

The linear system that results from the Helmholtz problem is an ill-conditioned matrix because of its high indefiniteness. The direct application of Krylov subspace method will suffer from slow convergence due to the unfavourable spectral properties of the original matrix A. The key to the success of Krylov method in the Helmholtz problem is the preconditioning technique.

There are three ways of applying the preconditioning matrix M, which are from the left, the right and from both sides in a split form. Different ways of preconditioning results in different residuals. This fact may affect the stopping criterion and then cause the iteration process to stop either prematurely or with delay. However, in most situations, the difference is not important in the convergence behaviour of different preconditionings.

As a qualified preconditioning matrix, the preconditioner M is supposed to satisfy the following basic requirements.

- It should be easy to obtain the inversion M^{-1}, which requires that it be inexpensive to solve the linear system $Mx = b$.
- The preconditioner M should be close to the original matrix A to some extent and also be nonsingular. The closer M to A is, the closer AM^{-1} to the identity matrix I is, and then the easier the Krylov convergence will be.
- Generally the preconditioning should preserve the symmetry if the original matrix A is symmetric.

In this chapter the preconditioners specializing in Helmholtz problem will be introduced and their properties will be discussed. Besides that a special method will be introduced to approximately construct these preconditioners. Throughout this thesis, the right preconditioning will be employed during the analysis and the computation. Because the right preconditioning leads to the same residual as the original one.

3.1 Shifted Laplacian Preconditioner

The application of preconditioning to the Helmholtz problem started from [2] which used the Laplacian operator as the preconditioner. The preconditioning was improved in [21] by adding an extra term to the Laplacian operator. Later on the work was extended and generalized in [11]. This class of preconditioners is named *shifted Laplacian preconditioner*.

The construction of the shifted Laplacian preconditioner is based on the matrix A that results from the discretized Helmholtz equation. In the 1D Helmholtz problem with Dirichlet boundary condition, the matrix A is given by

$$A := -\Delta_h - k^2 I,$$

where Δ_h is the discrete Laplacian operator. Then, the shifted Laplacian preconditioner is defined as

$$M := -\Delta_h - (\beta_1 + \iota\beta_2)k^2\, I \quad \text{for } \beta_1, \beta_2 \in \mathbb{R}. \tag{3.1}$$

Eigen Relation It is self-evident that both the left and right preconditioned matrices share the same eigenvalues

$$\lambda(M^{-1}A) = \lambda(AM^{-1}). \tag{3.2}$$

Furthermore, the connection between A and M leads to the following relation

$$A = M + (\beta_1 + \iota\beta_2 - 1)k^2\, I,$$

which is followed by the equality

$$\begin{aligned} M^{-1}A &= M^{-1}\left(M + (\beta_1 + \iota\beta_2 - 1)k^2\, I\right) \\ &= I + (\beta_1 + \iota\beta_2 - 1)k^2\, M^{-1} \\ &= \left(M + (\beta_1 + \iota\beta_2 - 1)k^2\, I\right) M^{-1} = AM^{-1}. \end{aligned}$$

So either the left or right preconditioning actually results in the same preconditioned matrix that is denoted by

$$\hat{A} := M^{-1}A = AM^{-1}. \tag{3.3}$$

For the sake of consistent residual[1], the right version of the shifted Laplacian preconditioner will be used in the rest of the thesis.

Symmetry The original matrix A is real and symmetric. The shifted Laplacian preconditioner M is not Hermitian but only symmetric. The shifted Laplacian preconditioning preserves the symmetry. With the help of (3.3), the proof is given by the following deduction

$$(AM^{-1})^T = M^{-T}A^T = M^{-1}A = AM^{-1}. \tag{3.4}$$

3.1.1 Spectrum Distribution

The eigenvalue of the discrete Laplacian operator Δ_h is given by

$$\mu_l = \frac{4}{h^2}\sin^2(l\pi h/2) \quad \text{for } l = 1, \cdots, n.$$

So the eigenvalue of the preconditioned matrix $\hat{A}$ is given by

$$\lambda_l = \frac{\mu_l - k^2}{\mu_l - (\beta_1 + \iota\beta_2)k^2} \quad \text{for } l = 1, \cdots, n. \tag{3.5}$$

The detailed spectral analysis of the preconditioned matrix is presented in [11, 35] for different choices of the shift $\beta_1 + \iota\beta_2$. One of the most important conclusions is that the eigenvalues are distributed on a circle.

Proof of the circular sprectrum distribution. Consider the 1D Helmholtz problem with Dirichlet boundary condition and assume $\beta_2 \neq 0$.

The eigenvalue of the preconditioned matrix can be written in the format of the complex number as

$$\lambda = \lambda_r + \iota\lambda_i. \tag{3.6}$$

[1]The Krylov convergence of the right preconditioned system shares the same residual as that of the original system.

Substitution of (3.6) into (3.5) yields the equality

$$\lambda_r(\mu - \beta_1 k^2) + \lambda_i \beta_2 k^2 - (\mu - k^2) - \iota \left(\lambda_r \beta_2 k^2 - \lambda_i(\mu - \beta_1 k^2)\right) = 0.$$

By equating the real and imaginary parts respectively, the above equality can be transformed into

$$(\lambda_r - 1)\mu = (\lambda_r \beta_1 - \lambda_i \beta_2 - 1)k^2 \tag{3.7}$$

$$\lambda_i \mu = (\lambda_r \beta_2 + \lambda_i \beta_1)k^2 \tag{3.8}$$

The right hand sides of the above two equations can be connected by multiplying (3.7) by λ_i and (3.8) by $(\lambda_r - 1)$. The resulting equality that excludes the term k^2 is given by

$$\beta_2 \lambda_r^2 - \beta_2 \lambda_r + \beta_2 \lambda_i^2 - (\beta_1 - 1)\lambda_i = 0.$$

In case of $\beta_2 \neq 0$, the above relation can be transformed into a quadratic form as

$$(\lambda_r - \frac{1}{2})^2 + (\lambda_i - \frac{\beta_1 - 1}{2\beta_2})^2 = \frac{\beta_2^2 + (1 - \beta_1)^2}{(2\beta_2)^2}. \tag{3.9}$$

Equation (3.9) is the expression of a circle that is centered at $(\frac{1}{2}, \frac{\beta_1 - 1}{2\beta_2})$ with the radius $\frac{\sqrt{\beta_2^2 + (1-\beta_1)^2}}{|2\beta_2|}$. Thus, the spectrum distribution is circular. □

The circular distribution explains the functionality of the shifted Laplacian preconditioning. All the eigenvalues are now on a circle which is a much tighter shape than that without preconditioning. The tightness is determined by the shift. The choice of $\beta_1 = 1$ results in the tightest circular distribution which is centered at $(\frac{1}{2}, 0)$ with the radius $\frac{1}{2}$. A basic illustration of the shifted Laplacian preconditioning is presented in Figure 3.1.

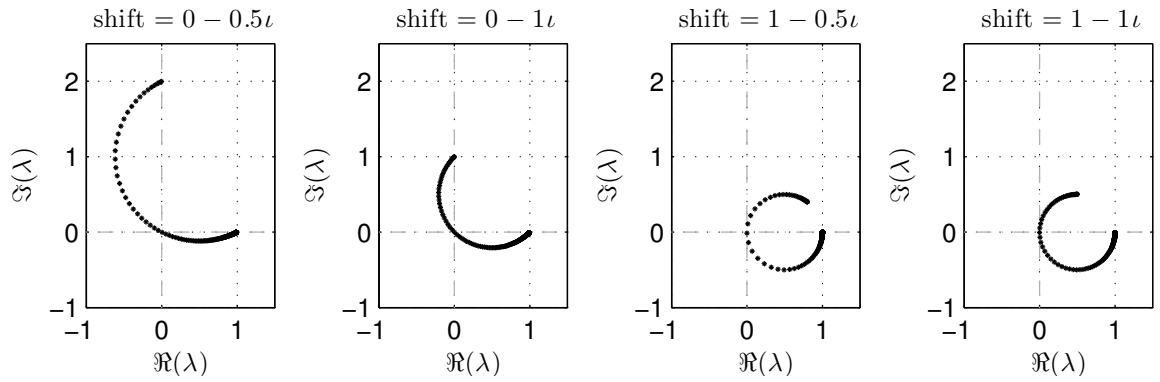


Figure 3.1: The spectrum distributions of the preconditioned matrix AM^{-1} with respect to several typical shifts when $k = 100$

The shifted Laplacian preconditioning will still result in some very small eigenvalues that are close to zero. So it is not necessarily that the condition number of the preconditioned matrix is much smaller than that of the original matrix. But the important effect is that the more tightly clustered spectrum distribution is favourable for the Krylov convergence.

3.2 Deflation Operator

The shifted Laplacian preconditioning succeeds in restricting the eigenvalues to a tight shape but still leaves a certain amount of small eigenvalues. The eigenvalue that is close to zero will slow down the Krylov convergence.

The deflation preconditioning, which was discussed in [10] specifically for Helmholtz problem, is used to overcome the weakness of the shifted Laplacian preconditioning by projecting the smallest eigenvalues towards the maximal eigenvalue. The deflation preconditioning is applied to the system that has been already preconditioned by the shifted Laplacian operator.

The early research on the deflation preconditioning was made in [24] for CG and in [22] for GMRES. Some recent research on deflation preconditioning is presented in [14, 23, 8], which provide the theoretical support for the application of deflation preconditioning to the Helmholtz problem.

3.2.1 The Motivation of Deflation

The idea of deflation preconditioning is related to the power method that iteratively computes the eigenvalue of largest or smallest magnitude. By using *Wielandt deflation*[5], the power method will be able to solve the eigenvalue of second largest of smallest magnitude.

Denote the spectrum of the preconditioned matrix $\hat{A}$ by $\sigma(\hat{A}) = \{\lambda_1(\hat{A}), \lambda_2(\hat{A}), \cdots, \lambda_n(\hat{A})\}$ with the corresponding eigenvectors $z_1, z_2, \cdots, z_n$. The sequence satisfies the condition $|\lambda_1| \leqslant |\lambda_2| \leqslant \cdots \leqslant |\lambda_n|$. The process of deflating λ_1 towards zero is given by

$$\hat{A}_1 = \hat{A} - \lambda_1 z_1 y^T,$$

where y is any vector that satisfies the requirement $y^T z = 1$. A generalized version of Wielandt deflation is given by

$$\hat{A}_{1,\gamma_1} = \hat{A} - \gamma_1 z_1 y^T \quad \text{where } \gamma_1 \text{ is arbitrary.} \tag{3.10}$$

Theorem 3.1. *The spectrum of the deflated matrix $\hat{A}_{1,\gamma_1}$ is given by*

$$\sigma(\hat{A}_{1,\gamma_1}) = \{\lambda_1 - \gamma_1, \lambda_2, \cdots, \lambda_n\}.$$

Proof. For $i = 1$, the condition $y^T z = 1$ leads to the following derivation

$$\hat{A}_{1,\gamma_1}\, z_1 = \hat{A}\, z_1 - \gamma_1 z_1 y^T\, z_1 = (\lambda_1 - \gamma_1)\, z_1.$$

For $i \neq 1$, denote y_i as the left eigenvector. The relation $y_i^T z_1 = 0$ leads to the following derivation

$$y_i^T\, \hat{A}_{1,\gamma_1} = y_i^T\, \hat{A} - \gamma_1 y_i^T\, z_1 y^T = y_i^T\, \hat{A} = y_i^T \lambda_i.$$

Thus, the first eigenvalue of $\hat{A}$ is deflated towards $\lambda_1 - \gamma_1$ while the other eigenvalues remain unchanged. And all the eigenvectors are still the same. □

There is no restriction on the choice of γ_1. Besides the choice $\gamma_1 = \lambda_1$ deflating λ_1 towards zero, there is another useful choice $\gamma_1 = \lambda_1 - \lambda_n$ which deflates towards λ_n.

The deflation of single eigenvalue can be extended to r smallest eigenvalues by using the diagonal matrix Γ. Denote the matrix of r eigenvectors by $Z = [z_1, z_2, \cdots, z_r]$ and take an arbitrary matrix Y that satisfies the relation $Y^T Z = I$. The diagonal elements of Γ are $\gamma_1, \gamma_2, \cdots, \gamma_r$. The deflated matrix is then given by

$$\hat{A}_{r,\gamma} = \hat{A} - Z\Gamma_r Y^T \quad \text{where } Y^T Z = I. \tag{3.11}$$

Theorem 3.2. *The spectrum of the deflated matrix $\hat{A}_{r,\gamma}$ is given by*

$$\sigma(\hat{A}_{r,\gamma}) = \{\lambda_1 - \gamma_1, \lambda_2 - \gamma_2, \cdots, \lambda_r - \gamma_r,\ \lambda_{r+1}, \cdots, \lambda_n\}.$$

The proof is similar to that of single eigenvalue deflation. Thus, the first r smallest eigenvalues λ_i are deflated towards $\lambda_i - \gamma_i$ while the rest $n-r$ eigenvalues remains the same.

So far the matrix Y is undefined. Now take the matrix of r left eigenvectors as Y. Then, the r-component diagonalization can be obtained by

$$\hat{E} := Y^T \hat{A}\, Z,$$

where $\hat{E}$ is a diagonal matrix that contains the first r smallest eigenvalues $\lambda_1, \lambda_2, \cdots, \lambda_r$. Then, it leads to the following eigenvalue relations

$$\hat{A}\, Z = Z\, \hat{E} \quad \text{and} \quad Y^T \hat{A} = \hat{E}\, Y^T.$$

Suppose that all the r smallest eigenvalues are deflated towards the same value λ_d. Then, the diagonal matrix Γ_r is given by

$$\Gamma_r = \hat{E} - \lambda_d I.$$

Now equation (3.11) can be rewritten as

$$\begin{aligned}
\hat{A}_{r,\gamma} =& \hat{A} - Z(\hat{E} - \lambda_d I)Y^T \\
=& \hat{A} - Z\,\hat{E}\,Y^T + \lambda_d\, ZY^T \\
=& \hat{A} - Z\,\hat{E}\hat{E}^{-1}\hat{E}\,Y^T + \lambda_d\, Z\hat{E}\hat{E}^{-1}Y^T \\
=& \hat{A} - \hat{A}\,Z\hat{E}^{-1}Y^T\hat{A} + \lambda_d\, \hat{A}Z\hat{E}^{-1}Y^T \\
=& \hat{A}\,(I - Z\hat{E}^{-1}Y^T\,\hat{A} + \lambda_d\, Z\hat{E}^{-1}Y^T).
\end{aligned} \tag{3.12}$$

It is meaningful to assume that $\hat{A}$ is invertible so that $\lambda_i(\hat{A}) \neq 0$ for any i. Then, the inversion of $\hat{E}$ is applicable[2].

A slight variation in the last term after the third equality of derivation (3.12) leads to another version, which is given by

$$\begin{aligned}
\hat{A}_{r,\gamma} =& \dots \\
=& \hat{A} - Z\,\hat{E}\hat{E}^{-1}\hat{E}\,Y^T + \lambda_d\, Z\hat{E}^{-1}\hat{E}Y^T \\
=& \hat{A} - \hat{A}\,Z\hat{E}^{-1}Y^T\hat{A} + \lambda_d\, Z\hat{E}^{-1}Y^T\hat{A} \\
=& (I - Z\hat{E}^{-1}Y^T\,\hat{A} + \lambda_d\, Z\hat{E}^{-1}Y^T)\,\hat{A}.
\end{aligned} \tag{3.13}$$

The derivation (3.12) and (3.13) give rise to the definition of deflation operator.

Definition 3.1. *Take $Y = [y_1, y_2, \cdots, y_r]$ and $Z = [z_1, z_2, \cdots, z_r]$ as the eigenvectors of $\hat{A}$. The deflation operator is defined as*

$$\begin{cases} \text{left} & P := I - \hat{A}\,Z\hat{E}^{-1}Y^T + \lambda_d\, Z\hat{E}^{-1}Y^T, \\ \text{right} & Q := I - Z\hat{E}^{-1}Y^T\,\hat{A} + \lambda_d\, Z\hat{E}^{-1}Y^T, \end{cases} \quad \text{where } \hat{E} = Y^T\hat{A}Z.$$

Both the left and right deflation operators are derived from the same expression (3.11). So the right and left deflation preconditioned matrices are identical to each other and therefore have the same eigenvalues

$$P\hat{A} = \hat{A}Q \quad \text{and} \quad \lambda(\hat{A}Q) = \lambda(P\hat{A}).$$

When $\lambda_d = 0$, the first r smallest eigenvalues are deflated to zero. The deflation operator is reduced to

$$P_D = I - \hat{A}\,Z\hat{E}^{-1}Y^T \quad \text{and} \quad Q_D = I - Z\hat{E}^{-1}Y^T\,\hat{A}.$$

[2]In the discretization of Helmholtz problem, it is likely to have an A which has a zero eigenvalue and then $\hat{A}$ also has a zero eigenvalue. Such a defect can be easily remedied by taking a slight different grid. Then, the resulting discretization can skip the very point that leads to a zero eigenvalue.

When $\lambda_d = \lambda_n$, the first r smallest eigenvalues are deflated to the eigenvalue of largest magnitude, i.e. λ_n. The deflation operator is denoted by

$$P_N = I - (\hat{A} - \lambda_n I)Z\hat{E}^{-1}Y^T \quad \text{and} \quad Q_N = I - Z\hat{E}^{-1}Y^T(\hat{A} - \lambda_n I).$$

Figure 3.2 demonstrates the deflation preconditioning. As the deflation subspace gets larger, more small eigenvalues will be deflated while the rest eigenvalues still stay at the original positions. The spectral properties will be improved for the Krylov convergence.

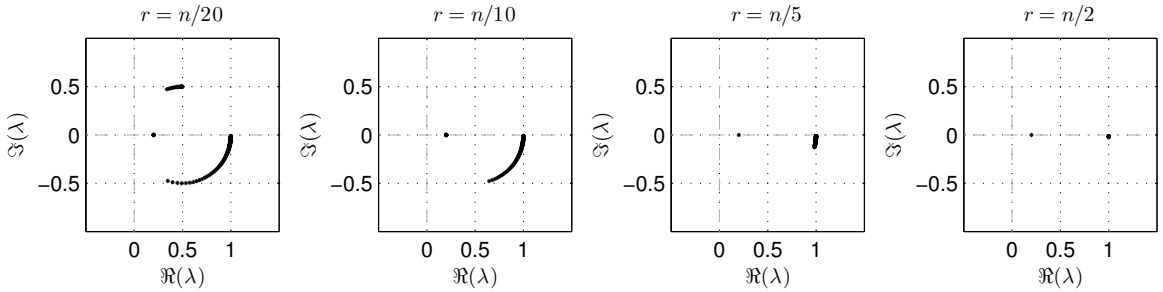


Figure 3.2: The spectrum distributions of the deflated matrix $\hat{A}Q$ towards $\lambda_d = 0.2$ when $k = 100$, shift $= 1 - \iota 1$

3.2.2 Generalized Deflation Subspace

The $n \times r$ rectangular matrices Y and Z forms the *deflation subspace*. In the previous subsection both Y and Z are chosen as the matrices that contain the eigenvectors of $\hat{A}$. The deflation subspace can be generalized by choosing any arbitrary $n \times r$ full rank matrices.

Now it is time to discuss some important properties of deflation preconditioning.

Projector In case of $\lambda_d = 0$, the deflation operator is actually a projector, which satisfies

$$P_D \cdot P_D = P_D \quad \text{and} \quad Q_D \cdot Q_D = Q_D.$$

The proof is done by direct matrix multiplication. However, the projection property is invalid when $\lambda_d \neq 0$. So it is rigorous to classify P and Q as deflation operator rather than a projector. Moreover, neither P nor Q is a full rank matrix.

Symmetry The preconditioned matrix $\hat{A}$ is symmetric as shown in (3.4). Provided $Y = Z$, the application of either P_D or Q_D will still preserve the symmetry of $\hat{A}$. When $\lambda_d \neq 0$, neither P nor Q can preserve the symmetry.

Eigen Relation In case of $\lambda_d = 0$, the left and right deflation preconditioning leads to the same result, i.e. $P_D\hat{A} = \hat{A}Q_D$. When $\lambda_d \neq 0$, the preconditioned matrices are not identical but the two preconditioned matrices still share the same spectrum.

Theorem 3.3. *Let $Y, Z \in \mathbb{R}^{n\times r}$ be any full rank matrices. Assume $\hat{A}$ is nonsingular and define $\hat{E} = Y^T\hat{A}Z$. Then,*

$$\sigma(P\hat{A}) = \sigma(\hat{A}Q).$$

Proof. For the sake of brevity, the notation $\overset{\sigma}{=}$ is used to state the equivalence in terms of eigenvalues, namely $\sigma(LHS) = \sigma(RHS)$.

For the left preconditioning, there is

$$\begin{aligned} P\hat{A} =& P_D\hat{A} + \lambda_d Z\hat{E}^{-1}Y^T\hat{A} = P_D\hat{A} + \lambda_d(I - Q_D) = P_D \cdot P_D\hat{A} + \lambda_d(I - Q_D) \\ =& P_D\hat{A}Q_D + \lambda_d(I - Q_D) = \lambda_d I + (P_D\hat{A} - \lambda_d I)\, Q_D \\ \overset{\sigma}{=}& \lambda_d I + Q_D\,(P_D\hat{A} - \lambda_d I) = Q_D P_D\hat{A} - \lambda_d Z\hat{E}^{-1}Y^T\hat{A} \\ =& (Q_D P_D - \lambda_d Z\hat{E}^{-1}Y^T)\,\hat{A}. \end{aligned}$$

For the right preconditioning, there is

$$\begin{aligned} \hat{A}Q =& \hat{A}Q_D + \lambda_d\hat{A}Z\hat{E}^{-1}Y^T = \hat{A}Q_D + \lambda_d(I - P_D) = \hat{A}Q_D \cdot Q_D + \lambda_d(I - P_D) \\ =& P_D\hat{A}Q_D + \lambda_d(I - P_D) = \lambda_d I + P_D\,(\hat{A}Q_D - \lambda_d I) \\ \overset{\sigma}{=}& \lambda_d I + (\hat{A}Q_D - \lambda_d I)\,P_D = \hat{A}Q_D P_D - \lambda_d\hat{A}Z\hat{E}^{-1}Y^T \\ =& \hat{A}\,(Q_D P_D - \lambda_d Z\hat{E}^{-1}Y^T). \end{aligned}$$

It is easily seen that $\hat{A}\,(Q_D P_D - \lambda_d Z\hat{E}^{-1}Y^T) \overset{\sigma}{=} (Q_D P_D - \lambda_d Z\hat{E}^{-1}Y^T)\,\hat{A}$, which leads to the eigen relation $\sigma(P\hat{A}) = \sigma(\hat{A}Q)$. □

Spectrum Once the deflation subspace is not the eigenvectors, the r smallest eigenvalues are still deflated towards the value λ_d but the rest $n - r$ eigenvalues will be modified.

Theorem 3.4. *Let $Y, Z \in \mathbb{R}^{n\times r}$ be any full rank matrices and define $\hat{E} = Y^T\hat{A}Z$. The spectrum of deflated matrix is given by*

$$\sigma(P\hat{A}) = \sigma(\hat{A}Q) = \{\lambda_d, \cdots, \lambda_d,\ \mu_{r+1}, \cdots, \mu_n\}. \tag{3.14}$$

Proof. Consider the left deflation preconditioning. The information of the first r eigenvalues is obtained by the following derivation.

$$\begin{aligned} P\hat{A}\,Z =& \hat{A}\,Z - \hat{A}Z\hat{E}^{-1}Y^T\hat{A}\,Z + \lambda_d Z\hat{E}^{-1}Y^T\hat{A}\,Z \\ =& \hat{A}\,Z - \hat{A}\,Z + \lambda_d\,Z \\ =& \lambda_d\,Z \end{aligned}$$

However, it is impractical to find out the rest $n - r$ eigenvalues since Y and Z are no longer the eigenvectors. The detailed knowledge of $\mu_{r+1}, \cdots, \mu_n$ depends on the exact choice of Y and Z.

The proof for $\sigma(\hat{A}Q)$ is easily done by the equivalent eigen relation shown in the previous paragraph. □

Condition Number In case of $\lambda_d \neq 0$ the deflation can change the lower bound of the spectrum to a larger value. So the condition number will be greatly improved. In case of $\lambda_d = 0$, the zero eigenvalues do not participate in the Krylov convergence [33] so the actual lower bound is still increased. Since P_D has preserved the symmetry of $\hat{A}$, the condition number can be given explicitly. Due to the zero eigenvalues, the *effective* condition number is introduced as

$$\kappa_{\text{eff}}(P_D\hat{A}) = |\mu_n|/|\mu_{r+1}|.$$

Since μ_{r+1} is relatively much larger than zero, the effective condition number $\kappa_{\text{eff}}(P_D\hat{A})$ is now much smaller and more favourable than $\kappa(\hat{A})$.

3.2.3 The Inaccuracy in the Deflation

In the practical implementation, the numerical computation will introduce rounding error into the inversion $\hat{E}^{-1}$ in the deflation operator. The inaccuracy in the deflation will deviate the theoretical preconditioning effect. The influence of the inaccuracy is investigated by using perturbation analysis.

Consider the case that the deflation subspace Y, Z consist of the left and right eigenvectors of $\hat{A}$. Then, $\hat{E} = Y^T \hat{A} Z$ is a diagonal matrix whose diagonal elements are the eigenvalues of $\hat{A}$. The inaccuracy of the inversion $\hat{E}^{-1}$ introduces the small perturbation $\epsilon_1, \cdots, \epsilon_r$ to the diagonal. The approximated inversion is given by

$$\hat{\mathbb{E}}^{-1} = diag(\frac{1-\epsilon_1}{\lambda_1}, \cdots, \frac{1-\epsilon_r}{\lambda_r}).$$

where $|\epsilon_i|_{i=1,\cdots,r} \ll 1$. Then, the eigenvalue computation of the deflated matrix with inaccuracy is given by

$$\begin{aligned}
\mathbb{P}\hat{A}\,Z =& \hat{A}\,Z - \hat{A}Z\hat{\mathbb{E}}^{-1}Y^T\hat{A}\,Z + \lambda_d Z\hat{\mathbb{E}}^{-1}Y^T\hat{A}\,Z \\
=& Z\hat{E} - Z\hat{E}\hat{\mathbb{E}}^{-1}\hat{E} + \lambda_d Z\hat{\mathbb{E}}^{-1}\hat{E} \\
=& Z\,(diag(\lambda_1, \cdots, \lambda_r) - diag(\lambda_1(1-\epsilon_1), \cdots, \lambda_r(1-\epsilon_r)) + \lambda_d diag(1-\epsilon_1, \cdots, 1-\epsilon_r)) \\
=& Z\,diag((1-\epsilon_1)\lambda_d + \lambda_1\epsilon_1, \cdots, (1-\epsilon_r)\lambda_d + \lambda_r\epsilon_r).
\end{aligned}$$

Due to the inaccuracy in the inversion, the first r eigenvalues are deflated towards the perturbed value $(1-\epsilon_i)\lambda_d + \lambda_i\epsilon_i$. However, it can be easily proved that the rest $n-r$ eigenvalues keep unchanged regardless of the inaccuracy. The spectrum of the $\mathbb{P}\hat{A}$ is given by

$$\sigma(\mathbb{P}\hat{A}) = \{(1-\epsilon_1)\lambda_d + \lambda_1\epsilon_1, \cdots, (1-\epsilon_r)\lambda_d + \lambda_r\epsilon_r,\ \lambda_{r+1}, \cdots, \lambda_n\}.$$

The spectrum of the deflated matrix with inaccuracy tells that it is insensible to choose $\lambda_d = 0$. Once the inversion is not exact, the eigenvalues would not be deflated towards zero but are actually towards some very small non-zero values. Then, the Krylov convergence will be slowed down by these very small eigenvalues close to zero.

In order to reduce the influence of inaccuracy to the minimum, the best choice is $\lambda_d = \lambda_n$, which brings the smallest relative deviation.

3.3 Multilevel Krylov multigrid method

The application of the shifted Laplacian preconditioner and the deflation operator has greatly improved the spectral properties of the Helmholtz equation. The deflated preconditioned system $AM^{-1}Q$ now has the favourable spectrum distribution and the resulting linear system is to be solved by the Krylov subspace method. The shifted Laplacian preconditioner is inverted approximately by the multigrid method. The deflation operator is computed approximately by the multilevel Krylov method [9].

The combination of the multigrid method and multilevel Krylov method gives d rise to the idea of multilevel Krylov multigrid method (MKMG), which was proposed in [10].

3.3.1 Flexible GMRES

MKMG computes both M^{-1} and Q in an implicit way. The deflated preconditioned system $AM^{-1}Q$ is never available explicitly. More important, the preconditioning is varying. Instead of GMRES with constant preconditioning, the flexible GMRES should be employed as Krylov solver.

The flexible GMRES was first introduced in [25]. The procedure remains the same in the construction of the orthonormal Krylov subspace as well as the solution of the least square

problem. The difference lies in the Krylov subspace where the solution is spanned. At every step, an additional vector is required to be stored

$$z_j := M_j^{-1} Q_j \, v_j, \tag{3.15}$$

where v_j is the j-th component in the Krylov basis for the system $AM^{-1}Q$ and z_j is that for the original matrix A. After the intermediate vector z_j is saved, the Krylov iterate is computed by

$$w_j := Az_j,$$

and then w_j is to be orthogonalized to the basis V_{j-1}. The reason why z_j requires the additional storage is that z_j forms the actual Krylov basis where the solution to A is spanned. The algorithm is given below.

Algorithm 4 Flexible GMRES with right preconditioning

1: Choose an initial guess x_0
2: Compute $r_0 = b - Ax_0$, $\beta = \|r_0\|_2$ and $v_1 := r_0/\beta$
3: **for** $j = 1, 2, \cdots$ until convergence **do**
4: Compute $z_j := M_j^{-1} Q_j v_j$
5: Compute $w_j := Av_j$
6: **for** $i = 1, 2, \cdots, j$ **do**
7: $h_{ij} := (w_j, v_i)$
8: $w_j := w_j - h_{ij} v_i$
9: **end for**
10: $h_{j+1,j} := \|w_j\|_2$
11: **if** $h_{j+1,j} \neq 0$ **then**
12: $v_{j+1} := w_j / h_{j+1,j}$
13: **else**
14: go to line 17
15: **end if**
16: **end for**
17: Set $m = j$
18: Build $Z_m = [z_1, z_2, \cdots, z_m]$ and $\bar{H}_m = \{h_{i,j}\}_{1 \leqslant i \leqslant m+1, 1 \leqslant j \leqslant m}$
19: Find $y =: \arg\min_y \|\beta e_1 - \bar{H}_m y\|_2$
20: Compute $x_m = x_0 + Z_m y$ as the resulting approximation.

The varying preconditioning For the time being, there are limited choices of Krylov solver for varying preconditioning. Besides GMRES, generalized conjugate residual method [26] and induced dimension reduction method [31] are the other Krylov solvers that can cope with varying preconditioning. The main obstacle is the difficulty in integrating varying preconditioning into the Krylov iteration.

3.3.2 Multilevel Krylov

Multilevel idea

The construction of deflation operator requires the inversion of a Galerkin matrix

$$\hat{E} := Y^T \hat{A} Z = Y^T A M^{-1} Z$$

in the coarser grid. For a small scale problem, the matrix size is small and the inversion can be computed exactly. However, in the large scale problem, the inversion has to be approximated in an implicit way.

In the right preconditioning step of Krylov method, the iterate can be written as

$$\begin{aligned} z :=& M^{-1}Qv \\ =& M^{-1}(I + Z\hat{E}^{-1}Y^T(\lambda_n I - \hat{A}))v \\ =& M^{-1}(v + Z\hat{E}^{-1}v_r), \end{aligned}$$

where

$$v_r := Y^T(\lambda_n I - \hat{A})v = Y^T(\lambda_n I - AM^{-1})v.$$

Denoting $\hat{E}^{-1}v_r$ by v', now approximating the inversion amounts to solving the Galerkin system

$$\hat{E}v' = v_r. \tag{3.16}$$

Because $\hat{E}$ results from the Galerkin projection, it will inherit the unfavoured spectral property from the matrix $\hat{A}$ with whom $\hat{E}$ is associated. In order to accelerate the Krylov convergence, the Galerkin system (3.16) is also preconditioned by the deflation operator. The construction of Q for (3.16) requires the inversion of a smaller Galerkin matrix $Y^T\hat{E}Z$. If it is sufficiently small, then $(Y^T\hat{E}Z)^{-1}$ is computed exactly. Otherwise, it forms a new Galerkin system which is again solved by Krylov method with deflation preconditioning.

The process will not stop until the Galerkin matrix in the nested deflation operator is sufficiently small for an exact inversion. Such a recursive application of Krylov method forms the multilevel Krylov method.

Approximate Galerkin projection

During the iteration, M^{-1} is always implicitly computed since it is approximated by the multigrid method. So the preconditioned matrix $\hat{A}$ is never explicitly available. By multiplying Y^T and Z, the consecutive projections onto the lower levels lead to a Galerkin matrix on the k-th level

$$\hat{E}_k := Y^T_{(k-1,k)} \cdots Y^T_{(1,2)}\, \hat{A}\, Z_{(1,2)} \cdots Z_{(k-1,k)}.$$

$\hat{E}_k$ will be multiplied by the Krylov iterate during the iteration. After $k-1$ matrix-vector multiplications, the iterate vector will have to multiply the M^{-1} in the $\hat{A}$ on the first level. So the multigrid iteration for M^{-1} in the finest grid should be conducted in order to accomplish the Krylov iteration on the k-th level. The system size on the k-th level is much smaller than that on the first level. It is not economical to perform in this way.

In [9], an approximation of $\hat{E}$ is proposed so that the inversion can takes place on an M of the same size that matches the k-th level. The approximation is based on the replacement of M^{-1} with $Z(Y^TMZ)^{-1}Y^T$, which leads to

$$\hat{E} := Y^T\hat{A}Z = Y^TAM^{-1}Z \approx \underbrace{Y^TAZ}_{A_{(2)}}(\underbrace{Y^TMZ}_{M_{(2)}})^{-1}\underbrace{Y^TZ}_{B_{(2)}}.$$

Now, the formulation of $\hat{E}$ is conducive to the application of multilevel Krylov method. With this approximation, the Galerkin system (3.16) is rewritten as

$$\hat{A}_{(2)}v' = v_r \quad \text{where } \hat{A}_{(2)} = A_{(2)}M^{-1}_{(2)}B_{(2)}.$$

The solution v' is approximated by the Krylov method. In order to accelerate the convergence,

the Galerkin system $\hat{A}_{(2)}$ is again preconditioned by a deflation operator

$$\begin{aligned} Q_{(2)} :=& I + Z_{(2,3)}\hat{E}_{(3)}^{-1}Y_{(2,3)}^T(\lambda_n I - \hat{A}_{(2)}) \\ \text{where } \hat{E}_{(3)} :=& Y_{(2,3)}^T\hat{A}_{(2)}Z_{(2,3)} \\ \approx& \underbrace{Y_{(2,3)}^T A_{(2)} Z_{(2,3)}}_{A_{(3)}}(\underbrace{Y_{(2,3)}^T M_{(2)} Z_{(2,3)}}_{M_{(3)}})^{-1}\underbrace{Y_{(2,3)}^T Z_{(2,3)}}_{B_{(3)}}. \end{aligned}$$

If $\hat{E}_{(3)}$ is not sufficiently small for an exact inversion, then its inversion is approximated by the Krylov method with preconditioning by $Q_{(3)}$.

In case of an m-level grid, the consecutive process will not stop until the Krylov iteration has arrived the $m-1$-th level, where the $\hat{E}_{(m)}$ in $Q_{(m-1)}$ is sufficiently small for an exact inversion on the m-th level. A detailed algorithm for the multilevel Krylov method is given below.

Algorithm 5 Multilevel Krylov method with approximate Galerkin projection

1: Initialization Phrase
2: Construct $A_{(k)}$, $M_{(k)}$ and $B_{(k)}$.
3: Choose $A_{(1)} := A$, $M_{(1)} := M$ and $B_{(k)} := I$.
4: **for** $k = 2, \cdots, m$ **do**
5: compute $A_{(k)} := Y_{(k-1,k)}^T A_{(k-1)} Z_{(k-1,k)}$
6: compute $M_{(k)} := Y_{(k-1,k)}^T M_{(k-1)} Z_{(k-1,k)}$
7: compute $B_{(k)} := Y_{(k-1,k)}^T B_{(k-1)} Z_{(k-1,k)}$
8: define $\hat{A}_{(k)} := A_{(k)} M_{(k)}^{-1} B_{(k)}$
9: define $Q_{(k)} := I + Z_{(k,k+1)}\hat{A}_{(k+1)}^{-1} Y_{(k,k+1)}^T(\lambda_n I - \hat{A}_{(k)})$
10: **end for**
11:
12: Iteration Phrase
13: Apply flexible GMRES to $\hat{A}_{(k)}Q_{(k)}\, x_{(k)} = b_{(k)}$.
14: Compute $v_r := Y_{(k,k+1)}^T(\lambda_n I - \hat{A}_{(k)})\, v_{(k)}$, where $M_{(k)}^{-1}$ is approximated by the multigrid iteration.
15: **if** $k+1 = m$ **then**
16: Invert $\hat{A}_{(k+1)}$ exactly by a direct method.
17: Set $v'_{(k)} := \hat{A}_{(k+1)}^{-1} v_r$.
18: **else**
19: Set $k = k+1$.
20: Set $b_{(k)} := v_r$. {The k is the result of the updating on line 19.}
21: Go to line 13.
22: **end if**
23: Compute $w_{(k)} := A_{(k)} M_{(k)}^{-1} B_{(k)}\,(v_{(k)} - Z_{(k,k+1)} v'_{(k)})$, where $M_{(k)}^{-1}$ is approximated by the multigrid iteration.
24: The matrix-vector multiplication is completed and set of $w_{(k)}$'s form the Krylov subspace.

25: The approximation $\tilde{x}_{(k)}$ is spanned by the Krylov subspace.
26: Set $v'_{(k)} = \tilde{x}_{(k)}$.

Remark On the first level, the Krylov subspace Z_m in the flexible GMRES spans the solution to the original system $Ax = b$ and the vector $z_j := M_j^{-1} Q_j\, v_j$ is saved. On the k-th

level for $2 \leqslant k \leqslant m-1$, the Krylov subspace Z_m contains the vectors $z_j := Q_{j(k)}\, v_j$, which span the solution to the Galerkin system $A_{(k)} M_{(k)}^{-1} B_{(k)}\, v'_{(k)} = v_{r(k)}$.

3.3.3 MKMG

The multilevel Krylov method shares a similar idea with the multigrid method in simplifying the computation difficulty. Both of them project the original problem from the finest grid to the coarsest grid level by level so that an matrix in the coarsest grid will be inverted exactly on the lowest level. Such an idea makes a connection between the two kinds of iteration in the numerical implementation.

Due to the definition of the deflation subspace, Y and Z can be chosen arbitrarily provided they are full rank matrices. So it is natural to consider the intergrid operators in the multigrid method as the candidates for the deflation subspace[10]. In this case $\hat{A}_{(k)}$ and $\hat{M}_{(k)}$ in the multilevel Krylov method become the coarse grid operators.

Now both multilevel Krylov method and multigrid method shares the same $M_{(k)}$ in the iteration. This implementation not only saves the storage but also reduces the computation.

Figure 3.3 demonstrates how the multilevel Krylov method collaborates with the multigrid method. In order to simplify the drawing, the illustration shows the implementation where only one iteration on each level is conducted for both methods. Generally, in practical implementation, several iterations are conducted for each method.

In the illustration, the multilevel Krylov method, denoted by •, is conducted from level 1 to $m-1$. On the m-th level, the Galerkin system is solved exactly by a direct method. The multigrid method uses V-cycle, which can be replaced with any other type of iteration cycle. The pre/post smoothing is denoted by ∘. On the j-th level the multigrid method approximates the inversion of $M_{(k)}$ in a $(m-j+1)$-level grid.

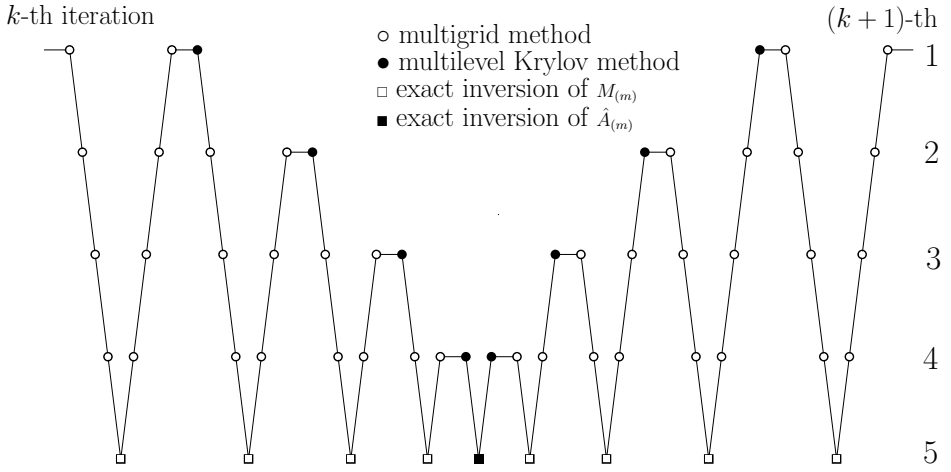


Figure 3.3: The illustration of multilevel Krylov multigrid method in a five-level grid

Due to the right preconditioning $\hat{A}_{(k)} Q_{(k)}$, the multigrid methods on the left part of the figure refer to the approximation of $M_{(k)}^{-1}$ for the construction of the deflation operator $\hat{A}_{(k)} \mathbf{Q_{(k)}}$ (see line 14 in algorithm 5). Those on the right part approximate $M_{(k)}^{-1}$ for preconditioning the Galerkin system $\hat{\mathbf{A}}_{\mathbf{(k)}} Q_{(k)} = \mathbf{A_{(k)}} \mathbf{M_{(k)}^{-1}} \mathbf{B_{(k)}} Q_{(k)}$ (see line 23 in algorithm 5). In case of the left preconditioning, the sequence is inverted.

Chapter 4

Fourier Analysis

In the iterative methods the spectrum of the operator matrix provides important information of the convergence. For stationary iterative methods and the multigrid method, the spectral radius of the operator matrices determines whether the method is convergent and how fast it can converge. Besides that, the distribution of the whole spectrum is of great significance to the convergence of Krylov subspace methods. So it is valuable to know the precise information of the spectrum, which predicts the convergence behaviour of the adopted iterative method.

However, it is more difficult to solve the eigenvalue problem of a large matrix than the system itself. The chances are that the eigenvalues of some particular problems can be obtained in an analytical way. Because the operators in these problems consist of regular components whose spectral properties are already known.

In the model Helmholtz problem on page 4 the main operators are linear transformations of the discrete Laplacian operator Δ_h. All of them share the same set of eigenvectors and their eigenvalues can be easily obtained.

With the help of this basic knowledge, it now becomes feasible to apply the eigenvalue analysis to the iterative methods for the Helmholtz problem. A powerful tool for the eigenvalue analysis is Fourier analysis which is able to find out the quantitative information of the spectrum.

4.1 Principles of Fourier Analysis

In this section the theory of Fourier analysis will be introduced. The introduction is based on the related content in [32].

4.1.1 Invariance Subspace

The fundamental idea of Fourier analysis is based on the fact that a certain space E is invariant under the discrete operator K. With respect to the invariance space, the discrete operator can be represented by a block diagonal matrix. The collection of the eigenvalues of each block matrix are equivalent to the spectrum of the discrete operator.

Assume K is an $n \times n$ matrix and the invariance space E is spanned by the column vectors of an $n \times m$ full rank matrix $\Phi = [\phi_1, \phi_2, \cdots, \phi_m]$. The invariance can be demonstrated by

$$K E \subset E \Longrightarrow K\,\Phi = \Phi\,\tilde{K}. \tag{4.1}$$

The $m \times m$ matrix $\tilde{K}$ is called the representation.

The representation reveals the information of how the operator acts in the invariance space. The vectors in E can be expressed as $v = \Phi c$ where c is an $m \times 1$ vector. The action of K on v actually transforms c into $\tilde{c}$ by

$$K\,v = K\,\Phi c = \Phi \tilde{K}\,c = \Phi\,\tilde{c}, \quad \text{where } \tilde{c} = \tilde{K} c.$$

The effect of K on v is represented by $\tilde{K}$. More basically, $\tilde{K}$ measures the effect of K on c in the space E. So the analysis on $\tilde{K}$ gives equivalent knowledge about the behaviour of K.

If a big space E is the union of several disjoint subspace E^l all of which are invariant under K, then the action of K on E can be represented by a block diagonal matrix as

$$K \overset{\wedge}{:=} [\tilde{K}^l] \quad \text{with } l \text{ as the block index.}$$

Each $\tilde{K}^l$ corresponds to the representation of K with respect to each subspace E^l. The analysis of the action of K on E is now the collection of the analysis on each $\tilde{K}^l$.

The size of each block matrix is equivalent to the dimension of the invariance space. So it is important to keep the subspace small so that the analysis of each block representation can be handled in an easy way, either theoretically or numerically.

In case the invariance subspace happens to be the eigenspace of the operator, then the analysis is simplified since the block representation is diagonal. All the basic operators in the following sections possess such a property.

4.1.2 Fourier Analysis for Multigrid Components

In this section the computation of the representation is conducted for each component in the multigrid method. Without loss of generality, the smoother employs the weighted Jacobi iteration while the restriction and prolongation operator are the full weighting operator and linear interpolation respectively.

Assume the system size is $(n-1) \times (n-1)$ where n is an even number. The fine grid on the first level is denoted by

$$\Omega_h := \{x_i = ih \text{ for } i = 1, 2, \cdots, n-1\},$$

where the amount of subintervals n is supposed to be an even integer. Similarly, the definition of the coarse grid on the second level is given by

$$\Omega_{2h} := \{x_i = i2h \text{ for } i = 1, 2, \cdots, \frac{n}{2} - 1\}.$$

The two-grid iteration operator is given by

$$T_1^2 = S_1^{\nu_2} K_1^2 S_1^{\nu_1} \quad \text{with } K_1^2 = I - P_2^1 M_2^{-1} R_1^2 M_1. \tag{4.2}$$

Here, K_1^2 is the coarse grid correction operator. S_1 is the fine grid smoothing operator with the pre- and post- smoothing steps ν_1 and ν_2. R_1^2 is the restriction operator and P_2^1 is the prolongation operator. M is the shifted Laplacian preconditioner

$$M_h = -\Delta_h - (\beta_1 + \iota\beta_2)k^2 I.$$

Space of harmonics In the 1D Helmholtz problem a two-dimensional space of harmonics is chosen as the invariance space

$$E_h^l := \begin{cases} \text{span}\{\varphi_h^l, \varphi_h^{n-l}\} & \text{for } l = 1, 2, \cdots, \frac{n}{2} - 1; \\ \text{span}\{\varphi_h^l\} & \text{for } l = \frac{n}{2}. \end{cases}$$

The space basis φ_h^l is the eigenvector of the discrete Laplacian $-\Delta_h$ on the fine grid Ω_h. Each pair of the bases $[\varphi_h^l, \varphi_h^{n-l}]$ coincides in the coarse grid Ω_{2h}, which means

$$\varphi_{2h}^l = \varphi_h^l = -\varphi_h^{n-l} \quad \text{on } \Omega_{2h}.$$

Invariance under M As M is the shifted Laplacian, φ_h^l is also the eigenvector of M and the invariance property follows as

$$M : E_h^l \to E_h^l \quad \text{for } l = 1, 2, \cdots, \frac{n}{2} - 1,$$

which leads to the diagonalized representations

$$\tilde{M}_1^l := \begin{bmatrix} \frac{4}{h^2}\sin^2(lh\pi/2) - (\beta_1 + \iota\beta_2)k^2 & 0 \\ 0 & \frac{4}{h^2}\cos^2(lh\pi/2) - (\beta_1 + \iota\beta_2)k^2 \end{bmatrix}$$

For $l = n/2$, the invariance holds with respect to $E_h^{n/2} = \text{span}\{\varphi_h^{n/2}\}$ and the above representation degenerates into a 1×1 matrix

$$\tilde{M}_1^{n/2} := \frac{2}{h^2} - (\beta_1 + \iota\beta_2)k^2. \tag{4.3}$$

Because M_2 is inverted exactly on the coarse grid. The representation of M_2 with respect to E_2^l is just a single eigenvalue of M_2 as

$$\tilde{M}_2^l := \frac{4}{(2h)^2}\sin^2(l2h\pi/2) - (\beta_1 + \iota\beta_2)k^2$$

On the coarse grid E_{2h}^l is invariant under M_2^{-1} and its representation is given by

$$\tilde{M}_2^{-1\,l} := \left(\frac{4}{H^2}\sin^2(lH\pi/2) - (\beta_1 + \iota\beta_2)k^2\right) \text{ where } H = 2h. \tag{4.4}$$

Representation of the Smoother Using Jacobi iteration the smoothing operator is given by

$$S = (1-\omega)I + \omega(I - D^{-1}M).$$

The subspace E_1^l is also invariant under S with the representation

$$\tilde{S}_1^l := \begin{bmatrix} (1-\omega) + \omega\frac{2\cos(lh\pi)}{2-(\beta_1+\iota\beta_2)h^2k^2} & 0 \\ 0 & (1-\omega) - \omega\frac{2\cos(lh\pi)}{2-(\beta_1+\iota\beta_2)h^2k^2} \end{bmatrix} \tag{4.5}$$

Intergrid operators R_1^2 and P_2^1, as the intergrid operators in multigrid method, do not possess the invariance property but have the following mapping relation which bridges between the fine and coarse grids. For $l = 1, 2, \cdots, n/2 - 1$, there is

$$R_1^2 : E_h^l \to \text{span}\{\varphi_{2h}^l\},$$
$$P_2^1 : \text{span}\{\varphi_{2h}^l\} \to E_h^l.$$

The representations of R_1^2 with respect to E_{2h}^l is given by

$$\tilde{R}_1^{2\,l} := [\cos^2(lh\pi/2), -\sin^2(lh\pi/2)]. \tag{4.6}$$

Similarly, the representation of P_2^1 with respect to E_h^l is given by

$$\tilde{P}_2^{1\,l} := \begin{bmatrix} \cos^2(lh\pi/2) \\ -\sin^2(lh\pi/2) \end{bmatrix}. \tag{4.7}$$

For $l = n/2$, no representation of R_1^2 and P_2^1 exists. The $n/2$-th mode does not exist on Ω_{2h} and the restriction from Ω_h to Ω_{2h} can only lead to a trivial result $R_1^2\varphi_l^{n/2} = 0$. The prolongation from Ω_{2h} to Ω_h cannot excite the $n/2$-th mode, either.

4.1.3 Two-grid Analysis

The representation of the two-grid iteration operator $\tilde{T}_1^2$ can be obtained by substituting the representations (4.3),(4.4),(4.5),(4.6) and (4.7) into equation (4.2). For $l = 1, 2, \cdots, n/2-1$, there is

$$\tilde{T}_1^2 := \tilde{S}_1^{\nu_2}\, \tilde{K}_1^2\, \tilde{S}_1^{\nu_1} \quad \text{with } \tilde{K}_1^2 := (I - \tilde{P}_2^1 \tilde{M}_2^{-1} \tilde{R}_1^2 \tilde{M}_1). \tag{4.8}$$

The intact mode The mode $n/2$ is excluded by the coarse grid correction and $\tilde{M}_2^{-1^{n/2}}$ does not exist on Ω_{2h}. The computation of $\tilde{K}^{n/2}$ is validated by direct multiplication without the presence of $\tilde{M}_2^{-1^{n/2}}$ as

$$\begin{aligned}\varphi_h^{n/2}\, \tilde{K}^{n/2} :=& (1 - P_2^1 R_1^2 M_1)\varphi_h^{n/2} \\ =& \varphi_h^{n/2} - P_2^1 R_1^2 \varphi_h^{n/2} \tilde{M}_1^{n/2} \\ =& \varphi_h^{n/2}\end{aligned}$$

The last equality follows the fact that $R_1^2 \varphi_h^{n/2} = 0$. So the result is $\tilde{K}^{n/2} = 1$ with respect to the subspace $E^{n/2} = \text{span}\{\varphi_h^{n/2}\}$. The $n/2$-th component is kept intact under the coarse grid operator. So the representation of the two-grid iteration operator for the $n/2$-th mode is

$$\tilde{T}_1^2 := \tilde{S}_1^{\nu_2}\, \tilde{S}_1^{\nu_1}. \tag{4.9}$$

The equation (4.9) presents the fact that the two-grid iteration operator has no coarse grid correction effect on the $n/2$-th mode but only the smoothing effect. The $n/2$-th error component is only smoothed but not corrected on a coarser grid.

The above interpretation can be easily validated by the numerical experimentation. The calculation of $|K_1^2, \varphi_h^{n/2} - \varphi_h^{n/2}|$ only shows machine error.

4.1.4 Multigrid Analysis

On a three-level grid, the coarse grid operator M_2 on the second level is not inverted exactly but the exact inversion M_2^{-1} is replaced by another two-grid approximation

$$(I - T_2^3) M_2^{-1},$$

where T_2^3 is the two-grid iteration operator between Ω_{2h} and Ω_{4h}. M_3 is inverted exactly in T_2^3 the coarse grid operator on the third level.

The three-grid analysis can be extended to an m-grid analysis by using the recursive expression. Including the smoothing operator, the multigrid iteration operator T_1^m is given by

$$\tilde{T}_k^m = \tilde{S}_k^{\nu_2}(I - \tilde{P}_{k+1}^k\, (I - \tilde{T}_{k+1}^m) \tilde{M}_{k+1}^{-1}\, \tilde{R}_k^{k+1} \tilde{M}_k) \tilde{S}_k^{\nu_1} \quad \text{with} \tilde{T}_m^m = 0, \tag{4.10}$$

for $k = 1, 2, \cdots, m-1$.

The above expression cannot cover all the modes in the finest grid on the first level. Because on every level except the coarsest one, there is always one intact mode that is not included in the coarse grid correction. On the k-th level the $n/2^k$-th mode is the intact mode. It is not activated by the components in the coarser grid on the $(k+1)$-th level but this mode should be existent to activate the corresponding two modes in the finer grid on the $(k-1)$-th level and then their following modes. For the $n/2^k$-th mode on the k-th level, the approximation to the inversion M_k^{-1} is given without the presence of T_{k+1}^m by

$$(I - S_k^{\nu_2} S_k^{\nu_1}) M_k^{-1}.$$

Then, the recursive computation of the representations excited by the $n/2^k$-th mode on the k-th level can be given by

$$\tilde{T}_i^m = \tilde{S}_i^{\nu_2}(I - \tilde{P}_{i+1}^i\,(I - \tilde{T}_{i+1}^m)\tilde{M}_{i+1}^{-1}\,\tilde{R}_i^{i+1}\tilde{M}_i)\tilde{S}_i^{\nu_1} \quad \text{with } \tilde{T}_k^m = S_k^{\nu_2}S_k^{\nu_1},$$

for $i = 1, \cdots, k-1$. In the finest grid on the first level there will be 2^{k-1} representations resulted from the $n/2^k$-mode on the k-th grid.

In an m-grid analysis every mode in the coarsest grid on the m-the level will finally activate 2^{m-1} modes in the finest grid on the first level. The resulting representation $\tilde{T}$ then contains $(n/2^{m-1} - 1)$ blocks matrices whose sizes are all $2^{m-1} \times 2^{m-1}$. Besides that, due to the intact mode, there will be $(m-1)$ block matrices whose size are respectively $2^{k-1} \times 2^{k-1}$ for $k = 1, \cdots, m-1$.

4.1.5 The Application to Preconditioning

Both the shifted Laplacian preconditioner and deflation preconditioner are made of the multigrid components. So the spectral properties of the preconditioning can be investigated by computing the representations of each preconditioner. The representation of the shifted Laplacian preconditioner is given by

$$\tilde{\hat{A}} = \tilde{A}\tilde{M}^{-1}$$

and that of the deflation operator is given by

$$\tilde{Q} = I - \tilde{P}_2^1\tilde{\hat{E}}_2\tilde{R}_2^1(\lambda_n I - \tilde{\hat{A}}) \text{ with } \tilde{\hat{E}}_2 = \tilde{R}_1^2\tilde{\hat{A}}\tilde{P}_2^1$$

Without Fourier analysis, the eigenvalues have to be computed by using the MATLAB built-in function eig. In this case, the computation takes place on the entire matrix. In Fourier analysis, the eig computation is just applied to the very small representation matrices. The advantage of Fourier analysis over the direct application of eig can be concluded in three aspects.

Computation time It is obvious that the Fourier analysis method is much cheaper, as demonstrated in Figure 4.1. Directly using eig, the computation time increases largely as the increase in wavenumber raises the system size. In contrast, Fourier analysis just has a tiny increment in the computation time.

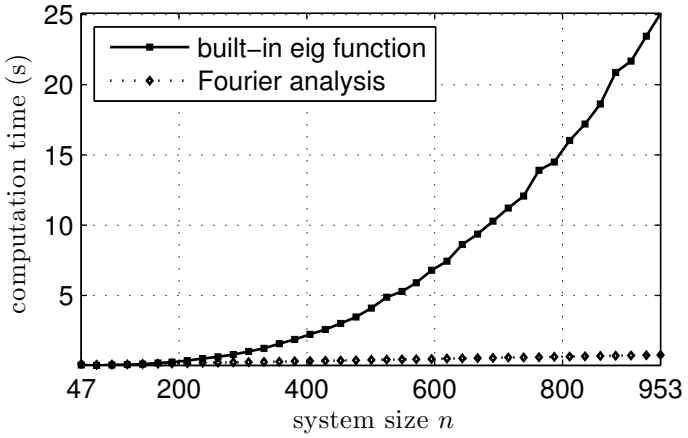


Figure 4.1: Comparison of the CPU time for the eigenvalue computation of $AM^{-1}Q$. The wavenumber k varies from 10 to 200.

Memory requirement In order to apply eig in MATLAB, the matrix should be stored as a full matrix. The increase in system size will have a heavy demand on memory. The memory limit will prevent studying the large problem. But Fourier analysis only requires the storage of the very small block matrices, which is much less demanding. So Fourier analysis extends the research to the large problem of high wavenumber and fine wave resolution, especially in 2D and 3D cases.

Accuracy The computation accuracy of Fourier analysis will not decrease as the system size goes up while the eig method has to suffer from the decreasing accuracy[1]. As the system size increases, the size of Fourier representation keeps the same except that the amount of these small block matrices increases. So the accuracy of computing every block matrix is unchanged. However, the accuracy of directly using eig will decrease. Because the computation is applied to the entire matrix whose size is increasing.

4.2 Analysis of the Preconditioning

The application of Fourier analysis makes it more convenient to investigate the spectral properties of the preconditioned system, especially when the wavenumber is very large.

The spectrum study starts from the case where all the inversions are computed in an exact way. Without special mention, the analysis is applied to the Helmholtz problem of $k = 100$. The wave resolution that denotes the amount of subintervals per wavelength is set as $gw = 30$. The default shift is $1 + 1\iota$ and the eigenvalues are deflated towards one. These parameters will change respectively in accordance to the research interest.

4.2.1 The Preconditioning Effect

Figure 4.2 demonstrates the effect of shifted Laplacian preconditioner with respect to various wavenumbers. The preconditioning has greatly reduced the high indefiniteness of the unpreconditioned system. The spectrum is now restricted to a unit circle. Preferably, the eigenvalues are more densely clustered around $(1, 0)$ and $(0.5, 0.5)$ than around the origin $(0, 0)$, which can effectively accelerate the Krylov convergence.

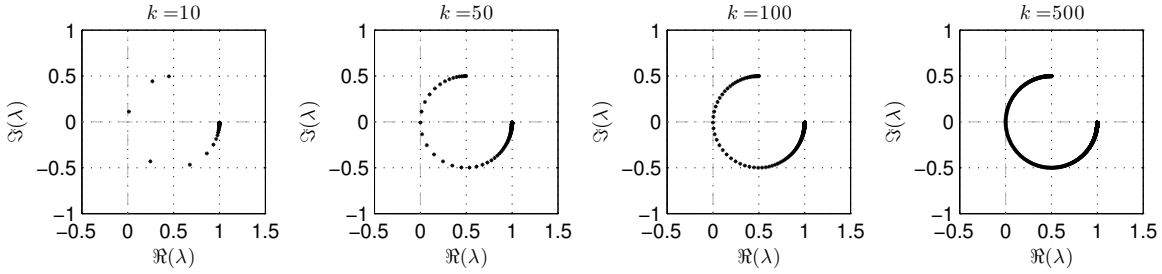


Figure 4.2: The spectrum distributions of the preconditioned system AM^{-1}

However, it is also clear that the increase in wavenumber will raise the amount of small eigenvalues close to zero. This unfavourable property will slow down the Krylov convergence in the case of a high wavenumber. The deflation technique is employed to remedy this defect, as shown in Figure 4.3. All the eigenvalues have been deflated to a very small and slim region. In fact, they are now located on a short arc of the unit circle. The spectral properties of the deflated system become favourable to the Krylov convergence.

[1]It is difficult to present a demonstration in this aspect. Because the eigenvalues of AM^{-1} and $AM^{-1}Q$ are complex and cannot be sorted in an ordered sequence. So the comparison of the results by two methods cannot be made

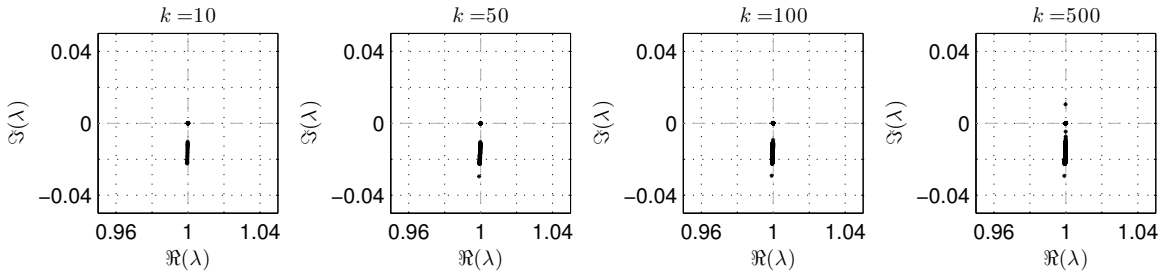


Figure 4.3: The spectrum distributions of the deflation preconditioned system $AM^{-1}Q$

4.2.2 The Choice of the Shift

The shift $\beta_1 + \iota\beta_2$ plays a decisive role in the shifted Laplacian preconditioner M. It is self-evident that β_1 cannot be chosen larger than one. Otherwise, the preconditioner would be even more indefinite than the original Helmholtz equation. On the other hand, if β_2 is chosen smaller than one or even negative, it will weaken the preconditioning effect for Krylov convergence since the preconditioner M becomes deviated from the original Helmholtz equation A. Generally, $\beta_1 = 1$ is a suitable choice, which makes the β_2 responsible for reducing the indefiniteness.

Figure 4.4 shows the spectrum distributions due to different β_2. The magnitude of β_2 determines the length of the arc on which the eigenvalues are located. A large magnitude β_2 will not only locate the eigenvalues on a shorter arc but also have more eigenvalues clustered around the origin. A smaller magnitude β_2 can locate fewer eigenvalues around the origin. Since the preconditioner of smaller β_2 is closer to the original matrix A, the product AM^{-1} is closer to the identity I, which results in more eigenvalues around one.

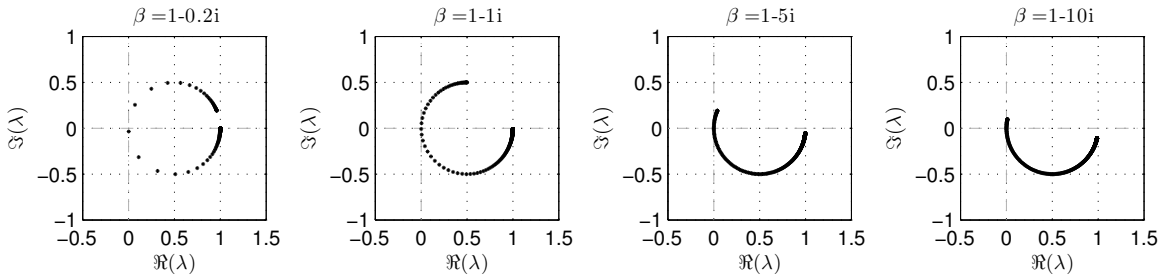


Figure 4.4: The influence of β_2 on the preconditioned system AM^{-1}

Figure 4.5 shows the influence of β_2 on the deflation preconditioned system $AM^{-1}Q$. A smaller β_2 leads to a more tightly clustered distribution around $(1, 0)$. It can be concluded that a smaller β_2 is more favourable for the Krylov convergence of both AM^{-1} and $AM^{-1}Q$. However, on the other side, a smaller β_2 makes the preconditioner M more similar to the original matrix A, which leads to a harder solution of the inversion M^{-1}.

It can be shown that the sign of β_2 only determines the sign of eigenvalues but has no influence on the spectral properties. Since A is real and $M_{(\beta_2)} = \overline{M}_{(-\beta_2)}$, the following relation holds

$$AM^{-1}_{(-\beta_2)} = \bar{A}\bar{M}^{-1}_{(\beta_2)} = \overline{AM}^{-1}_{(\beta_2)},$$

which is followed by the eigenvalue equality

$$\lambda(AM^{-1}_{(-\beta_2)}) = \overline{\lambda}(AM^{-1}_{(\beta_2)}).$$

So both β_2 and $-\beta_2$ result in the eigenvalues of the same modulus and therefore the same preconditioning effect.

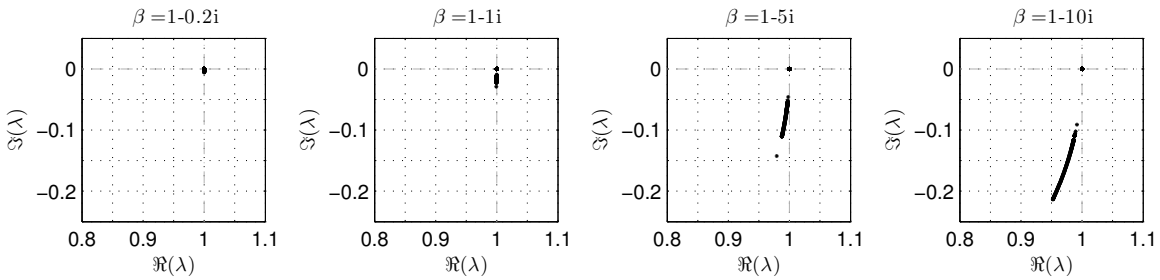


Figure 4.5: The influence of β_2 on the deflation preconditioned system $AM^{-1}Q$

4.2.3 The Influence of Wave Resolution

The system size is determined by both the wavenumber and the wave resolution gw. The influence of the wave resolution on the spectrum distribution is shown in Figure 4.6. It is preferable that the amount of the small eigenvalues close to the origin is not raised by the increase in the wave resolution. Then, the Krylov convergence will not be slowed down when the wave is better resolved.

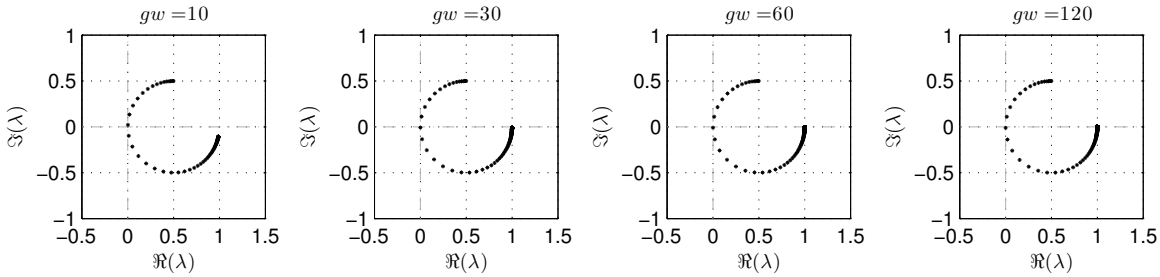


Figure 4.6: The influence of wave resolution gw on the preconditioned system AM^{-1} when $k = 50$

For the deflation preconditioned system $AM^{-1}Q$, the increase in wave resolution will make the spectrum distribution more tightly clustered around $(1, 0)$, except for the small system of $gw = 10$. If the Krylov solver is applied to $AM^{-1}Q$, the convergence will not be slowed down but be accelerated. The system $AM^{-1}Q$ contains only favourable eigenvalues around $(1, 0)$, the convergence will not suffer from the negative influence of the small eigenvalues around the origin. In a large system due to the increase in wave resolution, there are more favourable eigenvalues which will make convergence more rapidly.

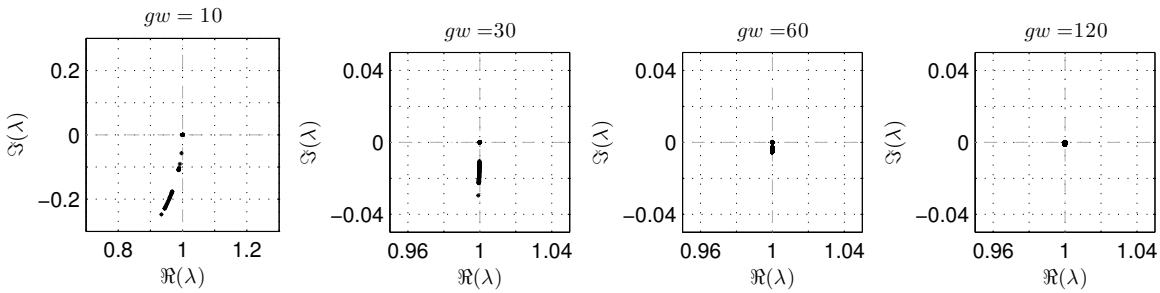


Figure 4.7: The influence of wave resolution gw on the deflation preconditioned system AMQ^{-1} when $k = 50$.

Please notice the axis scaling of the first subplot is different from others.

4.2.4 Different influence of k and gw

The system size will be raised by the increase in either wavenumber k or wave resolution gw. However, the two variables have distinct behaviour in the eigenvalue computation of AM^{-1}, which is given by

$$\lambda_l(AM^{-1}) = \frac{\frac{1}{h^2}4\sin^2(l\pi h/2) - k^2}{\frac{1}{h^2}4\sin^2(l\pi h/2) - k^2(\beta_1 + \iota\beta_2)}.$$

Using the relation $h = \frac{2\pi}{k}/gw = \frac{2\pi}{k\cdot gw}$, the above formula is transformed into

$$\lambda_l(AM^{-1}) = \frac{\left(\frac{gw}{\pi}\sin(\frac{l\pi^2}{k\cdot gw})\right)^2 - 1}{\left(\frac{gw}{\pi}\sin(\frac{l\pi^2}{k\cdot gw})\right)^2 - (\beta_1 + \iota\beta_2)}.$$

Denoting $\frac{gw}{\pi}\sin(\frac{l\pi^2}{k\cdot gw})$ by $x(l)$, the eigenvalue computation can be decomposed into to two parts

$$x(l) = \frac{gw}{\pi}\sin(\frac{l\pi^2}{k\cdot gw}) \quad \text{and} \quad f(x) = |\lambda_l| = \left|\frac{x^2-1}{x^2-(\beta_1+\iota\beta_2)}\right|.$$

In case of a fixed gw and an increasing k, the range[2] of function $x(l)$ keeps the same. The increasing system size n leads to more discrete values within this range. When k is fixed and gw is increasing, the range of $x(l)$ is amplified by the gw/π in front of the $\sin(\cdots)$. Although the amount of discrete values also increases, it is much less crowded in the range than the case of enlarged k with unchanged gw. Within a specific small range, the increase in gw slightly changes the amount of discrete values but the increase in k will largely raise the amount of discrete values, as shown in the left plot of Figure 4.8.

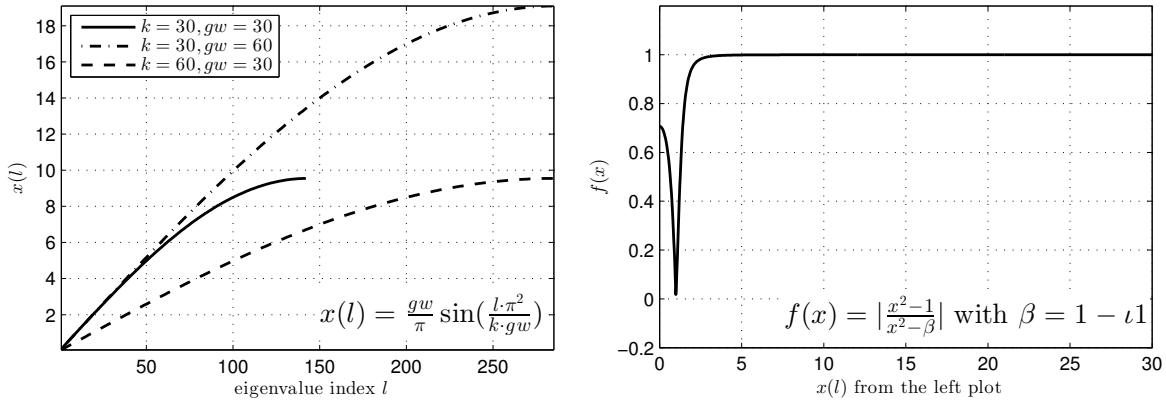


Figure 4.8: The mechanism of the eigenvalue computation of AM^{-1}

Such a mechanism has a significant influence on the final result, as shown in the right plot of Figure 4.8. Only small value x (but not too small) results in small $f(x)$. For a large k, more small $x(l)$'s result in more value $f(x)$'s, which is the modulus of the eigenvalue. For a large gw, the amount of small $x(l)$'s is not raised and thus the amount of small $f(x)$'s maintains the same.

4.2.5 A Variant of the Deflation Operator

A variant of deflation operator was proposed in [28]. The difference lies in the Galerkin matrix. In terms of left preconditioning, the deflation operator of the variant version is

[2]The range is the interval $[\min_{1\leqslant l\leqslant n}\{x(l)\}, \max_{1\leqslant l\leqslant n}\{x(l)\}]$

defined by

$$P_D := I - ZE^{-1}Y^T A \quad \text{where } E := Y^T AZ.$$

Here, the Galerkin matrix is associated with the original matrix A rather than the preconditioned system $M^{-1}A$. Then, the deflation preconditioned system is given by

$$P_D M^{-1} A\, u = P_D M^{-1}\, b.$$

Figure 4.9 shows the spectrum distribution of $P_D M^{-1} A$. It is apparent that the result is different from using the original deflation operator in [10]. The eigenvalues are not tightly clustered and there are some small eigenvalues around the origin. Because the deflation does not involve the effect of shifted Laplacian preconditioner, the increase in k leads to more eigenvalues which are loosely located away from the point $(1, 0)$.

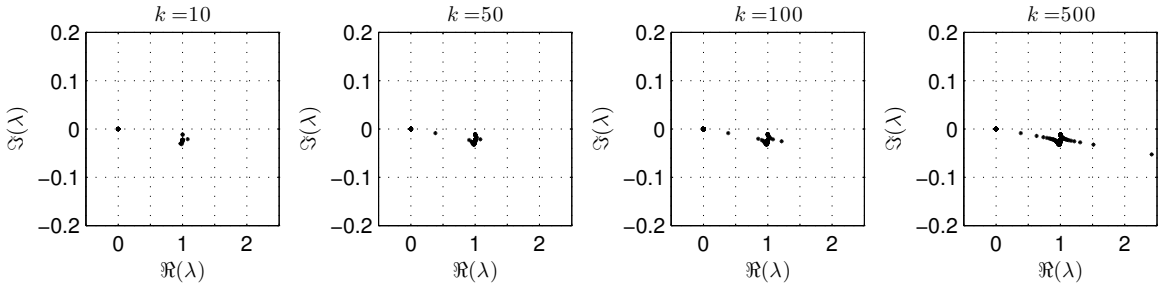


Figure 4.9: The spectrum distributions of the deflation preconditioned system $P_D M^{-1} A$. *Please notice the different axis scaling in x and y directions.*

4.3 The Approximation in MKMG

In the multilevel Krylov multigrid method, the matrices on different levels are obtained by Galerkin projection. According to the definition of the deflation operator, Galerkin projection should be consecutively applied to the entire preconditioned system, which is denoted by

$$(AM^{-1})_{(k)} := R_{(k-1,k)}(AM^{-1})_{(k-1)}P_{(k-1,k)}$$

on the k-th level. But in the practical implementation, the Galerkin matrix is replaced by an approximation $A_{(k)}M_{(k)}^{-1}B_{(k)}$, in which the Galerkin projection is applied respectively to each matrix as

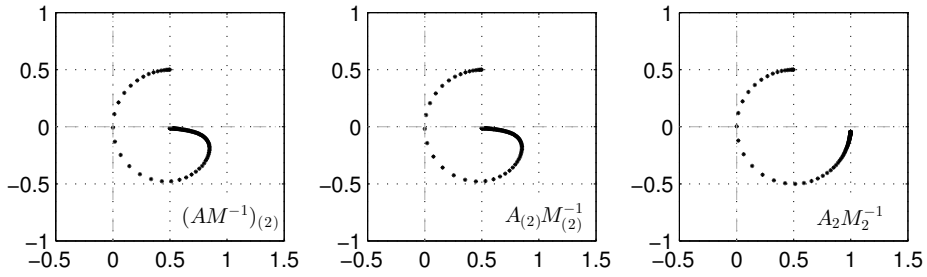
$$X_{(k)} := R_{(k-1,k)}X_{(k-1)}P_{(k-1,k)} \quad \text{where } X \text{ is } A, M, B.$$

Figure 4.10 shows that the approximation replacement is suitable.

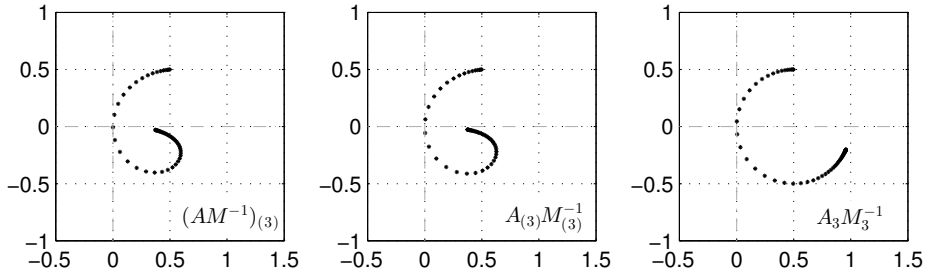
The first column shows the spectrum distribution of the theoretically constructed system $(AM^{-1})_{(k)}$. The second column shows that of the practically constructed system $A_{(k)}M_{(k)}^{-1}B_{(k)}$. For the sake of comparison, the preconditioned system $A_k M_k^{-1}$ is also presented, where both A_k and M_k are obtained by direct discretization on the k-th level.

The comparison has validated the replacement of $(AM^{-1})_{(k)}$ by $A_{(k)}M_{(k)}^{-1}B_{(k)}$. The similar spectrum distribution guarantees that the approximation replacement will have almost the same convergence behaviour as the theoretical construction. The only small difference between the first and second columns does not lie somewhere away from the origin, which can avoid the amplification of any small inaccuracy around zero.

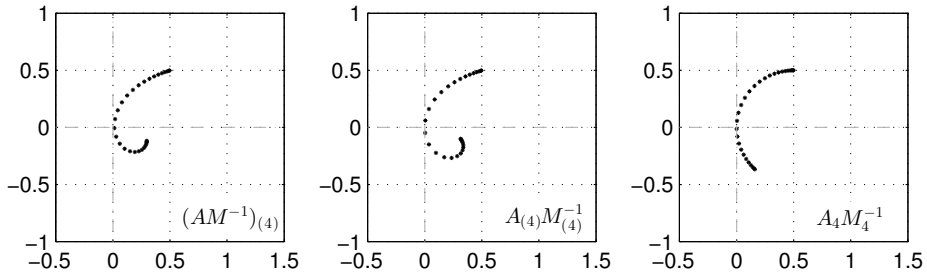
It is natural to think about replacing $(AM^{-1})_{(k)}$ with $A_k M_k^{-1}$ because the direction discretization operators A_k and M_k require no matrix multiplication and then can save the computational cost. However, as shown in the third column, the spectrum distribution of $A_k M_k^{-1}$ is very different from the other two, which implies there would not be high similarity between $(AM^{-1})_{(k)}$ and $A_k M_k^{-1}$.



(a) on the second-level of a four-level grid



(b) on the third-level of a four-level grid



(c) on the fourth-level (in the coarsest gird)

Figure 4.10: The influence of different approximations on the spectrum distribution in a four-level grid

4.4 Multigrid Analysis

In the practical implementation, the iteration is conducted in a multilevel grid. Only the matrix on the lowest level is inverted exactly and the inversions on the higher levels are approximated. In order to investigate the approximation $\mathbb{M}^{-1}$ by multigrid method, Fourier analysis is now applied to the multilevel grid.

4.4.1 Approximated Spectrum

Approximated Shifted Laplacian Preconditioning

The shifted Laplacian preconditioner is intended to improve the spectrum distribution of the discrete Helmholtz equation by moving the eigenvalues to a small region located between 0 and 1 on the real axis. So the effect of the approximation by the multigrid method is reflected by the spectrum distribution of the approximated preconditioned system. Provided that the approximated spectrum distribution is close to the exact one, it is of little concern whether the error in the iteration of solving M^{-1} has been reduced to a certain tolerance.

According to the previous explanation, the coarse grid correction introduces perturbation to the linear system. The perturbation is enhanced by employing more coarse grids. Such a perturbation growth is now demonstrated by the plots in Figure 4.11, where the spectrum distribution is plotted for the approximately preconditioned system $(I - T_1^m)M^{-1}A$ with

$m = 1, 2, 3, 4$. The one-grid analysis is just the exactly preconditioned system $M^{-1}A$ since the inversion is computed exactly in the sole grid.

It is obviously seen that the approximation in a multigrid introduces larger deviation from the exact preconditioning. The spectrum distribution of the exact preconditioning is circular and connected while the circular shape of the approximated ones are slightly distorted and split somewhere. However, all deviations are widely acceptable since all the approximated spectra are still bounded by the same square region that bounds the exact one. In terms of improving the spectral property, even the four-grid approximation has successful reached the desirable effect.

Remark It is worth noticing that the deviation mainly appears around the point $(1, 0)$. There is no severe distortion of splitting around the origin. So the approximation inaccuracy would not be amplified due to some very small value around zero.

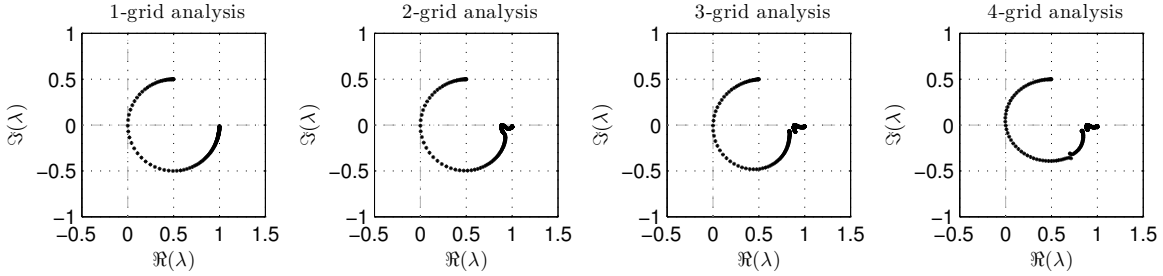


Figure 4.11: The influence of grid levels on the spectrum of the approximated system AM^{-1} after one step iteration

For the multi-grid approximation, the deviation of spectrum distribution can be easily reduced by using more iteration steps, as shown in Figure 4.12. The plots are the spectrum distribution of $(I-(T_1^4)^k)M^{-1}A$ with $k = 1, 2, 3, 4$. Comparing the plots of different steps, the observation tells that two or three iteration steps are enough to obtain a well-approximated result and using four iteration steps almost yields the same distribution as the exact one.

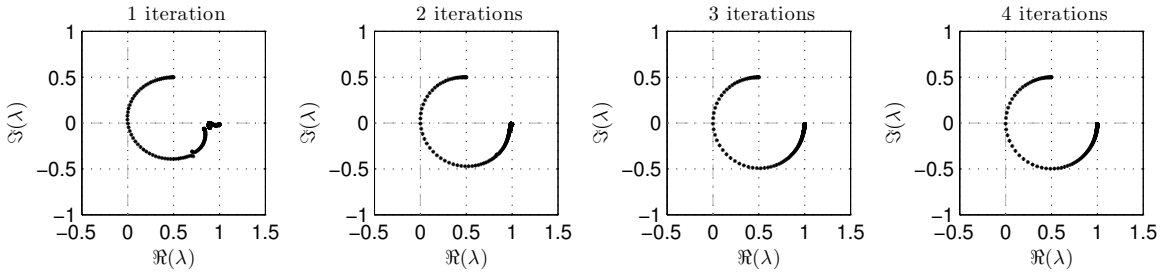


Figure 4.12: The influence of iteration steps on the spectrum of the approximated system AM^{-1} in a four-level grid

These two figures substantiates the application of multigrid method as a solver for the shifted Laplacian preconditioner. A good approximation can be cheaply obtained by using just several multi-grid iterations. The exact inversion can be done in a fairly coarse grid, which makes the solution much easier.

Approximated Deflation Preconditioning

The construction of the deflation operator is based on the preconditioned system AM^{-1}. When the AM^{-1} is approximated, then the deflation will also be effected even if the inversion

in the deflation operator is computed exactly. Figure 4.13 shows the spectrum distribution of the approximated system

$$AM^{-1}Q = AM^{-1}\left(I + P\hat{E}^{-1}R(\lambda_n I - AM^{-1})\right) \quad \text{where } \hat{E} = R(AM^{-1})P.$$

Here, all the AM^{-1}'s are approximated by the multigrid method, which means the Galerkin matrix $\hat{E}$ is associated with an approximated preconditioned matrix. The result shows that one iteration multigrid method is not enough to approximate the deflation preconditioning. The eigenvalues are not tightly clustered. But one more iteration can greatly improve the distribution to a tight clustering. However, even four iterations can not leads to a distribution that is visibly identical to the theoretical one. It is more difficult to approximate the deflation preconditioning $AM^{-1}Q$ than the sole preconditioning AM^{-1}.

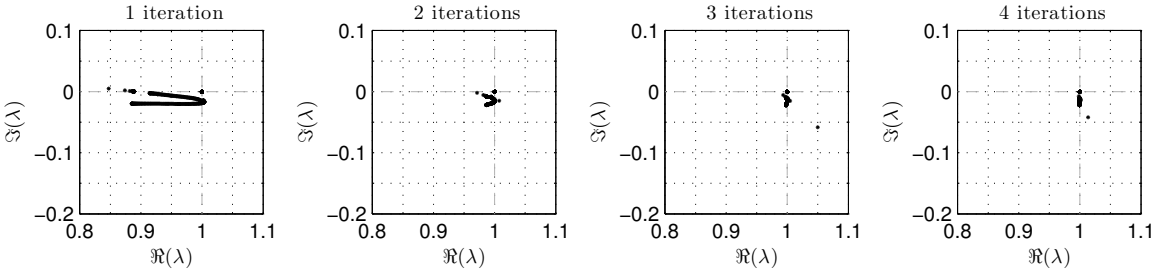


Figure 4.13: The influence of iteration steps on the spectrum of the approximated system $AM^{-1}Q$ in a four-level grid

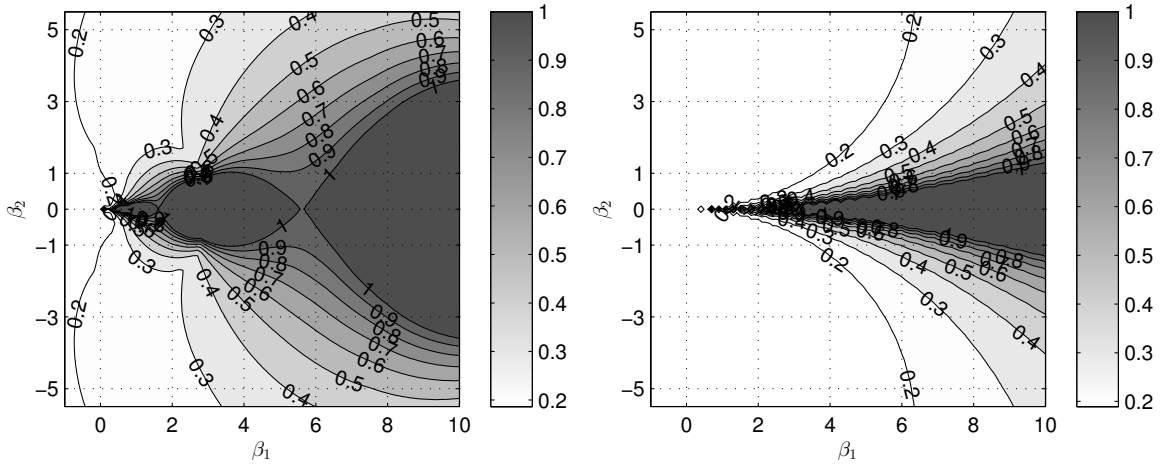
4.4.2 The Influence of the Shift on the Multigrid Convergence

The performance of the shifted Laplacian preconditioner is determined by its shift. For the multigrid method, the different choice of the shift has an important influence on the convergence factor of the iteration process. Figure 4.14 contains a set of four contour plots, which shows the magnitude of the convergence factor in a four-grid iteration with respect to β_1 and β_2. The darkest region in the plot refers to the choices that make the convergence factors larger than one, which leads to a divergence. A brighter region corresponds to the choices that results in a faster convergence.

Some useful observation can be obtained from the figure. Concerning the convergence behaviour, it is beneficial to choose a shift with a large imaginary part. The convergence factor decreases rapidly along the imaginary axis from zero in both positive and negative directions. The choice $\beta_1 + \iota\beta_2 = 1 + \iota 0$ is nothing but the discrete Helmholtz equation whose high indefiniteness fails the application of multigrid method. The darkest region indicates that the original Helmholtz equation is positively shifted and therefore the indefiniteness is being increased. So in order to guarantee a convergence, it is reasonable to have an imaginary shift and keep the real part unchanged, which refers to $1 + \iota\beta_2$. Besides that the convergence can be achieved by a shift of negative real part. But such a shift will deviate the preconditioner further away from the original matrix, which is not favoured by the Krylov convergence.

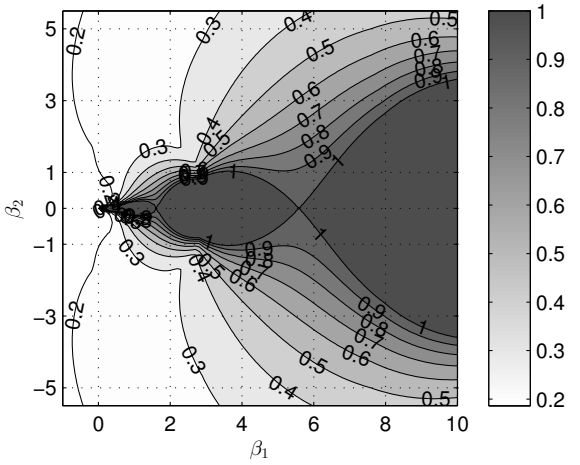
The set of plots reveals two properties that can simplify the research work. The symmetry of the plot shows the convergence factor is not related to the sign of the imaginary part. Furthermore, the wavenumber has no effect on the convergence factor.

However, the wave resolution has a significant influence on the convergence factor. As the wave resolution increase, it becomes easier to achieve the convergence. The darkest regions will reduce. And even a small shift in the imaginary part can lead to a very satisfactory convergence.

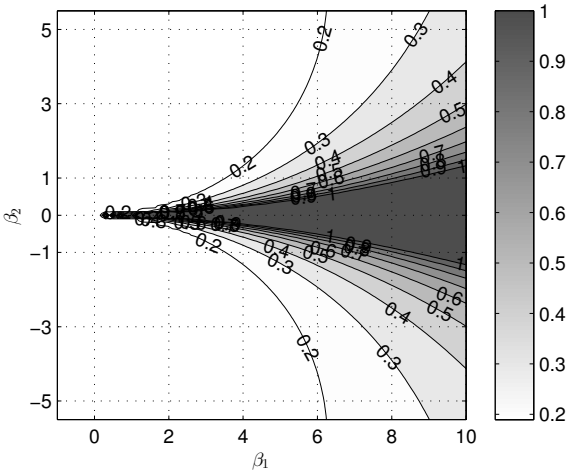


(a) $k = 30$ and $gw = 30$

(b) $k = 30$ and $gw = 120$



(c) $k = 120$ and $gw = 30$



(d) $k = 120$ and $gw = 120$

Figure 4.14: Contour plot of the convergence factor ρ_1^4 of the multigrid method

The Independence of the Sign of β_2

The contour of the convergence factor reveals the fact that the convergence factor is independent of the sign of β_2. Both β_2 and $-\beta_2$ result in the same convergence factor. This property can be explained by investigating the conjugate relation.

Proof of the β_2 sign independence. In the Helmholtz problem with Dirichlet boundary condition, the shifted Laplacian preconditioner results from shifting a real value matrix A by a complex value $\beta_1 + \iota\beta_2$. If A is shifted by $\beta_1 - \iota\beta_2$, then the conjugate relation holds

$$\overline{M}_{(\beta_1+\iota\beta_2)} = M_{(\beta_1-\iota\beta_2)}, \tag{4.11}$$

Here, the notation $\overline{\ }$ denotes the conjugation.

In the two-grid method, where the inversion is exact, the iteration operator reads

$$T_1^2 = S_1^{\nu 2} \left(I - P_2^1 M_2^{-1} R_1^2 M_1\right) S_1^{\nu 1}.$$

With the fact that both P and R are real, the conjugation of T_1^2 is given by

$$\overline{T_1^2} = \overline{S}_1^{\nu 2} \left(I - P_2^1 \overline{M}_2^{-1} R_1^2 \overline{M}_1\right) \overline{S}_1^{\nu 1}. \tag{4.12}$$

Using ω-Jacobi iteration, the smoothing operator S is given by

$$S = (1-\omega)I + \omega(I - \frac{1}{d}M) \quad \text{with } d = 2 - (\beta_1 + \iota\beta_2)h^2k^2,$$

which leads to the relation

$$\overline{S} = (1-\omega)I + \omega(I - \overline{M}/\overline{d}).$$

Then, a relation similar to (4.11) holds

$$\overline{S}_{(\beta_1+\iota\beta_2)} = S_{(\beta_1-\iota\beta_2)}. \tag{4.13}$$

Hence, substitution of (4.11) and (4.13) into (4.12) reveals the following relation

$$\overline{T_1^2}_{(\beta_1+\iota\beta_2)} = T_1^2{}_{(\beta_1-\iota\beta_2)}. \tag{4.14}$$

In the m-level multigrid method, the iteration operator on the k-th level is given by

$$T_k^m = S_k^{\nu 2}\left(I - P_{k+1}^k(I - T_{k+1}^m)M_{k+1}^{-1}R_k^{k+1}M_k\right)S_k^{\nu 1} \quad \text{for } 1 \leqslant k \leqslant m-1.$$

The conjugate relation can be easily proved by induction. The base case in the induction proof is already presented in equation (4.14). So the generalized conjugate relation is given by

$$\overline{T}_{(\beta_1+\iota\beta_2)} = T_{(\beta_1-\iota\beta_2)}. \tag{4.15}$$

For any matrix A, the relations between the eigenvalues of A and $\overline{A}$ are given by

$$\overline{\lambda}(A) = \lambda(\overline{A}) \quad \text{and} \quad |\lambda(A)| = |\lambda(\overline{A})|.$$

Together with (4.15), the final relation holds

$$\begin{aligned} &\overline{\lambda}(T_{(\beta_1+\iota\beta_2)}) = \lambda(\overline{T}_{(\beta_1+\iota\beta_2)}) = \lambda(T_{(\beta_1-\iota\beta_2)}) \\ \Longrightarrow\ &|\lambda(T_{(\beta_1+\iota\beta_2)})| = |\lambda(\overline{T}_{(\beta_1+\iota\beta_2)})| = |\lambda(T_{(\beta_1-\iota\beta_2)})|. \end{aligned} \tag{4.16}$$

The convergence factor is measured by the absolute value or the modulus. So the relation (4.16) well explains the independence of the sign of β_2. Using a β_2 of an opposite sign amounts to using the conjugate M, which leads to a conjugate iteration operator. The resulting eigenvalues are the conjugates but have the same modulus. □

k-independence

The observation of k-independence can be explained by the fact that k never acts alone in the representations of the multigrid iteration operator T_1^m. It always appears together with h in the product of h^2k^2. In case of a fixed wave resolution, the value of $h^2k^2 = (2\pi/gw)^2$ is confined to a constant. So the representations are independent of the change in k but only responds to the change in the wave resolution.

The smoothing factor in the iteration operator is given by

$$\mu = (1-\omega) + \frac{2\omega\cos(l\pi h)}{2 - (\beta_1 + \iota\beta_2)h^2k^2}. \tag{4.17}$$

In the two-grid correction operator the inversion on the coarse grid is computed exactly so $\tilde{A}_2^{-1}$ is a scalar. It then follows

$$\begin{aligned} \tilde{K} = \tilde{I} - \tilde{P}_2^1\tilde{A}_2^{-1}\tilde{R}_1^2\tilde{A}_1 = \tilde{I} - \tilde{P}_2^1\tilde{R}_1^2\tilde{A}_2^{-1}\tilde{A}_1 \quad \text{where} \\ \underbrace{\tilde{A}_2^{-1}}_{\text{scalar}}\tilde{A}_1 = \begin{bmatrix} \frac{4\sin^2(l\pi h/2) - (\beta_1+\iota\beta_2)h^2k^2}{\sin^2(l\pi h) - (\beta_1+\iota\beta_2)h^2k^2} & 0 \\ 0 & \frac{4\cos^2(l\pi h/2) - (\beta_1+\iota\beta_2)h^2k^2}{\sin^2(l\pi h) - (\beta_1+\iota\beta_2)h^2k^2} \end{bmatrix} \end{aligned} \tag{4.18}$$

In both (4.17) and (4.18), k also implicitly appears in the discrete value $\sin(\cdots)$ and $\cos(\cdots)$ since h is determined by the wave resolution together with k. However, k plays an almost trivial role in $\sin(\cdots)$ and $\cos(\cdots)$. In terms of a continuous viewpoint, $\sin(\cdots)$ is equivalent to $\sin(x)$ where x varies from 0 to π. So there is little influence of k on the computation of $\sin(\cdots)$ and $\cos(\cdots)$.

The conclusion on a two-level grid can be extended to the multigrid by induction. Thus, it has proved that the representation of the iteration operator is not related to the wavenumber k.

4.4.3 Optimal Shift for the Preconditioner

The construction of the preconditioner M is based on the original matrix A by shifting. The magnitude of the shift determines how far away the preconditioner is deviated from the original highly indefinite matrix. On this viewpoint a larger shift is favoured by the multigrid method for solving the inversion M^{-1}. However, Krylov subspace method is main solver of Helmholtz problem. The Krylov convergence will benefit from a preconditioner close to the original matrix, which means a smaller shift. Otherwise, the preconditioning for the Krylov solver becomes less useful.

Now the choice of the best shift is placed in a dilemma. In order to have the solution converge, a suitable shift should not only guarantee the convergence of the multigrid method but also be kept as small as possible. The convergence factor of the multigrid method is given by

$$\mathcal{G}(l, \beta_1, \beta_2) := \rho(T) = \max_{1 \leqslant l \leqslant n-1} |\lambda_l(T)|. \tag{4.19}$$

Thus, the optimal shift is defined as

$$(\beta_1 + \iota\beta_2)_{\text{opt}} := \arg\min\{|\beta_1 + \iota\beta_2| : \max_{1 \leqslant l \leqslant n-1} \mathcal{G}(l, \beta_1, \beta_2) \leqslant 1\}. \tag{4.20}$$

In order to secure the convergence, it is sensible to reduce the convergence rate to a constant slightly smaller than one as

$$(\beta_1 + \iota\beta_2)_{\text{opt}} := \arg\min\{|\beta_1 + \iota\beta_2| : \max_{1 \leqslant l \leqslant n-1} \mathcal{G}(l, \beta_1, \beta_2) \leqslant c < 1\}. \tag{4.21}$$

There will be some negative effect on the Krylov convergence but it is small.

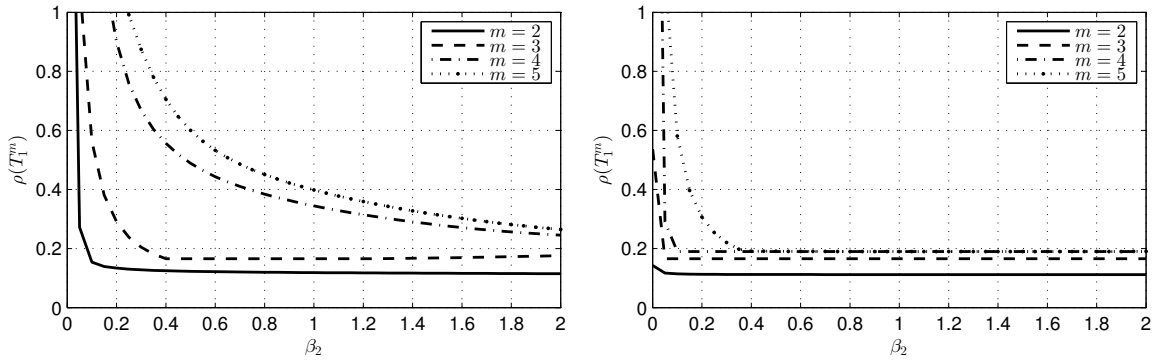
Generally, the real part of the shift is set as $\beta_1 = 1$. So the problem is reduced to finding out the smallest β_2 that can guarantee the multigrid convergence, namely the optimal β_2. As explained in section 4.4.2, the choice of the optimal β_2 should be independent of the wavenumber.

Theoretically, the analytical expression of $\mathcal{G}$ is a function of β_1 and β_2. But the derivation is not possible in a practical way due to the arithmetic complication, especially in the case of the multi-level grid analysis. So the optimal shift has to be found out by the numerical calculation.

Figure 4.15 shows the behaviour of convergence factor with respect to β_2. As a function of β_2, the convergence factor is monotonically decreasing. Besides that the increase in wave resolution will accelerate the convergence.

The result of the optimal β_2 is shown in Table 4.1. The optimal β_2 is largely influenced by the amount of levels. As more levels are used, it becomes less easy to converge and thus the optimal β_2 should be larger. It is preferable to see that the increase in wave resolution can greatly reduce the requirement on β_2 for convergence.

When the wave is extremely well resolved, the iteration already converges without the help of the imaginary shift. The shifted Laplacian preconditioner of $(\beta_1, \beta_2) = (1, 0)$ is identical to the original matrix A. Although it is convergence to solver the inversion M^{-1}, i.e. A^{-1},



(a) $gw = 30$ (b) $gw = 120$

Figure 4.15: The convergence factor for the m-level multigrid method when $\beta_1 = 1$

	$gw = 10$	$gw = 30$	$gw = 60$	$gw = 120$	$gw = 240$
$m = 2$	0.1096	0.0126	0	0	0
$m = 3$	0.3228	0.0616	0.0150	0	0
$m = 4$	0.3931	0.2002	0.0632	0.0155	0
$m = 5$	0.3931	0.2886	0.2012	0.0636	0.0156

Table 4.1: The optimal β_2 in the shift$1 + \iota\beta_2$ for $\rho(T_1^m) \leqslant c = 0.9$

it is impossible to solve the Helmholtz problem $Ax = b$ by multigrid method. Because the convergence factor is just a little smaller than one, which is far from sufficient as a solver. Secondly, the high wave resolution requires an impractically large memory storage.

4.5 A Newly Proposed Preconditioner

Recently, there is a new proposal for preconditioning the original matrix A. The new preconditioner is chosen as $M = A^*A$. In case of Dirichlet boundary condition, the matrix A is real and symmetric. The preconditioned matrix is given by

$$\hat{A} := AM^{-1} = A(A^*A)^{-1} = A^{-1}.$$

The preconditioning actually has inverted the matrix A. The magnitude of $\lambda(A)$ is very large so the magnitude of $\lambda(A^{-1})$ is very small and close to zero.

In the numerical computation the inversion of M is approximated by the ω-Jacobi iteration. The approximated inversion is given by

$$\mathbb{M}^{-1} := (I - J_\omega)M^{-1}$$

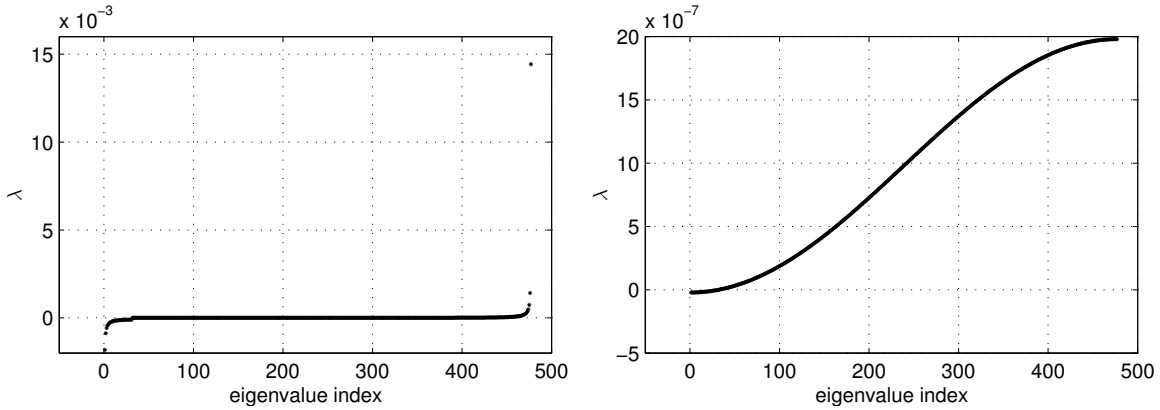
where

$$J_\omega = (1 - \omega)I + \omega(I - D^{-1}M) \quad \text{with } D := diag(diag(M)). \tag{4.22}$$

So the approximated preconditioned matrix is $A\mathbb{M}^{-1}$.

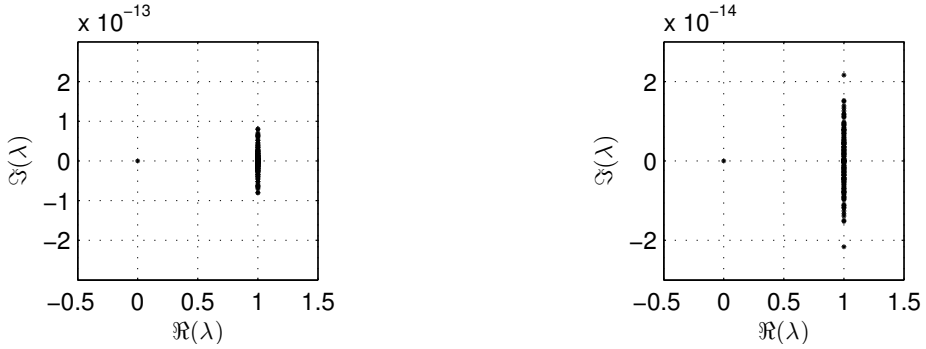
The eigenvalues of the preconditioned system are shown in Figure 4.16. All the eigenvalues are real. Although the two plots do not look similar, the axis scaling accounts for the difference. The inversion M^{-1} is approximated by one ω-Jacobi iteration. The approximation has a good match for most eigenvalues that are almost zero. But it loses the information of those eigenvalues whose modulus is relatively large.

The deflation preconditioning effect is shown in Figure 4.17. All the eigenvalues $\lambda(AM^{-1})$ are deflated towards 1. Some of the resulting eigenvalues around 1 contain a very small imaginary part. Others cluster at the origin.



(a) using the exact inversion M^{-1} (b) using the approximated inversion $\mathbb{M}^{-1}$

Figure 4.16: The eigenvalues of the preconditioned system AM^{-1} in a sorted sequence when $k = 100$ and $gw = 30$. *Please notice the different axis scaling in y direction.*



(a) using the exact inversion M^{-1} (b) using the approximated inversion $\mathbb{M}^{-1}$

Figure 4.17: The spectrum distribution of the deflation preconditioned system $AM^{-1}Q$ when $k = 100$ and $gw = 30$. *Please notice the different axis scaling in y direction.*

4.5.1 The failure of Rigorous Fourier Analysis

The eigenvalue analysis of the approximated inversion $\mathbb{M}^{-1}$ cannot be implemented by the rigorous Fourier analysis. Because it fails to find out an invariant subspace for the ω-Jacobi iteration operator.

Due to the definition of M, the elements of the diagonal matrix D in (4.22) are not uniform. The element $D(1,1)$ and $D(n,n)$ are identical to each other but different from other elements. So D cannot be treated as an identity matrix multiplied by a constant, as what happens in the shifted Laplacian preconditioner. Thus, D and M have different invariant subspaces. The application of the rigorous Fourier analysis has to stop in the ω-Jacobi operator.

In this case the local Fourier analysis can be used to tackle the problem. The two different end-elements will be considered as the boundaries; so they are ignored. The subspace will be invariant under the interior part of the operator.

Chapter 5

Numerical Experiments

In this chapter a series of numerical experiments is performed to attempt the Helmholtz problem. The observation reveals the effect of the preconditioning techniques as well as the intrinsic properties of the iteration solvers. In addition to that the numerical result gives validation to the analytical result by Fourier analysis in the previous chapter.

In order to keep consistent and make the results comparable, a unique one-dimensional Helmholtz equation is being computed for all experiments. The wave number is real and keeps constant throughout all the domain. The homogeneous Dirichlet boundary conditions are imposed on two end-points. A point source term is placed where it is close to the left boundary[1]. Mathematically, the model problem is given by

$$\begin{cases} -\Delta u(x) - k^2 u(x) = f(x) \text{ in } \Omega = (0,1), \\ f(x) = \delta(x - 0.1), \\ u(0) = u(1) = 0. \end{cases} \tag{5.1}$$

The continuous differential equation is discretized with a second-order finite difference method on a uniform grid. Without additional statement, the zero vector is used as the initial guess for all the iteration processes. And the tolerance $\|r_m\|_2/\|r_0\|_2 < 10^{-6}$ is set as the convergence criterion, which is the same as the default tolerance of the built-in Krylov solvers in MATLAB.

This model problem is proposed for studying the convergence behaviour of the iterative methods. Its physical meaning is of little concern. All the work that has been done is intended for the mathematical research though it is based on a physical model. In this point of view some computations are contributed to very high wavenumber. The wavenumber might be unphysically high but these computations complete the mathematical rigorousness.

5.1 Basic Convergence Behaviour

5.1.1 Overview

An overview of the convergence behaviour is obtained by solving the Helmholtz problem in three respective situations, namely A with no preconditioning, the sole preconditioning AM^{-1} and the deflation preconditioning $AM^{-1}Q$. In order to give a complete insight about the behaviour, the total number of iterations is set as the matrix size. The result is shown in Figure 5.1 by three iteration-residual curves.

This Helmholtz problem is of a moderate wavenumber; so the result is representative and the observation can be generalized. The unpreconditioned system leads to an extremely poor

[1] A source term closer to the boundary leads to a slower convergence. Such an unfavourable convergence property actually brings some convenience to the research work. Because the convergence behaviour of different methods would not be very close to each other, which makes the observation easier.

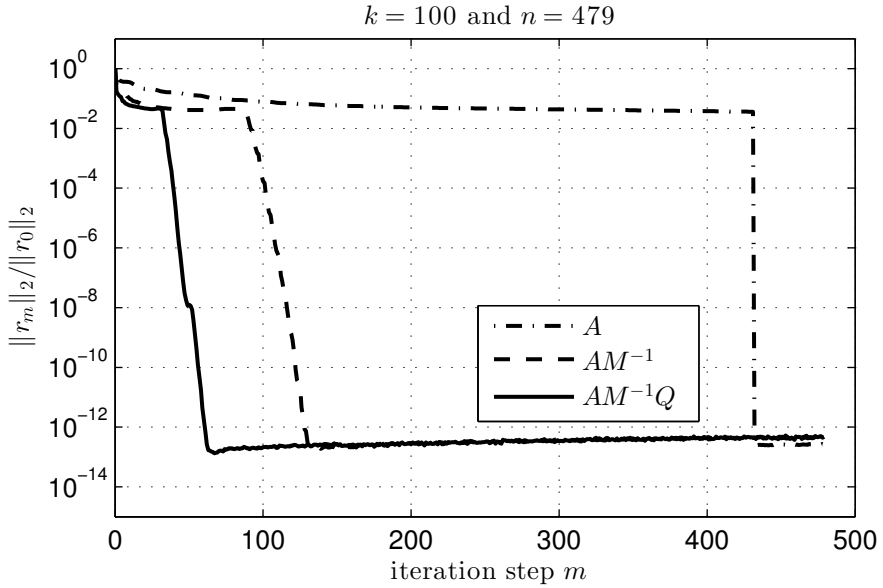


Figure 5.1: Overview of the convergence behaviour by different preconditioning

Krylov convergence. After a long slow convergence stage, the residual has only been reduced by a factor of $\mathcal{O}(10^2)$, which means the failure of the solver. The residual does not drop to a very small value until the number of iterations has almost reached the value of system size. However, this behaviour well matches the property of Krylov methods as the iteration process must converge after at most n steps where n is the matrix size. In the practical computation $\|r_n\|_2$ only contains the machine error.

The two curves of the preconditioned systems show a good convergence behaviour. In both situations the residual begins to rapidly decline after a short stage of slow convergence. The process converges to the almost exact solution after an acceptable amount of iterations in total. The deflation preconditioning shows an obvious advantage over the sole preconditioning. Although both of them have a similar linear convergence rate during the fast stage, the deflation preconditioning suffers a much shorter stage of slow convergence and starts the fast convergence earlier. The deflation preconditioning only takes half of the iterations as the sole preconditioning to reach the same accuracy.

It is undoubted that the Helmholtz problem can be hardly solved without the help of preconditioning. The deflation preconditioning results in a better effect than the sole preconditioning.

5.1.2 The Influence of wave resolution

The system size is associated with the wave resolution gw, the value of which equals the amount of subintervals of each wave. However, the convergence would not be slowed down by the increase in the system size due to high wave resolution. The convergence behaviour[2] is summarized in Table 5.1 with respect to wave resolution. To the left of the slash is the iteration number for $AM^{-1}Q$ while on its right is that for AM^{-1}.

The observation from the above table has a perfect match with the result of Fourier analysis on the preconditioning in the previous chapter. When the solely preconditioned system is solved, the number of iterations almost keeps the same for different wave resolutions. Because the amount of unfavourable small eigenvalues of AM^{-1}, which are responsible for the slow convergence, is not raised by the increase in wave resolution. So the convergence would not deteriorate.

[2]The orthogonalization method is modified Gram-Schmidt.

gw	$k = 10$	$k = 50$	$k = 100$	$k = 200$	$k = 300$	$k = 400$	$k = 500$
10	15/15	39/69	61/112	107/204	143/292	185/366	238/510
30	15/21	30/68	44/109	85/210	120/300	156/386	199/478
60	7/21	13/68	43/120	74/216	85/310	110/404	145/502

Table 5.1: Number of iterations with respect to different wave resolution gw

In the case of deflation preconditioning, the convergence turns out to be faster as the wave resolution improves. Once the deflation is applied, all the eigenvalues will cluster around $(1, 0)$. The convergence would never be retarded by the small eigenvalues around the origin. The increase in wave resolution leads to a larger system and therefore results in more favourable eigenvalues of AM^{-1}. Thus, the convergence is accelerated.

However, the total computation time is not necessarily reduced in case of higher wave resolution. A larger system is more demanding in the matrix-vector multiplication. But at least the computation time would not increase in the same fast speed as the wave resolution.

5.1.3 The functionality of deflation

The deflation preconditioning has a better effect than the sole preconditioning. The benefit gets larger as the wavenumber increases. Figure 5.2 shows the comparison of the number of iterations. The convergence of both preconditionings is proportional to the wavenumber. But the growing rate of the deflation preconditioning is far smaller than that of the sole preconditioning.

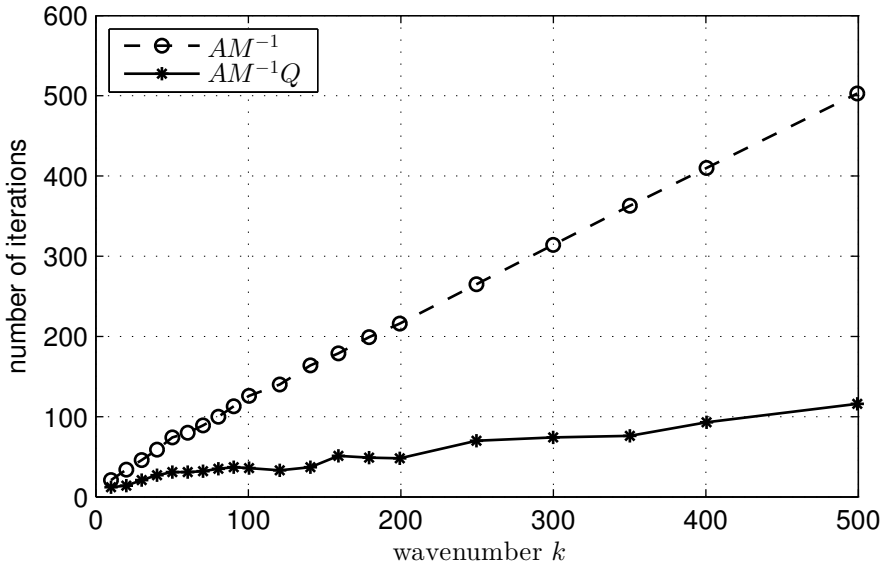
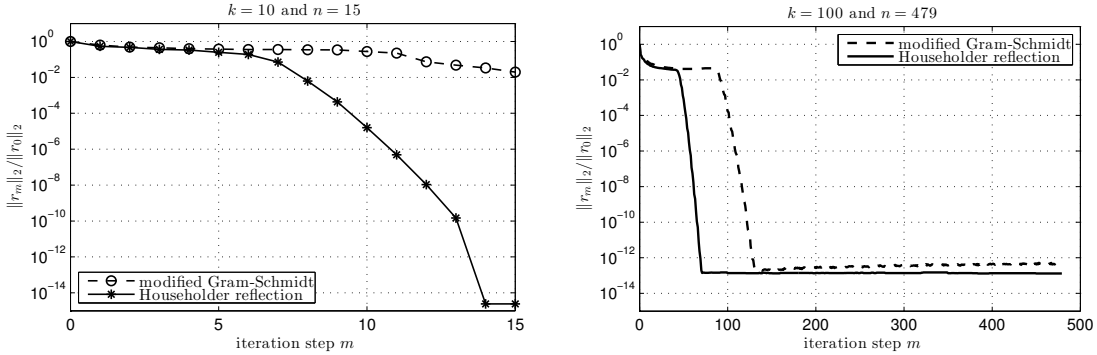


Figure 5.2: Number of iterations by two different kinds of preconditioning

According to the definition, the functionality of deflation is to move the small eigenvalues to the value of the largest eigenvalue so that the entire spectrum distribution is more clustered and keeps away from the origin. Then, the application of Krylov solver to a deflation preconditioned system will not suffer from the slow convergence due to the small eigenvalues. Thus, as the wavenumber increases, the spectral properties will not deteriorate. The numerical results can be supported by the Fourier analysis done in the previous chapter.

5.1.4 The influence of orthogonalization

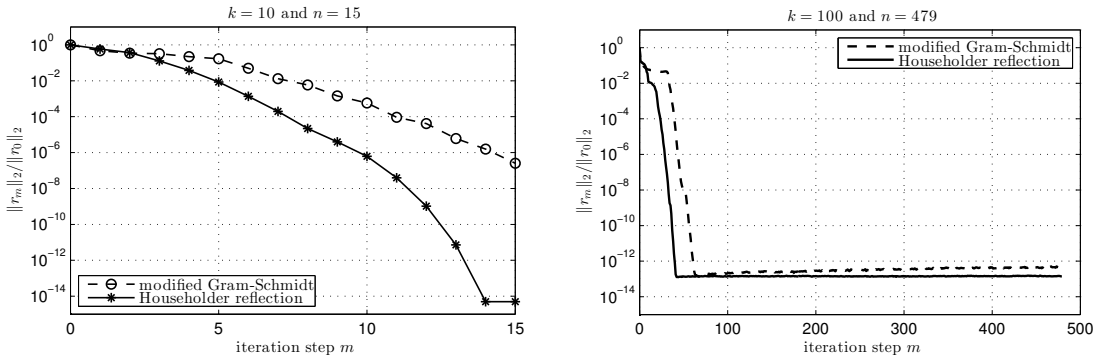
In GMRES, there are two orthogonalization methods to construct the orthonormal Krylov subspace. The orthogonalization with Householder reflection is less sensitive to the rounding error (more numerically stable) than that with modified Gram-Schmidt method. The observation in Figure 5.4 shows that such an advantage will lead to faster convergence.



(a) very small system (b) normal size system

Figure 5.3: Convergence behaviour of AM^{-1} by different orthogonalization methods

The observation can be obtained in both cases of sole preconditioning and deflation preconditioning. The explanation is that the error in orthogonalization adds to the total error of the iteration, which consequently retards the convergence. In the situation of an extremely small system, the Gram-Schmidt method can even not converge to the machine error while the Householder reflection is still stable enough to succeed. For a normal size system, both of them can finally reach the same accuracy except that it is always a little faster to use Householder reflection. In terms of computational cost, the choice of the orthogonalization method is a compromise between convergence speed and computational time.



(a) very small system (b) normal size system

Figure 5.4: Convergence behaviour of $AM^{-1}Q$ by different orthogonalization methods

5.2 The Influence of Approximation on the Convergence

In the practical computation for solving the Helmholtz problem, both the inversion of shifted Laplacian preconditioner M^{-1} and the deflation operator Q are approximated during the Krylov convergence. Table 5.2 lists the number of Krylov iterations for the systems which are

approximated to different degrees[3]. The data in every cell contains the number of iterations for $AM^{-1}Q$, $AM^{-1}\mathbb{Q}$ and $A\mathbb{M}^{-1}\mathbb{Q}$ respectively. Here, $AM^{-1}Q$ denotes the system with both the exact inversion M^{-1} and the exact construction of Q; $AM^{-1}\mathbb{Q}$ means that Q is approximated by MKMG but the M^{-1}'s on all the levels are exactly computed; $A\mathbb{M}^{-1}\mathbb{Q}$ refers to the practical implementation where both M^{-1} and Q are approximated.

gw	$k=10$	$k=50$	$k=100$	$k=200$	$k=300$	$k=400$	$k=500$
10	6/9/9	11/18/18	14/26/27	21/43/44	28/59/61	33/71/70	39/98/111
30	4/5/5	6/13/13	6/15/17	8/36/37	9/54/55	10/73/75	11/92/ 94
60	3/4/4	4/ 5/ 8	4/ 7/ 9	5/12/16	6/18/22	6/24/27	6/32/ 34

Table 5.2: Number of iterations with respect to different degrees of approximations i.e. $AM^{-1}Q$ / $AM^{-1}\mathbb{Q}$ / $A\mathbb{M}^{-1}\mathbb{Q}$

It is clear that the approximation does have some negative effect on the convergence. When all the matrices are exact, the convergence is much faster and only a few steps of iterations is required. Once the approximation is involved, the convergence will be slowed down even in the case of the exact M^{-1}. When Q is approximately constructed by MKMG, there is no significant influence whether or not the M^{-1} is exact. The little influence of the inaccuracy in M^{-1} is verified by the experiment on the solely preconditioned system AM^{-1}, whose data is listed in Table 5.3.

gw	$k=10$	$k=50$	$k=100$	$k=200$	$k=300$	$k=400$	$k=500$
10	11/11	36/36	60/60	108/105	153/149	193/188	265/258
30	12/12	36/36	60/58	114/108	161/152	209/196	255/240
60	12/12	36/36	63/62	113/111	161/158	207/204	255/250

Table 5.3: Number of iterations with respect to different degrees of approximations i.e. AM^{-1} / $A\mathbb{M}^{-1}$

There is tiny difference between the two numbers in each cell. First of all, it substantiates the successful application of multigrid method to approximating the inversion M^{-1}. Secondly, when the wavenumber is high, it even takes slightly fewer iteration steps by using the approximation. The pleasant surprise can be explained by the fact that the application of preconditioning is intended for improving the spectral properties. It is of little importance whether the inversion M^{-1} is closely approximated. The inaccuracy happens to bring out more eigenvalues closer the favourable point $(1,0)$, which are advantageous accelerate the Krylov convergence.

Remark The different behaviour of the inaccuracy in the approximated M^{-1} and Q gives the numerical verification to the result of Fourier analysis in the previous chapter. It has already been seen that it is more difficult to approximate the deflation preconditioning even when the inversion of the small Galerkin matrix is exact.

5.2.1 Smoothing Steps in the Multigrid Methods

In the multigrid method ν_1 is the number of pre-smoothing steps and ν_2 is that of post-smoothing steps. In the eigenvalue analysis it is impossible to distinguish the independent behaviour of the two smoothings, for the equality $\lambda(S^{\nu_2}\cdot K\cdot S^{\nu_1}) = \lambda(S^{(\nu_1+\nu_2)}\cdot K)$ has merged the effect of the pre- and post- smoothing. But the numerical experiment on various pairs of (ν_1,ν_2) is able to give a direct understanding of the independent behaviour.

[3]The orthogonalization method is Householder reflection. So the value for $A\mathbb{M}^{-1}\mathbb{Q}$ is smaller than its counterpart in Table 5.1 on page 53.

Table 5.4 lists the number of iterations for the solely preconditioned system AM^{-1}. The comparison shows that the convergence is not sensitive to the smoothing. One or two steps of either pre- or post- smoothing is sufficient. And more steps of smoothing does no more good to the convergence. So there is no distinction between pre-smoothing and post-smoothing in the case of sole preconditioning.

(a) using one multigrid iteration

ν_2 \ ν_1	0	1	2	3	4
0	/	87	60	56	55
1	86	60	56	55	54
2	59	55	54	53	54
3	55	54	53	54	54
4	54	53	54	54	55

(b) using two multigrid iterations

ν_2 \ ν_1	0	1	2	3	4
0	/	58	58	59	59
1	58	58	59	59	60
2	58	59	59	60	60
3	59	59	60	60	60
4	59	60	60	60	60

Table 5.4: Number of iterations for AM^{-1} with respect to different smoothing steps in the multigrid method when $k = 100$ and $gw = 30$

The number of iterations for the deflation preconditioned system $AM^{-1}Q$ is listed in Table 5.5, which presents a different pattern. More steps of smoothing will be helpful for the convergence. Furthermore, it is worth doing more post-smoothing than pre-smoothing. It is very efficient to have two steps of post-smoothing with no pre-smoothing. After that, more steps of smoothing would not improve the convergence.

(a) using one multigrid iteration

ν_2 \ ν_1	0	1	2	3	4
0	/	43	41	37	36
1	24	23	24	24	24
2	16	16	17	16	16
3	14	14	14	13	13
4	13	13	13	13	13

(b) using two multigrid iterations

ν_2 \ ν_1	0	1	2	3	4
0	/	38	33	31	28
1	21	18	19	18	18
2	15	15	16	16	15
3	15	15	15	16	16
4	16	15	16	16	16

Table 5.5: Number of iterations for $AM^{-1}Q$ with respect to different smoothing steps in the multigrid method when $k = 100$ and $gw = 30$

The comparison of the subtable (a) and (b) within in every table shows that two multigrid iterations would not lead to a faster convergence than one iteration, especially when the solely preconditioned system AM^{-1} is being solved. The slightly strange behaviour corresponds with the result in Table 5.3 on the previous page. Therefore, the same explanation can account for the influence of smoothing in the multigrid method.

The specific multigrid setup depends on the exact problem, i.e. k and gw. Without loss of generality, it is reasonable to use one iterations with $(\nu_1, \nu_2) = (1, 2)$. There will not be a big loss in the convergence behaviour when this setup is used for other problems of different k and gw. Because all the early analysis and results have shown that the multigrid approximation has done a good job in preconditioning. More effort brings little help to the convergence.

5.2.2 Iteration Steps in Multilevel Krylov Method

At each step of the Krylov iteration for $AM^{-1}Q$, the multilevel Krylov method approximates the deflation operator Q by taking several steps of flexible GMRES iterations on different levels. The internal iterations take place from the second level until the $(m-1)$-th level and the inversion of the Galerkin matrix is computed exactly on the m-th level. Actually the

number of Krylov iterations determines the dimension of the Krylov subspace that spans the approximate solution.

The MKMG setup $(\diamond, \#_2, \cdots, \#_{m-1}, \circ)$ denotes the number of Krylov iterations on different levels, where $\diamond$ and $\circ$ is only to state the position of the first and last level of the multilevel Krylov iteration.

In Figure 5.5, the MKMG setup $(\diamond, 2, 2, 2, 2, \circ)$ is considered as the standard reference. The result shows that more Krylov iterations on the higher levels are helpful for the convergence. However, it is unnecessary to have more Krylov iterations on the lower levels, which may instead slow down the convergence.

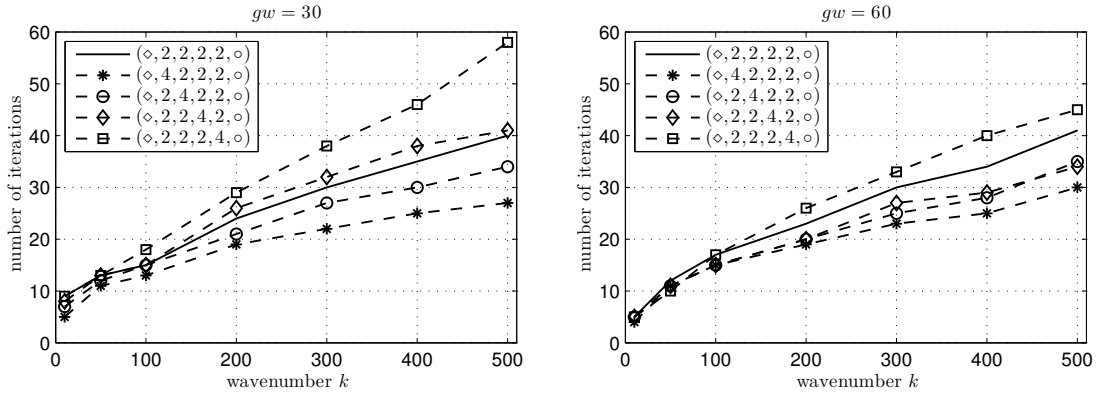


Figure 5.5: Number of iterations with respect to different MKMG setup in a six-level grid

It is straightforward to understand that the iteration on the higher levels brings more benefit to the convergence of the whole iteration. The Galerkin system on the higher levels are more closely connected with the original Helmholtz equation. As the level goes down, the smaller Galerkin system gets further away from the original system and its influence is weakened.

Furthermore, it is not necessarily cheaper to have more iterations on lower levels. Due to the recursive iteration in MKMG, although the system on the lower level is small, one iteration on the first level actually requires a multiple of $\#_k$ iterations on the k-th level in total. For instance, the setup $(\diamond, 2, 2, 4, 2, \circ)$ will lead to 2 iterations on the second level but $2 \times 2 \times 4 = 16$ iterations on the fourth level for every iteration on the first level.

In terms of a general viewpoint, it is more efficient to invest the iteration on the higher levels.

5.3 The Best Shift for the Krylov Convergence

The shift $\beta_1 + \iota\beta_2$ in the shifted Laplacian preconditioner plays a critical role in the preconditioning of Helmholtz problem. Its spectral influence on the preconditioning has been discussed in section 4.2.2. In this section its influence on the Krylov convergence will be investigated by numerical experiments.

A best shift is defined as the one that leads to the fastest Krylov convergence, i.e. fewest number of iterations. Since the real part of the shift is generally fixed as $\beta_1 = 1$, the task is reduced to find out the best β_2 that leads to the fewest number of iterations for the Krylov convergence.

The result is listed in Table 5.6 and 5.7 respectively for the sole preconditioning and deflation preconditioning. The independence of the sign of β_2 was proved in section 4.4.2 so only the positive β_2 is listed. In the parentheses is the fewest number of iterations by using the best β_2. During the computation of the best β_2, the minimal step size for β_2 is chosen

as 0.01. So some of the resulting best β_2's are identical. If the minimal step size is chosen smaller, then the small difference will be distinguishable.

	$k = 10$	$k = 50$	$k = 100$	$k = 200$	$k = 300$	$k = 400$	$k = 500$
$gw = 10$	0.01(7)	0.01(17)	0.01(27)	0.04(47)	0.02(67)	0.04(85)	0.05(116)
$gw = 30$	0.01(6)	0.01(11)	0.01(16)	0.01(24)	0.07(31)	0.08(39)	0.06(47)
$gw = 60$	0.01(5)	0.01(7)	0.01(9)	0.01(12)	0.01(15)	0.02(17)	0.01(21)

Table 5.6: The best β_2 that leads to the fastest convergence for the solely preconditioned AM^{-1} when $\beta_1 = 1$

	$k = 10$	$k = 50$	$k = 100$	$k = 200$	$k = 300$	$k = 400$	$k = 500$
$gw = 10$	0.01(7)	0.01(14)	0.01(20)	0.35(34)	0.33(46)	0.39(53)	0.37(74)
$gw = 30$	0.01(5)	0.13(10)	0.16(16)	0.09(23)	0.10(29)	0.10(36)	0.10(43)
$gw = 60$	0.01(4)	0.01(8)	0.03(9)	0.02(14)	0.02(15)	0.02(17)	0.01(21)

Table 5.7: The best β_2 that leads to the fastest convergence for the deflation preconditioned $AM^{-1}Q$ when $\beta_1 = 1$

When the solely preconditioned system AM^{-1} is solved, the best β_2 is very small. The smaller β_2 makes the AM^{-1} closer to the identity matrix and then it is easier of the Krylov iteration to converge.

When the deflation preconditioned system $AM^{-1}Q$ is solved, the best β_2 is a little larger. The multigrid method not only approximates the M^{-1} for the original matrix A but computes the approximation of $M_{(k)}^{-1}$ in the deflation operator. More complication is introduced by the deflation so it requires a larger β_2 that makes the multigrid convergence easier.

Compared to the optimal β_2 that was studies in section 4.4.3, the best β_2 is fairly smaller. Although the small β_2 is highly likely to fail the multigrid convergence, the results shows that it has still guaranteed the Krylov convergence. It once more proves that the accuracy of the approximated inversion M^{-1} is of trivial significance provided the spectral properties have been improved.

Remark In some situations the best β_2 is the smallest possible discrete value, i.e. the step size 0.01, which indicates an almost zero shift. In this case it should be distinguished that the preconditioning does not multiply the original matrix A by its exact inversion but by the multigrid approximated inversion $\mathbb{A}^{-1}$.

Chapter 6

Summary

6.1 Overview

The work in this thesis focuses on analyzing the spectral properties of the linear system that results from the discretization of one-dimensional Helmholtz problem with homogeneous Dirichlet boundary condition. The study on the spectrum distribution with respect to different parameters gives a reasonable explanation to the convergence behaviour of solving Helmholtz problem.

In Chapter 2 three types of iterative methods are introduced. Although the multigrid method cannot tackle the Helmholtz problem as the main solver, it can well approximates the inversion of shifted Laplacian preconditioner M. As the main solver, the Krylov subspace method solves the linear system $Ax = b$ or the preconditioned systems $AM^{-1}\tilde{x} = b$ and $AM^{-1}Q\tilde{x} = b$.

Because the spectral properties of the original matrix A are very unfavourable, the preconditioning plays the key role in solving the Helmholtz problem. Two types of preconditioning are introduced and discussed in Chapter 3. The application of shifted Laplacian preconditioner to the original matrix A can greatly improve the spectral properties by reducing the spectrum distribution to a compact circular shape. Although the convergence has now been largely accelerated, it could be further improved by moving the small eigenvalues to somewhere away from the origin. The deflation operator completes this task. For the sake of numerical stability, the small eigenvalues are deflated towards the eigenvalue of largest magnitude rather than zero; otherwise, the deflation effect would be spoiled by the inaccuracy in the construction of deflation operator.

In practical implementation, both the shifted Laplacian preconditioner and the deflation operator are computed approximately in an implicit way. So the varying preconditioning motivates the application of flexible GMRES. Because the small Galerkin system $\hat{E}$ in the construction of deflation operator is associated with the original matrix A, it is as difficult to compute its inversion $\hat{E}^{-1}$ as to solve the original Helmholtz problem. The MKMG in section 3.3 recursively approximates the inversion of small Galerkin systems on the lower levels and therefore completes the construction of deflation operator in an implicit way at each Krylov iteration step.

6.2 Conclusion

6.2.1 Analytical Results

Preconditioning effect The detailed spectral analysis has been conducted in Chapter 4 by using Fourier analysis. The study shows the functionality of shifted Laplacian preconditioner and deflation operator.

- The shifted Laplacian preconditioner restricts the eigenvalues to a circle, as proved mathematically in section 3.1.1 on page 22. But the increase in wavenumber will generates more small eigenvalues around the origin, which will retard the Krylov convergence.

- The deflation operator moves the small eigenvalues far away from the origin.

The shift $\beta_1 + \iota\beta_2$ plays the key role in the shifted Laplacian preconditioner. Generally, the real part β_1 is set to one so that the spectrum distribution of AM^{-1} can be restricted to the most compact shape. The study on the imaginary part β_2 shows that

- The imaginary part β_2 determines the length of the arc on which the eigenvalues are distributed. A smaller β_2 is always favourable for the Krylov convergence but theoretically will increase the difficulty in approximating M^{-1} by multigrid method.

- Either positive or negative β_2 results in the same preconditioning effect.

In addition to the wavenumber k, the wave resolution gw is the other factor that determines the size of the linear system. The study on gw shows that

- The increase in wave resolution would not deteriorate the spectral properties in case of AM^{-1} and that of AM^{-1} will be even more favourable.

Multigrid convergence Fourier analysis is also applied to the preconditioning by $\mathbb{M}^{-1}$, which is the approximated M^{-1} by multigrid method. The analysis shows that

- The multigrid method is capable of efficiently approximating the inversion M^{-1} by just several iteration at a cheap cost.

- The deflation preconditioning is more difficult to approximate. It requires more multigrid iterations for M^{-1} in order to provide a more accurately approximated M^{-1} and therefore to produce a better deflation preconditioning.

The analysis is also done on the convergence factor of multigrid method. The result shows that

- The convergence factor is independent of wavenumber and the sign of β_2. Both computational and theoretical proof have been provided on pages 46–47.

- It is easier to achieve the multigrid convergence for M^{-1} in case of high wave resolution, which results in a smaller optimal β_2 that is more favourable for the Krylov convergence..

6.2.2 Numerical Observations

In Chapter 5 a set of numerical experiments gives the observations about the different action of the preconditionings on the convergence behaviour. The numerical result substantiates the analytical conclusion by Fourier analysis in Chapter 4 while it also reveals some new valuable information that cannot be obtained by theoretical analysis.

- The deflation preconditioning $AM^{-1}Q$ outperforms the sole preconditioning AM^{-1} by a much smaller growth rate in the number of iterations as the wavenumber increases.

- The orthogonalization method in GMRES has a significant influence on the convergence behaviour. Householder reflection has the advantage over the modified Gram-Schmidt method in both convergence speed and convergence robustness.

- The multigrid method has done an efficient job in approximating the inversion M^{-1}. The convergence behaviour is little influenced by the inaccuracy in M^{-1}. It is of importance to obtain a better approximated deflation operator because the convergence is more vulnerable to the inaccuracy in the construction of deflation operator.
- In MKMG, it is more effective and efficient to do more internal Krylov iterations on the higher levels.

6.3 Suggestions on Future Work

This thesis has made a connection between the convergence behaviour of iterative methods and the spectral properties of the linear system. The achievement gives rise to more ideas that can be implemented in the future work.

Higher dimension So far, all the work is done on a one-dimensional Helmholtz problem. It is necessary to extend the analysis to the higher dimensional cases, especially the 2D Helmholtz problem which has an important application in engineering. The theory of Fourier analysis for the higher dimensions is in principle the same as that for one-dimension except that the formulae become more complex.

Local Fourier analysis The Fourier analysis that has been used is in the scope of rigorous Fourier analysis. In order to have a wider application, it is necessary to apply the local Fourier analysis. Then, the investigation of convergence behaviour can be done to those iteration operators that fail the rigorous Fourier analysis. In fact, rigorous Fourier analysis can be considered as the special case of local Fourier analysis while local Fourier analysis is the generalization.

Krylov solver The varying preconditioning has limited the type of Krylov solvers that can be employed. Besides the flexible GMRES that is used in this thesis, there are another two suitable options. They are generalized conjugate residual method [26] and induced dimension reduction method [31]. Both of them can handle the varying preconditioning. It is worth comparing the convergence behaviour of different Krylov solvers.

Appendix A

Optimal ω for the Jacobi Iteration

A.1 Problem Formulation

The ω-Jacobi iteration method, whose iteration operator is given by

$$J_\omega = (1-\omega)I + \omega D^{-1}(D-M) \quad \text{where } D = diag(diag(M)),$$

is adopted as the smoothing component of the multigrid method for approximating the inversion of the shifted Laplacian preconditioner

$$M = \frac{1}{h^2}\begin{bmatrix} d & -1 & & \\ -1 & d & -1 & \\ & \ddots & \ddots & \ddots \\ & & -1 & d \end{bmatrix} \quad \text{with } d = 2-(\beta_1+\iota\beta_2)h^2k^2.$$

Then, the smoothing operator is in the form

$$S := (1-\omega)I + \omega(I - \frac{1}{d}M). \tag{A.1}$$

The eigenvalue of the shifted Laplacian preconditioner is already known. Then, the eigenvalue of the smoother can be easily obtained. For $l = 1, 2, \cdots, n$, there is

$$\begin{aligned}\lambda(l,\omega) =& (1-\omega) + \omega(1 - \frac{2(1-\cos(l\pi h)) - (\beta_1+\iota\beta_2)h^2k^2}{2-(\beta_1+\iota\beta_2)h^2k^2}) \\ =& (1-\omega) + \frac{2\omega\cos(l\pi h)}{2-(\beta_1+\iota\beta_2)h^2k^2}.\end{aligned}$$

The functionality of the smoother is to quickly reduce the error of high frequency components, namely for $l = (n+1)/2, \cdots, n$. So the smoothing factor is defined by

$$\mu(\omega) := \max_{(n+1)/2 \leqslant l \leqslant n} |\lambda(l,\omega)|, \tag{A.2}$$

which gives the information of how much error at least could be reduced by one iteration. For the sake of convenience, a continuous equivalence to (A.2) is generally used so that this factor is n-independent. By taking $l\pi h = \theta$, the continuous smoothing factor is given by

$$\mu^*(\omega) := \max_{\pi/2 \leqslant \theta \leqslant \pi} |\lambda(\theta,\omega)|. \tag{A.3}$$

For the sake of brevity, two denotations are introduced in order to simplify the expression of λ.

$$A = 2-\beta_1 h^2k^2 \quad \text{and} \quad B = \beta_2 h^2k^2 \tag{A.4}$$

In terms of A and B, the expression of the modulus of the eigenvalue is given in its quadratic form by

$$\begin{aligned}|\lambda(\theta,\omega)|^2 =& \left(1-\omega+\frac{2\omega A}{A^2+B^2}\cos\theta\right)^2+\left(\frac{2\omega B}{A^2+B^2}\cos\theta\right)^2\\ =&(1-\omega)^2+\frac{4A}{A^2+B^2}\omega(1-\omega)\cos\theta+\frac{4\omega^2\cos^2\theta}{A^2+B^2}.\end{aligned} \tag{A.5}$$

Obviously, $|\lambda(\theta,\omega)|^2$ shares the same maximizer and minimizer with $|\lambda(\theta,\omega)|$.

A.2 Analytical Derivation

The fastest convergence results from the ω that minimizes the smoothing factor. This idea motivates the definition of the optimal ω

$$\omega_{\text{opt}} := \arg\min_{\omega\in(0,1]}\{\mu^*(\omega)\}. \tag{A.6}$$

The analytical expression of ω_{opt} can be obtained in the explicit form.

Maximizer θ_i The derivation of ω_{opt} starts from determining the θ that maximizes the modulus of eigenvalue. One candidate for the global maximizer over $[\pi/2,\pi]$ is $\theta_1=\pi/2$ as the left end point of the domain. There are two other possibilities obtained by

$$\begin{aligned}\frac{d}{d\theta}|\lambda(\theta,\omega)|^2 = \frac{4A\omega(\omega-1)\sin\theta}{A^2+B^2}-\frac{8\omega^2\sin\theta\cos\theta}{A^2+B^2}=0 \text{ for } \theta\in[\frac{\pi}{2},\pi]\\ \Longrightarrow\begin{cases}\theta_2=\pi \text{ (also as the right end point);}\\ \theta_3=\arccos\frac{A(\omega-1)}{2\omega} \text{ and } \sin\theta_3\neq 0.\end{cases}\end{aligned} \tag{A.7}$$

The θ_2 and θ_3, as the local maximizer of $\mu^*(\theta,\omega)$, should be verified by the negtiveness condition

$$\left.\frac{d^2}{d\theta^2}|\lambda(\theta,\omega)|^2\right|_{\theta_i,i=2,3}=\frac{4A\omega(\omega-1)\cos\theta_i}{A^2+B^2}-\frac{8\omega^2\cos 2\theta_i}{A^2+B^2}<0. \tag{A.8}$$

Then, the smoothing factor becomes

$$\mu^*(\omega)=\lambda(\theta_i,\omega)$$

Minimizer ω The ω that minimizes the smoothing factor is derived in a similar way by

$$\begin{aligned}\frac{d}{d\omega}|\lambda(\theta_i,\omega)|^2 = 2(\omega-1)+\frac{4A(1-2\omega)\cos\theta_i}{A^2+B^2}+\frac{8\omega\cos^2\theta_i}{A^2+B^2}=0\\ \Longrightarrow\omega_0=\frac{A^2+B^2-2A\cos\theta_i}{A^2+B^2-4A\cos\theta_i+4\cos^2\theta_i}.\end{aligned} \tag{A.9}$$

Besides that the candidate for the minimizer ω_{opt} is supposed to satisfy

$$\left.\frac{d^2}{d\omega^2}|\lambda(\theta_i,\omega)|^2\right|_{\omega_0}=2-\frac{8A\cos\theta_i}{A^2+B^2}+\frac{8\cos^2\theta_i}{A^2+B^2}>0. \tag{A.10}$$

Disqualified θ_3 Substitution of $\cos\theta_3=\frac{A(\omega-1)}{2\omega}$ into (A.9) yields $\omega_0=1$ and $\theta_3=\pi/2$. As a candidate for the local maximizer, $\theta_3=\pi/2$ is disqualified since the condition (A.8) is violated. So $\theta_3=\pi/2$ is eliminated but $\theta_1=\pi/2$ is still a candidate as the global maximizer.

The final choice Substitution of $\theta_1 = \pi/2$ and $\theta_2 = \pi$ into (A.5) yields

$$|\lambda(\theta_1,\omega)|^2 = (1-\omega)^2 \text{ and } |\lambda(\theta_2,\omega)|^2 = (1-\omega)^2 + \frac{4((A+1)\omega^2 - A\omega)}{A^2+B^2}.$$

So it follows[1]

$$\begin{aligned}\mu^*(\omega) &= \max\{|\lambda(\theta_1,\omega)|, |\lambda(\theta_2,\omega)|\} \\ &= \begin{cases} |1-\omega| & \text{for } \omega \leqslant \frac{A}{A+1}, \\ \sqrt{(1-\omega)^2 + \frac{4((A+1)\omega^2 - A\omega)}{A^2+B^2}} & \text{for } \omega \geqslant \frac{A}{A+1}. \end{cases}\end{aligned} \tag{A.11}$$

If ω_{opt} is located in $[\frac{A}{A+1}, 1]$, then the value of μ^* is determined by the point $\theta_2 = \pi$ and it is minimized by

$$\omega_2 = \frac{A^2+B^2+2A}{A^2+B^2+4A+4} \quad \text{with } \mu^*(\omega_2) = \sqrt{(1-\omega_2)^2 + \frac{4A\omega_2(\omega_2-1)+4\omega_2^2}{A^2+B^2}}.$$

Otherwise, the value of μ^* is obtained at the point $\theta_1 = \pi/2$ and the optimal ω is given by

$$\omega_1 = \frac{A}{A+1} \quad \text{with } \mu^*(\omega_1) = |1-\omega_1|.$$

Hence, the optimal ω is given by

$$\omega_{\text{opt}} = \max\{\omega_1, \omega_2\}. \tag{A.12}$$

A.3 Numerical Calculation

It is easily found that ω_{opt} is independent of the wavenumber k but dependent on the value of h^2k^2. h^2k^2 is associated with the wave resolution gw which refers to the amount of subintervals per wavelength. The plot in Figure A.1 shows that h^2k^2 has an significant influence on the determination of ω_{opt}. The explanation can be made by comparing the order of magnitude.

(a) $gw = 10$ i.e. $h^2k^2 = 0.395$ (b) $gw = 50$ i.e. $h^2k^2 = 0.016$

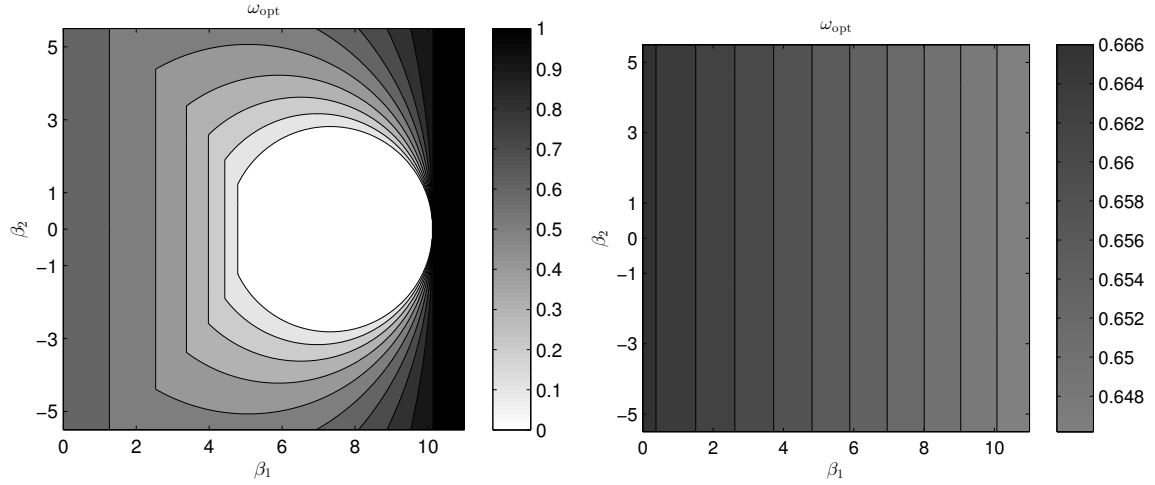


Figure A.1: The contour plot of ω_{opt} with respect to the shift $\beta_1 + \iota\beta_2$

[1]The piecewiseness is based on the condition $A+1 > 0$, i.e. $\beta_1 h^2k^2 < 3$.

Generally, the minimal requirement on the wave resolution is that every single wave is resolved by 10 subintervals, i.e. $gw = 10$, which leads to

$$kh = \mathcal{O}(10^{-1}) \Longrightarrow k^2h^2 = \mathcal{O}(10^{-2}).$$

Recall the definition of A and B in equation (A.4). For a well-resolved wave, i.e. small h^2k^2, it is valid to have

$$\left.\begin{aligned} \omega_1 =& \frac{2+\mathcal{O}(10^{-2})}{3+\mathcal{O}(10^{-2})} \approx \frac{2}{3}, \\ \omega_2 =& \frac{4+\mathcal{O}(10^{-2})+\mathcal{O}(10^{-4})+2(2+\mathcal{O}(10^{-2}))}{4+\mathcal{O}(10^{-2})+\mathcal{O}(10^{-4})+4(2+\mathcal{O}(10^{-2}))+4} \approx \frac{1}{2}, \end{aligned}\right\} \Rightarrow \omega_{\text{opt}} = \omega_1 \approx \frac{2}{3}. \tag{A.13}$$

The estimate gets sharp as β_1 is kept small. If the resolution is poor, i.e. large h^2k^2, the small terms cannot be ignored, making the estimate invalid. Especially, when β_1 is large, ω_{opt} is more likely to be determined by ω_2 which depends on both β_1 and β_2 as shown in Figure A.1a.

β_2-independence Figure A.1 also shows that the choice of ω_{opt} is independent of β_2 in most cases. Because these cases lead to the choice of ω_1 which only depends on β_1. As seen in (A.13), ω_1 is larger than ω_2 provided the wave is not poorly resolved. Even when the wave is a little poorly resolved, the estimate (A.13) will still keep valid provided β_1 is small.

Throughout the thesis, β_1 is fixed as one and the wave resolution gw would not be smaller than 10, which guarantees the condition $\omega_1 \geqslant \omega_2$. So it is valid to take the β_2-independent choice $\omega_{\text{opt}} = \omega_1$.

The β_2-independent ω_{opt} will bring convenience to the research since the convergence of ω-Jacobi iteration has now excluded the influence of β_2. The preconditioning effect of different β_2 has no relation with the choice of ω_{opt}, which simplifies the number of control variables.

Bibliography

[1] W. E. Arnoldi, *The principle of minimized iterations in the solution of the matrix eigenvalue problem*, Quarterly of Applied Mathematics, 9 (1951), pp. 17–29.

[2] A. Bayliss, C. I. Goldsteiny, and E. Turkel, *An iterative method for Helmholtz equation*, Journal of Computational Physics, 49 (1983), pp. 443–457.

[3] ——, *On accuracy conditions for the numerical computation of waves*, Journal of Computational Physics, 59 (1985), pp. 396–404.

[4] W. L. Briggs, V. E. Henson, and S. F. McCormick, *A Multigrid Tutorial*, SIAM, Philadelphia, 2nd ed., 2000.

[5] R. L. Burden and J. D. Faires, *Numerical Analysis*, Brooks/Cole, Boston, 9th ed., 2011.

[6] R. W. Clayton and B. Engquist, *Absorbing boundary conditions for wave-equation migration*, Geophysics, 45 (1980), pp. 895–904.

[7] Y. A. Erlangga, *Advances in iterative methods and preconditioners for the Helmholtz equation*, Archives of Computational Methods in Engineering, 15 (2008), pp. 37–66.

[8] Y. A. Erlangga and R. Nabben, *Deflation and balancing preconditioners for Krylov subspace methods applied to nonsymmetric matrices*, SIAM Journal on Matrix Analysis and Applications, 30 (2008), pp. 684–699.

[9] ——, *Multilevel projection-based nested Krylov iteration for boundary value problem*, SIAM Journal of Scientific Computing, 30 (2008), pp. 1572–1595.

[10] ——, *On a multilevel Krylov method for the Helmholtz equation preconditioned by shifted Laplacian*, Electronic Transaction on Numerical Analysis, 31 (2008), pp. 403–424.

[11] Y. A. Erlangga, C. Vuik, and C. W. Oosterlee, *On a class of preconditioners for solving the Helmholtz equation*, Applied Numerical Mathematics, 50 (2004), pp. 409–425.

[12] O. G. Ernst and M. J. Gander, *Why it is difficult to solve Helmholtz problems with classical iterative methods*, in Numerical Analysis of Multiscale Problems, I. G. Graham, T. Y. Hou, O. Lakkis, and R. Scheichl, eds., vol. 83 of Lecture Notes in Computational Science and Engineering, Springer, 2012, pp. 325–363.

[13] R. Fletcher, *Conjugate gradient methods for indefinite systems*, in Numerical Analysis Proceedings of the Dundee Conference on Numerical Analysis, 1975, G. A. Watson, ed., vol. 506, 1976, pp. 73–89.

[14] J. Frank and C. Vuik, *On the construction of deflation-based preconditioners*, SIAM Journal of Scientific Computing, 23 (2001), pp. 442–462.

[15] R. W. Freund, *A transpose-free quasi-minimal residual algorithm for non-Hermitian linear systems*, SIAM Journal on Scientific Computing, 14 (1993), pp. 470–482.

[16] R. W. Freund and N. Nachtigal, *QMR: a quasi-minimal residual method for non-Hermitian linear systems*, Numerische Mathematik, 60 (1991), pp. 315–339.

[17] N. A. Gumerov and R. Duraiswami, *Fast Multipole Methods for the Helmholtz Equation in Three Dimensions*, Elsevier Science, Oxford, 2005.

[18] I. Harai, *A survey of finite element methods for time-harmonic acoustics*, Computer Methods in Applied Mechanics and Engineering, 195 (2006), pp. 1594–1607.

[19] M. R. Hestenes and E. Stiefel, *Methods of conjugate gradients for solving linear systems*, Journal of Research of the National Bureau of Standards, 49 (1952), pp. 409–436.

[20] KTH, TU Berlin, TU Delft, and FAU, *Computer simulations for science and engineering*. `http://www.kth.se/en/studies/programmes/master/em/cosse`. COSSE is an Erasmus Mundus joint master programme.

[21] A. Laird and M. Giles, *Preconditioned iterative solution of the 2d Helmholtz equation*, eprints archive, University of Oxfor Computing Laboratory, 06 2002.

[22] R. B. Morgan, *A restarted GMRES method augmented with eigenvectors*, SIAM Journal on Matrix Analysis and Applications, 16 (1995), pp. 1154–1171.

[23] R. Nabben and C. Vuik, *A comparison of deflation and the balancing preconditioner*, SIAM Journal of Scientific Computing, 27 (2006), pp. 1742–1759.

[24] R. A. Nicolaides, *Deflation of conjugate gradients with applications to boundary value problems*, SIAM Journal on Numerical Analysis, 24 (1987), pp. 355–365.

[25] Y. Saad, *A flexible inner-outer preconditioned GMRES algorithm*, SIAM Journal on Scientific Computing, 14 (1993), pp. 461–469.

[26] Y. Saad, *Iterative methods for sparse linear systems*, SIAM, Philadelphia, 2nd ed., 2003.

[27] Y. Saad and M. H. Schultz, *GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM Journal on Scientific and Statistical Computing, 7 (1986), pp. 856–869.

[28] A. H. Sheikh, D. Lahaye, and C. Vuik, *A scalable Helmholtz solver combining the shifted Laplace preconditioner with multigrid deflation*, internal report, Delft University of Technology, Faculty of Electrical Engineering, Mathematics and Computer Science, Delft Institute of Applied Mathematics, 01 2011.

[29] I. Singer and E. Turkel, *High-order finite difference methods for the Helmholtz equation*, Computer Methods in Applied Mechanics and Engineering, 163 (1998), pp. 343–358.

[30] P. Sonneveld, *CGS, a fast lanczos-type solver for nonsymmetric linear systems*, SIAM Journal on Scientific and Statistical Computing, 10 (1989), pp. 36–52.

[31] P. Sonneveld and M. van Gijzen, *IDR(s): a family of simple and fast algorithms for solving large nonsymmetric systems of linear equations*, SIAM Journal on Scientific Computing, 31 (2008), pp. 1035–1062.

[32] U. Trottenberg, C. Oosterlee, and A. Schüller, *Multigrid*, Academic Press, London, 2001.

[33] A. van der Sluis and H. A. van der Vorst, *The rate of convergence of conjugate gradients*, Numerische Mathematik, 48 (1986), pp. 543–560.

[34] H. A. van der Vorst, *Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems*, SIAM Journal on Scientific and Statistical Computing, 13 (1992), pp. 631–644.

[35] M. B. van Gijzen, Y. A. Erlangga, and C. Vuik, *Spectral analysis of the discrete Helmholtz operator preconditioned with a shifted Laplacian*, SIAM Journal of Scientific Computing, 29 (2007), pp. 1942–1958.