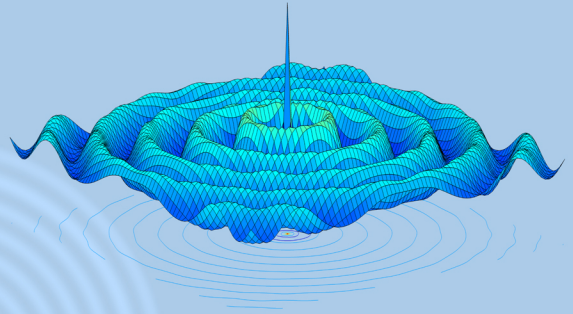


Helmholtz Solver based on SLP and Deflation

Abdul Hanan Sheikh



$$P_{h,2h} = M_h^{-1}[I_h - A_h Q_h] + \sigma Q_h$$

Development Of The Helmholtz Solver
Based On A Shifted Laplace Preconditioner
And A Multigrid Deflation Technique



Development of the Helmholtz Solver

based on

a Shifted Laplace Preconditioner and a Multigrid Deflation technique

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof.ir. K.C.A.M. Luyben,
voorzitter van het College van Promoties,
in het openbaar te verdedigen op dinsdag 18 November 2014 om 10.00 uur

door

Abdul Hanan Sheikh,
Master of Science (M.Sc.) Mathematics, Quaid-i-Azam University, Islamabad
Pakistan

geboren te Shikarpur, Pakistan.

Dit proefschrift is goedgekeurd door de promotor:
Prof.dr.ir. C. Vuik

Samenstelling promotiecommissie:

Rector Magnificus
Prof.dr.ir. C. Vuik
Dr.ir. D.J.P. Lahaye
Prof. dr. ir. H.X. Lin
Prof. dr R. Nabben
Prof. dr. I. Livshits
Prof. dr. W. Vanroose
Prof. dr. ir. E.C. Slob

voorzitter
Technische Universiteit Delft, promotor
Technische Universiteit Delft, co-promotor
Technische Universiteit Delft, NL
Technische Universitt Berlin, Germany
Ball State University, Indiana, US
Universiteit Antwerpen, Belgium
Technische Universiteit Delft



This thesis has been completed in partial fulfillment of the requirements of Delft University of Technology (Delft, The Netherlands) for the award of the Ph.D. degree. The research described in this thesis was supported in parts by two institutions. (1) Delft University of Technology, (2) QUEST Nawabshah-Sindh Pakistan. I thank them sincerely for their support.

Published and distributed by: Abdul Hanan Sheikh
E-mail: hanangul12@yahoo.co.uk

ISBN # 978-94-6186-355-3

Keywords: Helmholtz, CSLP, Shifted Laplace Preconditioner, Multigrid, Deflation, Fourier Analysis.

Copyright © 2014 by Abdul Hanan Sheikh

All rights reserved. No part of the material protected by this copyright notice may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage and retrieval system, without written permission of the author.

Printed in The Netherlands

To my parents,
Zulekha
my mother, who loves to see me reading books,
Mahmood-ur-Rahman
my father, who has been a great source of inspiration to me.

Preface

This booklet is culmination of the academic years I had spent in numerical analysis group at Delft University of Technology, Netherlands. I had my ups and downs, as usual thing during the PhD time. By and large, stay has been memorable experience. I learned many lessons, both personally and professionally. The topic covered in this thesis has made researchers toiling for few decades.

Indefiniteness of Helmholtz problems has been tricky for researching numerical methods. From Krylov to multilevel methods, from multigrid to domain decomposition, every iterative method has experienced difficulty to solve Helmholtz. Detecting the problem, by investing a lot on Fourier analysis, a tidy and simple-to-implement multilevel method has been presented. I have tried to keep my thesis self-explanatory. Even an appendix is dedicated to the implementation of solver. Considering the size of thesis, however it is always possible to miss out many bits. Still I will be reachable via email for help to understand thesis.

In brief, I have tried to address the difficult problems which encounters eigenvalues on both sides the axis, positive and negative. Solvers proposed are well justified by extensive Fourier analysis and its results. In addition, few variants are also discussed and compared. They also lead to new avenues of research. I truly would not have been able to finish this thesis, without help of many people.

Special thanks to Domenico Lahaye I wish to present sincere gratitude to Domenico for every thing he has done during my PhD candidacy. Yes! Every thing. His dedication can be thought of this that he never said no when I visit his office without any appointment. His progressive criticism has furnished me a lot made me to think critically, which is necessary for a researcher. With this jolly reply, “no question are simple”, he attended my tiny and simple questions with enthusiasm. He even took care of scientific relation, marketing of your research etc. I thank Domenico Lahaye for all his personal help and professional supervision.

More special thanks to Kees Vulk I wish to express my profound thanks to Kees for providing me the opportunity to work at Delft in numerical analysis group under

his own guidance. He has been wonderful guide and research troubleshooter. He always took favorable view of things. He provided great balance in his supervision. He has been amazingly responsive supervisor. Many times, he responded with his valuable feedbacks within acute deadlines. Apart from numerical analysis, I learnt a lot of professionalism from him. I owe Kees for what I am in academics today.

ACKNOWLEDGEMENTS A number of people indirectly contributed the work in this thesis. First and foremost my very attentive parents, Ammi and Abbu. They did a lot for their kids. They cherished their kids thoroughly. For their being preoccupied for better education, work out for that with available resources, for their keen attention in all my problems, cannot be thanked in words. I am also indebted to my grandpa, baba, for letting me know the meaning of life. It surpasses my articulation skills to thank these people in a befitting manner. I also wish to thank my siblings, who have been there to make my life cheerful. I am indebted to Dr. Asif ali for being inspiration and for believing me and making me believe in myself too.

Numerical analysis research is unimaginable without computing facilities. During my PhD, I have experienced quality computing facilities and technical support. Kees Lemmens takes all the credits for this. I also thank Franca Post and Dr. Wim Horssen for their logistic help in many official matters. I also thank secretaries Mechteld, Deborah and Evelyn, they were always efficient to help out in department's local problems.

In the department, I learnt a lot things from different colleagues. I enjoyed working with native and foreign colleagues, which include Mehfooz, Bowen, Fred, Jennifer, Fons, Duncan, Ibrahim, Pavel, Reijer, Dennis, Rohit, Xiaozhou, Jing, Guido, Manuel, Thea, Piyao and others. An special mention to my colleagues, with whom i chanced to share the office. These include Ibrahim and Mehfooz in beginning, later these were Fahim, Behrouz, and Reinaldo. The translation of the summary (sammenvatting) and proposition (in Dutch), is a personal favour by colleagues and friends Dave and Joost. I shall take this opportunity to thank all these friends for their help and the pleasant working environment they created as a group, working in which was a lot of enjoyment and pleasure.

In the Netherlands, I met many Pakistani friends. Some already moved and some are still here. They helped to relive and their company provided home-like feelings. Number is beyond to pen all the names, though who are still around to make my weekends refreshing includes Akram, Umer Chaudhry, Nomi bhai, Tabish, Imran, Usama, Dev, Rajab, Hyder and Nick (he is almost Pakistani).

Finally, I thank the members of the committee, Prof. Ira, Prof. Wim, and Prof. Reinard and others for devoting some of their precious time to scrutinize my thesis. Thank you all very much for your time and interest.

Abdul Hanan Sheikh
Delft. October 19, 2014

Summary

The Helmholtz equation is the simplest possible model for the wave propagation. Perhaps this is the reason, despite denying traditional iterative methods like Krylov subspace methods, Multigrids, etcetera, numerical solution of the Helmholtz equation has been an interesting and abundant problem to researchers since years. The work in this dissertation is also classified as an attempt to develop fast and robust iterative methods for the solution of the Helmholtz equation. This works is specified for applications in seismic imaging-Geophysics, where usually high frequency are used. Thus we will be targeting large wavenumber Helmholtz problems.

The finite difference discretization of the Helmholtz equation with typically given Absorbing (Sommerfeld) boundary conditions gives rise to symmetric, non-Hermitian, indefinite linear systems. Resolution of large wavenumber requires larger number of grid points, thus large linear systems. Many (sparse) direct solvers and hybrid (direct and iterative) solvers have been proposed, but it is quite obvious for very large problems that (sparse) direct solvers have been too much depending upon memory, which makes them less acceptable. Quite a lot of work has been invested in researching iterative solution methods for the Helmholtz equation since many decades. The indefiniteness, which increases with respect to an increase in the wavenumber, poses extra problems for iterative solvers and robust solution of indefinite (large) linear system forms an important research activity. Many iterative techniques like domain decomposition methods, multigrid methods and preconditioners for Krylov subspace methods have been proposed but non of them has been quite robust.

For multigrid methods, indefiniteness arises difficulties in having both good smoothing property and constructing appropriate coarse-grid approximations of the problem, which are responsible for further reduction of low frequency errors. Many attempts have been spent in algebraic variants of multigrid methods. Some of them works well with limitation of homogeneity. Most of them fails to show satisfactory con-

vergence. The same holds for Krylov subspace methods. One of the difficulties for Krylov methods is to find a cheap and performing preconditioner for the indefinite Helmholtz equation. An overview of preconditioners, ranging from classical to matrix based, for indefinite Helmholtz linear system has been give in this thesis. A matrix-based complex shifted Laplace preconditioner (CSLP) has been seen as best in the available ones. However, with increasing wavenumbers CSLP shows a slow convergence behavior. We address this issue continuing using CSLP while taking care of its requirement of specific complex shifts.

The projection-type preconditioners have been widely investigated by researchers in numerical analysis community. We propose the projection-type deflation preconditioner to tackle the near-singular nodes, which are the cause of the decay the convergence of, this otherwise well performing, CSLP. Like multigrid, this deflation preconditioner, named as ADEF1, requires to solve coarse problems at different coarser levels. An optimized algorithm has been tested and proposed suggesting iterative solution of coarse problems at different levels. This finalizes as a multilevel preconditioner. The re-discretization coarsening strategy that we propose and investigate in this thesis is aimed at reducing the memory size and maintaining stencil size. The multilevel Krylov method (MLKM) has also been investigated and compared with its counterpart ADEF1.

The rigorous Fourier analysis (RFA) to investigate the convergence of iterative methods forms a separate research theme, which is included in the thesis. We analyse the proposed multilevel preconditioners ADEF1 and MLKM for two-levels. Analysis shows spectral behavior of the preconditioner, which can be taken as favorable for Krylov methods. RFA points out near-singular modes and highlights their contribution in prevailing stagnation. Further the convergence can be enhanced by adapting coarse grid operator at different levels.

The proposed preconditioners have been tested on academic as well as the benchmark Marmousi problem. A huge reduction in number of iterations can be noticed. A comparison in the amount of iterations and solve time, specially for three-dimensional problem, shows that the invested work has paid-off. Proposed preconditioners has been uniformly performing for one- to three-dimensions as well as for heterogeneous medium problems.

Samenvatting

De Helmholtzvergelijking is het meest eenvoudige model voor het transport van golven. Wellicht is dit de reden dat, ondanks het uitsluiten van traditionele iteratieve oplossingsmethoden als Krylovdeelruimtemethodes, multigridmethodes, enzovoorts, het numeriek oplossen van de Helmholtzvergelijking al jarenlang een interessant en veelvoorkomend probleem voor onderzoekers is. Dit werk is ook een poging om snelle en robuuste iteratieve methodes te ontwikkelen voor het oplossen van de Helmholtzvergelijking. Dit werk richt zich op toepassingen in seismische beeldvorming in geofysica, waar doorgaans hoge frequenties worden toegepast. Daarom zullen wij ons richten op Helmholtzproblemen met een hoog golfgetal.

De eindige differentie discretisatie van de Helmholtzvergelijking met absorberende (Sommerfeld) randvoorwaarden leidt tot symmetrische, niet-Hermitische, indefiniete stelsels lineaire vergelijkingen. Het werken met grote golfgetallen vereist een groter aantal roosterpunten, en dus grotere lineaire stelsels. Vele directe solvers (voor ijle matrices) en hybride solvers (direct en iteratief) zijn voorgesteld, maar het is duidelijk dat voor zeer grote problemen directe solvers te veel geheugen vereisen, waardoor zij ongeschikt zijn. Al meerdere decennia lang wordt er veel tijd besteed aan onderzoek naar iteratieve oplossingsmethoden voor de Helmholtzvergelijking. De mate van indefinietheid, die toeneemt bij een toename van het golfgetal, geeft extra problemen voor iteratieve solvers. Robuuste oplossingsmethoden voor (grote) indefiniete lineaire stelsels is een belangrijk onderzoeksonderwerp. Veel iteratieve methodes zijn voorgesteld: domeindecompositiemethodes, multigridmethodes en preconditioners voor Krylovdeelruimtemethodes. Geen van deze methodes is robuust genoeg.

Voor multigridmethodes geeft indefinietheid problemen, zowel in het hebben van goede damping van snelle componenten, als bij het construeren van grofroosterbenaderingen van het probleem, die verantwoordelijk zijn voor het verder verminderen van lagefrequentiefouten. Veel pogingen zijn gedaan met een algebraïsche variant van de multigridmethode. Enkele hiervan werken goed onder de restrictie van homogeniteit, maar de meeste convergeren niet snel genoeg. Dit is ook het geval bij Krylovdeel-

ruimtemethodes. Eén van de moeilijkheden is het vinden van een werkende, goedkope preconditioner voor de indefiniete Helmholtzvergelijking. Een overzicht van preconditioners voor de indefiniete Helmholtzvergelijking, variërend van klassieke tot op matrix gebaseerde methodes, is gegeven in dit proefschrift. Een op matrix gebaseerde *complex shifted Laplace preconditioner* (CSLP) wordt gezien als de beste beschikbare methode. Echter, met een toename in het golfgetal, vertraagt de convergentie van CLSP. We behandelen dit probleem door CLSP te gebruiken en extra aandacht aan de voorwaarden voor specifieke complexe verschuivingen te geven.

De op projecties gebaseerde preconditioners zijn uitgebreid onderzocht door de onderzoekers uit de numerieke analyse. Wij stellen een op projecties gebaseerde deflatiepreconditioner voor om de bijna singuliere punten aan te pakken, die de oorzaak zijn van de teruglopende convergentie van de normaal gesproken goed werkende CSLP. Zoals in multigridmethodes, worden bij deze deflatiepreconditioner, genaamd ADEF1, problemen op verschillende, grovere roosters opgelost. Een geoptimaliseerd algoritme, waarbij het probleem iteratief op verschillende niveaus wordt opgelost, is getest en voorgesteld. De herdiscretisatie-vergroffingsstrategie die we voorstellen en onderzoeken in dit proefschrift is gericht op het verminderen van het geheugengebruik en het behoud van de stencilgrootte. Ook hebben wij de *multilevel Krylov method* (MLKM) onderzocht en vergeleken met ADEF1.

De *rigorous Fourier analysis* (RFA), om de convergentie van iteratieve methodes te onderzoeken, vormt een afzonderlijk onderzoeksthema en is ook behandeld in dit proefschrift. We analyseren de voorgestelde *multilevel preconditioners*, ADEF1 en MLKM, op twee niveaus. Analyse laat spectraal gedrag zien, dat als voordelig voor Krylov methodes gezien kan worden. RFA wijst de bijna singuliere eigenwaarde aan en geeft hun bijdrage weer aan de aanhoudende stagnatie. De convergentie kan verder verbeterd worden door een grofroosteroperator toe te passen op verschillende niveaus.

De voorgestelde preconditioners zijn getest op zowel academische problemen als het Marmousi-testprobleem. Een zeer grote vermindering in het aantal iteraties is te zien. Een vergelijking van het aantal benodigde iteraties en oplossingstijden, specifiek voor driedimensionale problemen, toont aan dat ons werk zijn vruchten heeft afgeworpen. De voorgestelde preconditioner presteert uniform voor een- tot driedimensionale problemen, en voor problemen met een heterogeen medium.

Contents

Preface	v
Summary	vii
Samenvatting	ix
1 Introduction	1
1.1 Background and Related Work	2
1.2 Aim of Research	4
1.3 ADEF1 preconditioner	5
1.4 Principal Findings	5
1.5 Thesis Organization	5
2 Model Problems and Their Discretization	7
2.1 One-Dimensional Problem	8
2.2 Two-Dimensional Constant Wave Number Problem	9
2.3 Two-Dimensional Non-Constant Wave Number Problem	9
2.4 Three-Dimensional Constant and Non-Constant Helmholtz Problem	11
2.5 Discretization and Pollution in Wave Propagation	12
2.6 Spectrum of the Helmholtz operator	16
3 Preconditioner Survey	19
3.1 Preconditioners for Helmholtz: Survey	19
3.2 Standard Preconditioners for Helmholtz	20
3.3 Historic Development of the Shifted Laplace Preconditioner	21
3.4 Multigrid approximation of CSLP	23
3.5 Spectrum of the CSLP/Convergence Obstacles	24

3.6	Down-sides: Need of Projection	25
4	Deflation	27
4.1	Deflation as Preconditioner	27
4.2	Variants of Deflation Preconditioner	28
4.2.1	Deflation using the CSLP Preconditioned Operator: Idealized Variant (V1)	29
4.2.2	Deflation using the CSLP Preconditioned Operator: Practical Variant (V2)	29
4.2.3	Deflation using the Helmholtz Operator (V3)	30
4.2.4	Deflation using the Complex Shifted Helmholtz Operator (V4)	30
4.2.5	Deflation using Re-discretization (V5)	30
4.3	Shifting Deflated Spectrum	31
4.4	Choice of Deflation Vectors	32
4.4.1	Deflation Vectors for Helmholtz	33
4.5	Implementation	35
4.6	ADEF1 preconditioner	36
4.6.1	ADEF1 Multilevel	37
4.6.2	ADEF1 Multilevel Algorithm Implementation	39
4.7	Two-Level Krylov Method (TLKM)	40
4.7.1	Multi-Level Krylov Method (MLKM)	42
4.7.2	Coarse Grid Operator Complexity	42
4.7.3	MLKM: Multilevel Implementation	43
4.8	Complexity Analysis	44
4.9	Concluding Remarks	44
5	Spectral Analysis	49
5.1	Rigorous Fourier Analysis Framework	49
5.2	One-Dimensional Rigorous Two-Grid Analysis of ADEF1	50
5.2.1	Basis Diagonalization	51
5.2.2	Closed-form Expression for Eigenvalues	55
5.3	ADEF1 Simplified Analysis	58
5.4	Two-Dimensional Rigorous Two-Grid Analysis of ADEF1	62
5.4.1	Basis Diagonalization	62
5.5	One-Dimensional Rigorous Two-Grid Analysis of TLKM	65
5.5.1	Basis Diagonalization	67
5.5.2	Closed-form Expression for Eigenvalues	69
5.6	TLKM Simplified Analysis	73
5.7	TLKM Practical Variant	74
5.8	A Comparison: ADEF1 and TLKM	77
5.9	Re-Discretized Coarse Grid Operator	79

6	Numerical Experiments	83
6.1	One-Dimensional Constant Wave Number Problem	83
6.2	Two-Dimensional Constant Wave Number Problem	84
6.3	Shifts in SLP	86
6.4	Two-Dimensional Non-Constant Wave Number Problem	87
6.4.1	Wedge Problem	87
6.4.2	Marmousi Problem	88
6.5	Three-Dimensional Constant Wave Number Problem	90
6.6	Three-Dimensional Non-Constant Wave Number Problem	92
6.7	Finite Element Discretization	94
6.7.1	Algebraic Deflation Vectors	94
6.7.2	Performance of Algebraically coarsened FEM discretization	94
7	Conclusions and Future Work	99
7.1	Conclusions	99
7.2	Further Implications	101
7.3	Future Work/Outlook	102
 Appendices		
A	ADEF1 Implementation	103
A.1	Introduction	103
A.2	Constructing DATA files	103
A.2.1	How to adapt	104
A.3	Solving part of the software	104
B	Investigation of the behavior of small eigenvalues of ADEF1 with perturbed eigenvectors	107
B.1	Introduction	107
B.2	Definition of the problem	107
B.3	ADEF1 deflation method	108
B.4	Analysis	108
B.5	Numerical experiments	109
B.6	Matlab Code	111
	 Curriculum Vitae	 123

List of Tables

3.1	Number of GMRES iterations without preconditioner, with $ILU(0)$ and $ILU(0.01)$ preconditioners.	21
3.2	A comparison of the number of iterations for ILU type preconditioners and CSLP preconditioners using different shifts for various wave numbers.	22
4.1	Costs per application of a two-level preconditioner in one Krylov iteration.	45
6.1	GMRES iterations preconditioned by Two-Level deflation preconditioner applied on one a dimensional Helmholtz problem with Sommerfeld boundary conditions.	84
6.2	Number of GMRES iterations for two dimensional constant wavenumber problem with Dirichlet boundary conditions for various wave numbers and grid resolutions using the SLP preconditioner $M_{h,(1,0.5)}$ with and without deflation.	85
6.3	Number of GMRES iterations for two dimensional constant wavenumber problem with Sommerfeld boundary conditions for various wave numbers and grid resolutions using the SLP preconditioner $M_{h,(1,0.5)}$ with and without deflation.	86
6.4	Number of outer GCR iterations preconditioner by ADEF-1 for the two-dimensional problem with Sommerfeld boundary conditions for various wave numbers and $kh = 0.3125$ using the shifts $(\beta_1, \beta_2) = (1, 1)$	87
6.5	Number of $M_{h,(1,0.5)}$ preconditioned GMRES iterations with and without deflation for Problem (2) for various wave numbers and grid resolutions.	88
6.6	CSLP and ADEF1 performance comparison for Marmousi problem without damping i.e. $\alpha = 0$	89

6.7	CSLP and ADEF1 performance comparison for Marmousi problem without damping i.e. $\alpha = 0.05$	90
6.8	CSLP and ADEF-1 performance comparison for Marmousi problem without damping i.e. $\alpha = 0$ for Marmousi problem discretized with a mesh size equivalent of 20 grid points per wavelength.	90
6.9	PETSc Solve time and iteration comparison between Bi-CGSTAB preconditioned with CSLP-F and FGMRES(20) preconditioned with ADEF1-F	91
6.10	PETSc Solve time and iterations for three dimensional problem with wavelength resolved over 20 grid-points . Bi-CGSTAB has been preconditioned by CSLP-F, where flexible GMRES is preconditioned by ADEF1.	92
6.11	PETSc Solve time comparison between BiCGSTAB preconditioned with SLP-F and FGMRES(20) preconditioned with ADEF1-F. Discretization satisfies the relation $kh \leq 0.625$ i.e. 10 gp/wl	93
6.12	PETSc Solve time comparison between BiCGSTAB preconditioned with CSLP-F and FGMRES(20) preconditioned with ADEF1-F. Grid resolution is such that there are 20 gp/wl.	94
6.13	Comprehensive iteration count; FEM compared with FDM discretization with different solver types and difference wave numbers.	97
6.14	Comprehensive solve time; FEM compared with FDM discretization with different solver types and difference wave numbers.	97

List of Figures

2.1	The wedge problem introduced in [91].	10
2.2	The Marmousi problem introduced in [116].	12
2.3	The Marmousi problem introduced in [116].	13
2.4	Layered wave number distribution and point source location in three dimensional unit cube domain.	14
2.5	The trend of the increasing number of negative eigenvalues for the two-dimensional Helmholtz against the wavenumber. The Grid resolution is set such that there are 20 grid-points per wavelength. On the right, the ratio of the negative eigenvalues to the total eigenvalues is plotted.	17
3.1	Convergence factor for $M_h(0, \beta_2)$ with the Gauss Seidel smoother versus imaginary shift β_2	25
3.2	Convergence factor for $M_h(0, \beta_2)$ with the Jacobi smoother versus imaginary shift β_2	25
3.3	Spectrum of the $M_{h,(1,0.5)}^{-1}A_h$ for different values of the wave number k	26
4.1	Approximation of the coarse grid operator in Equation 4.1 disperse the eigenvalues in the vicinity of origin in the spectrum of the deflated preconditioned Helmholtz operator given on the left-hand side of Equation 4.17	32
4.2	A schematic representation of the two level ADEF1 algorithm considered.	39
4.3	Comparing iterations to solve one dimensional Helmholtz with two-level solvers ADEF1 and TLKM ideal and practical variants.	46
5.1	Non-zero part of $\sigma(B_{h,2h,(1,1)})$ in the one-dimensional problem for various wavenumbers k satisfying $\kappa = 0.3215$	56

5.2	Non-zero part of $\sigma(B_{h,2h,(1,1)})$ in the one-dimensional problem for $k = 2000$ and for various values of κ	57
5.3	Non-zero part of $\sigma(B_{h,2h,(1,\beta_2)})$ in the one-dimensional problem for $k = 2000$ and for various values of β_2 satisfying $\kappa = 0.625$	57
5.4	Magnitude of non-zero part of $\sigma(B_{h,2h,(1,1)})$ in the one-dimensional problem for $\kappa = 0.625$ and $\kappa = 0.3215$ and various values of k	58
5.5	Real part of eigenvalues of the deflated unpreconditioned operator $\widehat{B}_{h,2h,(\beta_1,\beta_2)}$ versus the index ℓ for various values of the wavenumber. The top and bottom show all and a selection of indexes ℓ close to the near-null space, respectively.	60
5.6	Real part of spectrum; ADEF1 preconditioner applied at \mathbf{A}_h . Wavenumber considered is $k = 10000$ with grid size is set such that there are 20 grid points per wavelength	61
5.7	Non-zero part of $\sigma(B_{h,2h,(1,1)})$ for the two-dimensional problem for various values of k satisfying $\kappa = 0.625$	64
5.8	Magnitude of the five eigenvalues smallest in size of $\sigma(B_{h,H,(1,1)})$ versus the wave number in the two-dimensional problem for $\kappa = 0.625$ (left) and $\kappa = 0.3215$ (right).	65
5.9	Magnitude of the non-zero part of $\sigma(B_{h,2h,(1,1)})$ in the two-dimensional problem for $\kappa = 0.625$ and $\kappa = 0.3215$ for various values of k	66
5.10	Spectrum of the TLKM ideal variant with CSLP for various wave numbers resolved over a grid with 20 gridpoint per wavelength.	70
5.11	Real part of the non-zero eigenvalues of $B_{h,2h,(\beta_1,\beta_2)}$ versus the index ℓ for TLKM ideal variant.	72
5.12	Real part of the non-zero eigenvalues of $B_{h,2h,(\beta_1,\beta_2)}$ versus the index ℓ for TLKM ideal variant.	72
5.13	Real part of the non-zero eigenvalues of $B_{h,2h,(\beta_1,\beta_2)}$ versus the index ℓ for TLKM ideal variant.	73
5.14	Real part of the spectrum; The TLKM ideal preconditioner applied to \mathbf{A}_h . Wavenumber considered is $k = 10000$ with grid size is set such that there are 20 grid points per wavelength	75
5.15	Eigenvalues of $B_{h,2h,(\beta_1,\beta_2)}$ in the complex plane considering TLKM practical variant without shift term Q_h for various values of the wavenumber.	77
5.16	Real part of the eigenvalues of $B_{h,2h,(\beta_1,\beta_2)}$ versus the index ℓ using the TLKM practical variant with shift term Q_h for various values of the wavenumber. The top and bottom show all and selection of indices ℓ close to the near-null space, respectively.	78
5.17	ADEF1; $\kappa = 0.3215$	79
5.18	MLKM Practical; $\kappa = 0.3215$	79
5.19	MLKM Ideal; $\kappa = 0.3215$	79

5.20	Spectrum of deflated operators using re-discretized coarse grid operator(a-d) and Galerkin coarse grid operator (e-f).	81
6.1	Number of $M_{h,(1,1)}$ preconditioned GMRES iterations with and without deflation for the one-dimensional constant wave number problem for k ranging between 10 and 800 (left) and between 1,000 and 20,000 (right).	85
6.2	Number of $M_{h,(1,\beta_2)}$ preconditioned GMRES iterations with and without deflation versus β_2 for Problem (1) for $k = 50$ and Problem (2) for $f = 30$, both problems with Sommerfeld boundary conditions.	88
6.3	CPU time per grid point in constant wavenumber problem using 10 (left) and 20 (right) grid points per wavenumber	92
6.4	Comparison of convergence history of CSLP and ADEF1 preconditioners for three dimensional problem with wave number $k = 20$ and $k = 40$ on left and right figures respectively.	93
B.1	Smallest eigenvalue of $P_{ADEF1}A$ as function of δ for various values of ϵ	110
B.2	The absolute value of the smallest eigenvalue of $P_{ADEF1}A$ as function of δ for various values of $\epsilon = -0.999910^{-4}$	110

List of Algorithms

1	Preconditioned Deflated GMRES for system $Au = b$	36
2	Implementation ADEF1 Preconditioner $P_{h,ADEF1} = M_{h,(\beta_1,\beta_2)}^{-1}P_h + Q_h$.	41
3	Implementation MLKM-P Preconditioner $P_{h,TLKM} = M_{h,(\beta_1,\beta_2)}^{-1}P_h + Q_h$	47

Introduction

This chapter serves as an introduction to the work presented in this thesis, which deals with the numerical solution of the Helmholtz equation. The efficient computational modeling of wave scattering is important for many practical problems. Time harmonic wave scattering phenomena finds applications in various scientific fields like acoustics, electromagnetism, atomic spectroscopy [16], radar and sonar technology, seismics [26] and recently it has got attention in medical imaging. The scattering problem treated in this thesis is a crucial part of the inverse problem, which is of greater importance in energy industry (seismic exploration) and is mathematically more challenging. The time harmonic wave scattering phenomena are modeled by means of Helmholtz equation in the frequency domain. Many applications of the Helmholtz equations involve unbounded domains. Dirichlet and Neumann boundary conditions are often used. However, Dirichlet boundary conditions truncates the physical domain and causes wave reflections. Numerical treatment of the Helmholtz equation requires that these wave reflections should be avoided. Also if Neumann boundary conditions are posed on all boundaries, the problem is not well posed and have non-unique solutions [67]. In order to make the problem well-posed, Sommerfeld boundary conditions are imposed, which were introduced in [38]. An alternative boundary condition: perfectly matched layers (PML) has been proposed in [12]. Methods for modeling time-harmonic wave scattering (in an inhomogeneous medium) typically involve applying finite difference or finite element techniques to the Helmholtz equation. These techniques give rise to the linear systems, where the coefficient matrix is sparse, symmetric, often complex valued non-Hermitian and large. For small wave numbers, the coefficient matrix is close to the matrix for the Laplace equation. For large wave numbers, the linear system is indefinite in a way that the coefficient matrix bears negative eigenvalues. The coefficient matrix of the linear system shows poor conditioning as the discretization is refined.

1.1 Background and Related Work

During the past few decades, the numerical methods for Helmholtz have been subject of active research. These methods involve solution of large and indefinite linear systems arising from a discretization of Helmholtz equation. Methods to solve linear systems can be classified as direct and iterative methods.

Direct methods can handle the linear system efficiently till a certain size. However they are not favorable for sparse linear systems. The zero elements in the matrix might be filled with non-zero elements in the elimination process. There are few direct methods, which take care of the sparsity [34, 32, 72]. They are also limited to sizes of the problem. For the indefinite and non symmetric matrices, pre-processing (pivoting, permutation and scaling) can be performed before the factorization in order to minimize the fill-in and improve the accuracy of the factorization [35]. These methods perform well in the one dimensional and two dimensional case [66]. However their memory requirements increase with the size of the problem, dropping-off their efficiency for three dimensional Helmholtz problems. A parallelization of sparse direct solver have also been tried. In [86], a three dimensional Helmholtz problem was numerical solved using MUMPS [5, 4]. It is reported that the solver requires too much memory.

Iterative methods are memory efficient, in general. They are attractive for sparse linear systems in an obvious way as they are designed for sparse matrices. Krylov methods are robust iterative methods. There are many Krylov methods specified by properties of the linear system. Indefiniteness limits the choice of Krylov methods for the Helmholtz equation. GMRES [99] and Bi-CGSTAB [115] are suitable choices for this system. Also the IDR [106] has been used recently in [10]. The performance of Krylov methods depends upon the well conditioning of the matrices. Usually preconditioners are applied in order to condition the matrices and to transform the spectrum of the linear system into a favorable one for Krylov methods. First, we describe some preconditioning techniques described in the literature for Helmholtz problems.

Incomplete factorizations (ILU) of the coefficient matrix (in a linear system) are simple and popular preconditioning techniques. A factorization preconditioner may however lead to unstable, highly ill-conditioned incomplete factors in the indefinite case. Some remedies have been proposed to treat the instability in context of the Helmholtz equation. In [56] a specific factorization is designed that aims at performing an analytic incomplete factorization (AILU). This approach has limitations to extend to the heterogeneous medium problem. Incomplete LU factorization with a threshold, which is introduced in [96], is recommended in [70] along with a finite element discretization of the Helmholtz operator. Finally, an other approach consists in performing an incomplete factorization of a complex shifted Helmholtz operator

as a preconditioner for the original Helmholtz problem [77]. However, the convergence of ILU preconditioned Krylov methods is found to be generally slow at high wavenumbers and storing the ILU factors may not be always affordable. Factorization based preconditioners depends on the wave number as well as mesh size [102]. Furthermore it is recognized that ILU methods are difficult to parallelize [65]. Another important class of preconditioners relies on domain decomposition techniques which are proposed in [13, 51, 58, 8, 50, 51]. These methods solve the original problem by splitting the physical domain into smaller sub-domains where the solution of the local problems is affordable with direct methods. The correction in the coarse problem (in sub-domain) give a convergence rate independent of the number of sub-domains. Due to the indefiniteness at high wavenumbers, Helmholtz type problems are challenging for domain decomposition methods for some reasons. First in order to be effective, a large number of coarse problems has to be considered. Secondly local problems may be close to singular. Further details on domain decomposition for Helmholtz are discussed in [58, 119]. Recent work on domain decomposition for Helmholtz can be found in [117, 28].

Multigrid methods [18, 22, 63, 107, 113] have also been considered as solvers as well as preconditioners for Helmholtz problems. Nevertheless they also encounter difficulties to cope with highly indefinite problems. Regarding Helmholtz problems, classical multigrid ingredients such as standard smoothing and coarse grid correction are found to be ineffective [21, 37, 46, 48]. First, smoothers cannot smooth error components on the intermediate grids. Second, the wavenumber in the discrete Helmholtz operator makes its approximations poor on coarse meshes, the effect of the coarse grid correction being then deteriorated. In [33, 76, 71], strategies have been proposed to adapt the multigrid technique for the solution of Helmholtz problems. A first strategy consists of the use of few grids in the hierarchy of the multigrid preconditioner [33, 71] such that the grid approximation is effective on the considered grids. If more than two grids are considered, non-standard smoothers (Krylov based such as GMRES) on the coarser levels can be used to avoid the ineffectiveness of the standard smoothers on intermediate grids [37]. However, in three dimensions, a reduced number of grids in the multigrid hierarchy could lead to a coarse problem whose factorization can be problematic in terms of computational resources. A second approach is to solve Helmholtz problems with a wave-ray multigrid algorithm [19]. These methods are based on two representations of the error on the coarse grids of the hierarchy. These representations enable then both the smoother and coarse grid corrections to be efficient. Further developments have been reported in [75]. This method performs well in the homogeneous case, but in the case of heterogeneous medium, ray functions must be computed. This has been reported in [74]. The method yet has been performed on one dimensional problem. Few alternative methods can be found in [9, 85, 37, 62, 39, 49].

Operator based multigrid-preconditioners also has been in use since long for the indefinite Helmholtz problem. Earlier presentation of these preconditioners can be found in [11] and [73] in which a Laplace operator and a Laplace operator with a real shift, respectively, are proposed. Both preconditioners lead to good results for medium size wave numbers. For large wave numbers numerical results on the contrary show a steep increase in the number of iterations. The paper [77] can be seen as pioneering paper on shifted Laplace preconditioners, in which incomplete LU decompositions of a shifted Laplace operator are used as a preconditioner. However for the first time, a preconditioner based on original with a *complex* shift has been reported in [43]. This class of Laplace preconditioners with a complex shift proposed and further studied in [43, 46, 47], where the solver requires a number of iterations that grows only linearly as the wave number increases. Inspired by this work, a number of generalizations appeared shortly afterwards in [42, 93, 17, 1, 44] together with applications in different industrial contexts in [120, 3, 94, 89, 90, 114, 2, 87]. The real and imaginary shifts in these type of preconditioners determine the performance of the solver heavily. Therefore this attracts people to work on optimal choice of the shifts in preconditioner. Related work is reported in [30, 54] Some more developments about complex shifted Laplace preconditioners are given in [93, 24, 23]. The convergence of the shifted Laplace preconditioners is spectrally analyzed in [59, 29] extensively. This analysis shows that the smallest eigenvalues of the preconditioned operator rush to zero as the wave number increases. This explains the non-scalability of the complex shifted Laplace preconditioner (CSLP). Despite of the drawback revealed via analysis in [59], CSLP has outperformed the existing preconditioners (for the Helmholtz)of the era.

1.2 Aim of Research

The basic approach of isolating the troublesome aspects of the complicated problems into small and easier ones, solve the latter, and then using these solutions in various ways to solve the original complicated problem, has been employed since early times. No doubt the CSLP preconditioner shows robustness. However it tends to encounter small eigenvalues, which stagnates the convergence. Inspired of the robustness of CSLP, it is naturally to attend the local issue of small eigenvalues in CSLP preconditioned Helmholtz. Deflation has been used to treat a bunch of unwanted eigenvalues. Deflation type methods have been reported for many problems in literature. Specific for Helmholtz , in [45, 42], deflation has been proposed in combination with CSLP. As deflation vectors the columns of the bilinear interpolation operator from coarse to fine grid are used. The deflation can be seen as a second level preconditioner that attempts to remove small eigenvalues. The resulting method is quite involved and somewhat complicated to implement.

The aim of this work is to design an attractive and simple to implement multilevel deflation based solver, utilizing the robustness of CSLP. Analysis of this solver is also performed. The ADEF1 preconditioner has been proposed. ADEF1 is a simple deflation based preconditioner, which is constructed using the Helmholtz operator. It is also combined with a CSLP preconditioner. ADEF1 allows flexible shifts in CSLP, which further renders a better approximation of CSLP preconditioner with the standard multigrid components. An acceptable and competitive numerical solution to indefinite Helmholtz equation has been accomplished with the ADEF1 preconditioner.

1.3 ADEF1 preconditioner

In order to solve large linear systems, it is necessary to accumulate all factors like solver, preconditioning, implementation of algorithms on commonly available hardware (computational resources) etc. The ADEF1 preconditioner is used as a multilevel preconditioner in practice. It involves the hierarchy of meshes like multigrid. Implementation has been simplified. Despite of the complexity of this multilevel preconditioner, it is justified by its performance demonstration. In number of iterations, it seems parameter independent, which favors the solve time in comparison with other preconditioner(s).

1.4 Principal Findings

We also analyze the proposed ADEF1 preconditioner via rigorous Fourier analysis. In order to present conclusive findings from analysis, we also analyze the variants of deflation preconditioner defined in [44] and other, which involve a re-discretized coarse grid operator and shifted coarse grid operator. Spectral formulae are derived which helps to consider the performance behavior of ADEF1. Graphical presentation highlights pros and cons of CSLP, ADEF1, and other deflation preconditioners. ADEF1 transforms the spectrum of the Helmholtz operator into a cluster around 1. Special care has been recommended to be taken for dealing with the coarse grid operator.

1.5 Thesis Organization

The thesis is divided into the following chapters.

- In Chapter 2, the model problems and its importance in scientific fields are discussed. Finite difference discretization for the Helmholtz equation in one, two and three dimensions is detailed. In the final part of this chapter, spectral properties of the discretized Helmholtz operator are given.

- The solvers for Helmholtz, history of preconditioners for Helmholtz, multigrid approximation of CSLP preconditioners are discussed in Chapter 3. Chapter 3 also includes a discussion about a number of drawbacks of CSLP.
- Chapter 4 gives details about deflation. It starts to define deflation in general and later specific for the Helmholtz operator. Deflation related topics covered in this chapter are deflation vectors, deflation vectors for Helmholtz, deflation in combination with CSLP, variants of deflation for Helmholtz, need of multilevel approach etc. Implementation has been explored in this chapter too.
- Chapter 5 deals with the analysis of the deflation variants presented in the earlier chapter. Graphical presentation of the analysis results as well as theoretical formulae makes the chapter worthwhile.
- The findings from analysis of our proposed ADEF1 preconditioner are confirmed by numerical results, which are presented in Chapter 6. Numerical experiments include one, two, and three dimensional problems. Results are also compared with various solver types.
- Chapter 7 is devoted to conclusions of the work with a recommendation of future research.

Model Problems and Their Discretization

The simulation of waves are of great importance in various engineering areas like acoustics, electromagnetic, wireless technology and geophysical seismic imaging. The physical phenomenon of steady-state wave propagation over a physical domain Ω is modeled by means of the Helmholtz equation

$$-\Delta u(\mathbf{x}) - k^2(\mathbf{x}) = f(\mathbf{x}), \quad (2.1)$$

where $u(\mathbf{x})$ is the wave field, $k(\mathbf{x})$ the wave number and \mathbf{x} the spatial variable. The wave number can be related with wavelength λ by following expression

$$k(\mathbf{x}) = \frac{2\pi}{\lambda}, \quad (2.2)$$

where $\lambda = \frac{\omega}{c(\mathbf{x})}$ with $\omega = 2\pi f$ the angular frequency and $c(\mathbf{x})$ the phase velocity. The wave number $k(\mathbf{x})$ can be written as $k(\mathbf{x}) = \frac{2\pi f}{c(\mathbf{x})}$.

Equation (2.1) has to be provided with boundary conditions on the boundary $\partial\Omega$ to ensure its well-posedness. Physically, either the wave propagates till an indefinite distance or is scattered by obstacles in the medium. The boundary conditions, vanishing the wave field at boundaries, can be described by homogeneous Dirichlet conditions

$$u(\mathbf{x}) = 0, \quad (2.3)$$

and the reflecting ones by Neumann conditions

$$\frac{\partial u}{\partial \eta} = 0, \quad (2.4)$$

where η is the *outward directional normal*.

This thesis aims to target the application in seismic wave migration in subsoil, in which a source emits the waves from a certain point on the surface of the Earth with given frequencies. The pressure wave then travels through the layers of earth and propagates back to the surface due to heterogeneity in the physical domain. The pressure wave is recorded by several receivers. This simulation is used to reproduce the wave propagation through the heterogeneous medium. This helps to generate an expository map of the bed of earth. Thus location and layers' thickness can be detected. This frequency domain problem needs to be solved and is of our interest. Subsequently, Fourier transformation converts the frequency domain solution to time domain solution. Time domain solution predicts the structure of the earth for oil exploration.

The wave scattering occurs in many applications in a medium with an unbounded domain. To ensure the numerical solution on a finite domain, the computations on an unbounded domain are avoided by imposing artificial boundary conditions. Non-homogeneous Neumann boundary conditions of the type

$$\frac{\partial u(\mathbf{x})}{\partial \eta} = \iota k u(\mathbf{x}), \quad (2.5)$$

models the absorbing layers on the boundary of domain. These boundary conditions are also called Sommerfeld radiation boundary conditions [38]. In [14], the perfectly matched layer (PML) method is reported for electromagnetic waves. In the application mentioned, the harmonic point source located at \mathbf{x}_s is described by the standard delta function $f(\mathbf{x})$

$$f(\mathbf{x}) = \delta(\mathbf{x} - \mathbf{x}_s). \quad (2.6)$$

The problems used throughout this thesis and their motivation is given in the next sections.

2.1 One-Dimensional Problem

On the standard *unit* domain $(0, 1)$, the Helmholtz equation in 1D with constant wave number k reads as

$$-\Delta u(x) - k^2 u(x) = f(x), \quad (2.7)$$

equipped with Dirichlet boundary conditions at both ends of the domain

$$u(x) = 0, \quad (2.8)$$

or Sommerfeld radiation boundary conditions at both ends of the domain i.e. at boundary $\partial\Omega$

$$\frac{\partial u(x)}{\partial \eta} - \iota k(x)u = 0. \quad (2.9)$$

The one-dimensional problem with earlier boundary conditions serves as the model to get theoretical insights into the solvers using rigorous Fourier analysis (RFA). We refer to Section 5.2. While the large wave number one dimensional problem with Sommerfeld boundary conditions 2.9 is used as test problem for different solvers.

2.2 Two-Dimensional Constant Wave Number Problem

The Helmholtz equation on domain $\Omega = (0, 1) \times (0, 1)$ reads,

$$-\Delta u(x, y) - k^2 u(x, y) = f(x, y). \quad (2.10)$$

We assume constant k and the domain to be bounded by the absorbing layers modeled by Sommerfeld boundary conditions on boundary $\partial\Omega$

$$\frac{\partial u(x, y)}{\partial \eta} - iku(x, y) = 0. \quad (2.11)$$

For this problem, the source functions $f(x, y)$ is chosen as

$$f(x, y) = \delta\left(x - \frac{1}{2}, y - \frac{1}{2}\right), \quad (2.12)$$

meaning that the waves propagate from the center of the domain *outwards*. This problem with a very large wave number is used as a test problem for numerical experiments.

Equation (2.10) along with the source function defined in Equation (2.12) supplied with homogeneous Dirichlet boundary conditions

$$u(x, y) = 0 \quad (2.13)$$

on boundary $\partial\Omega$, is considered for analysis purposes.

2.3 Two-Dimensional Non-Constant Wave Number Problem

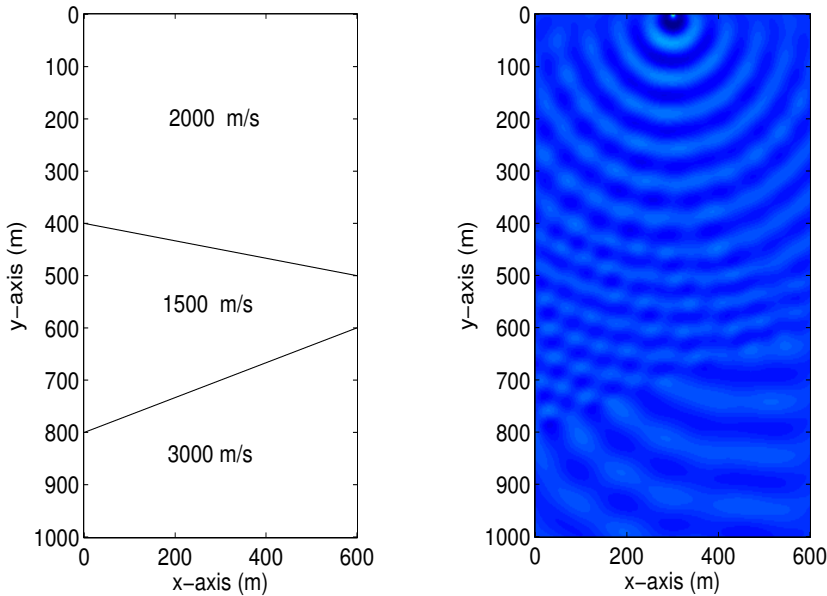
In physical problems from geophysical seismic imaging, heterogeneity appears. This leads to a contrast in wave number. The first problem with an inhomogeneous medium is a three layered so-called *Wedge problem* introduced in [91], in which the domain $\Omega = (0, 600) \times (0, 1000)$ is subdivided into three layers with different velocity hence difference wave number. Velocity distribution over layers is shown in Figure 2.1(a). In each layer, the velocity c is constant with the value shown. A point source defined by Equation (2.6) is located at $x_s = 300$ and $y_s = 0$, the resulting wave diffraction

pattern is shown in Figure 2.1(b).

With the Sommerfeld radiation boundary conditions Equation (2.5), the problem reads

$$\left. \begin{aligned} -\Delta u(x, y) - k(x, y)^2 u(x, y) &= f(x, y), \quad \text{on } \Omega = (0, 600) \times (0, 1000) \\ f(x, y) &= \delta(x - 300, y) \quad x, y \in \Omega, \end{aligned} \right\} \quad (2.14)$$

where $k(x, y) = \frac{2\pi \text{freq}}{c(x, y)}$ is given in terms of velocity as shown in Figure 2.1(a). In this thesis, a set of five different frequencies 10, 20, 40 and 80 Hz is considered for numerical tests.



(a) Domain of the wedge problem with velocity distribution over layers. (b) Pattern of waves diffraction through layers of different velocity.

Figure 2.1: The wedge problem introduced in [91].

The next problem mimics an industrial model problem and is called a Marmousi problem. This is a well known benchmark problem, used in many papers. The geometry of the problem is based on a profile through the North Quenguela through in Cuanza basin. It is an in-depth-image of the earth with 158 horizontal layers, making it highly heterogeneous. The original Marmousi data-set [116] is used for a two-dimensional Helmholtz problem with a range of velocities from 1500 m/s to 5500 m/s distributed over 158 layers in a rectangular domain $\Omega = (0, 9200) \times (0, 3000)$.

Conveniently, we consider some adaptation in the original problem; the original domain (3000×9200) has been trimmed into $\Omega_h = (2048 \times 8192)$ allowing geometrical coarsening of the discrete velocity data in an uncomplicated way, as the domain remains in powers of 2. The original velocity has also been adapted to make it less contrasted and the velocity $c(x, y)$ range is $2587.5 \leq c \leq 3325$.

Grid is resolved for different frequencies such that for maximum wavenumber k , $kh \leq 0.039$ for $freq = 1$ and $kh \leq 0.39$ for $freq = 10, 20, \text{ and } 40$. Thus we have unnecessarily large number of grid points per wave length for the problem with $freq = 1$ and approximately 15 grid points per wave length for the rest of frequencies $freq = 10, 20, \text{ and } 40$.

With the description given above, the problem reads

$$\left. \begin{aligned} -\Delta u(x, y) - k(x, y)^2 u(x, y) &= f(x, y), \quad \text{on } \Omega = (0, 8192) \times (0, 2048) \\ f(x, y) &= \delta(x - 4000, y) \quad x, y \in \Omega, \end{aligned} \right\} \quad (2.15)$$

where $k(x, y) = \frac{2\pi f}{c(x, y)}$ is generated in terms of the velocity function $c(x, y)$ over the domain Ω . The numerical experiments has been performed with all four different frequencies $f = 1, 10, 20, \text{ and } 40$ Hz. The range of velocities is shown in Figure 2.3(a). The Sommerfeld radiation boundary condition, given in Equation (2.5), is supplied on all sides of the domain.

2.4 Three-Dimensional Constant and Non-Constant Helmholtz Problem

Three-dimensional problems are mimic industrial problems. We will also solve these practical problems with our proposed solvers. For this, we consider the Helmholtz equation in a simple three-dimensional domain; the unit cube $\Omega = (0, 1)^3$ given by

$$\left. \begin{aligned} -\Delta u(x, y, z) - k(x, y, z)^2 u(x, y, z) &= f(x, y, z), \quad \text{on } \Omega = (0, 1)^3 \\ f(x, y, z) &= \delta(x - \frac{1}{2}, y - \frac{1}{2}, z - \frac{1}{2}) \quad x, y, z \in \Omega, \end{aligned} \right\} \quad (2.16)$$

Firstly, we consider a constant wave number throughout the domain.

Next the wave number is contrasted in three layers over the domain as shown in Figure (2.4). Wave numbers in bottom and top layers are scaled by $a = 1.2$ and $b = 1.5$ respectively. The point source location is set at $(\frac{1}{2}, \frac{1}{2}, 1)$ which allows wave propagation throughout the layers. For the space discretization, the step size h must satisfy the relation $k_2 h \leq 0.625$. The Sommerfeld boundary condition, presented in Equation (2.5), is imposed on all faces of the cube.

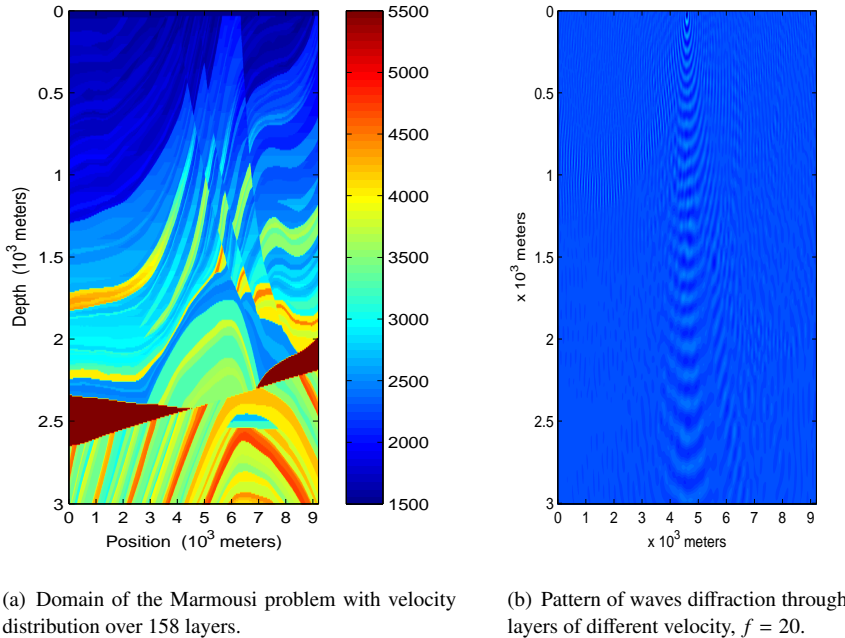


Figure 2.2: The Marmousi problem introduced in [116].

Remark 1. Few applications involve (acoustic) attenuation. For these type of applications, the Helmholtz equation is written as

$$-\Delta u(\mathbf{x}) - (1 - \alpha i)k^2(\mathbf{x})u(\mathbf{x}) = f(\mathbf{x}), \quad (2.17)$$

where $0 \leq \alpha \ll 1$ indicates the fraction of damping in the medium.

The “damped” Helmholtz equation (2.17) is generalized. Generally we will use the Helmholtz equation without damping as defined in perspective sections, however it will be mentioned exclusively if the damping term is incorporated.

2.5 Discretization and Pollution in Wave Propagation

A numerical solution of any (partial differential) equation involves the step of getting a discrete analogue of the given equation. The problem under consideration can be discretized by many methods including finite element methods (FEM), finite difference methods (FDM) and others. FEM methods with a discussion of the discretization accuracy are reported in [68, 6, 12]. The references [114, 53, 105] exploits a variety of

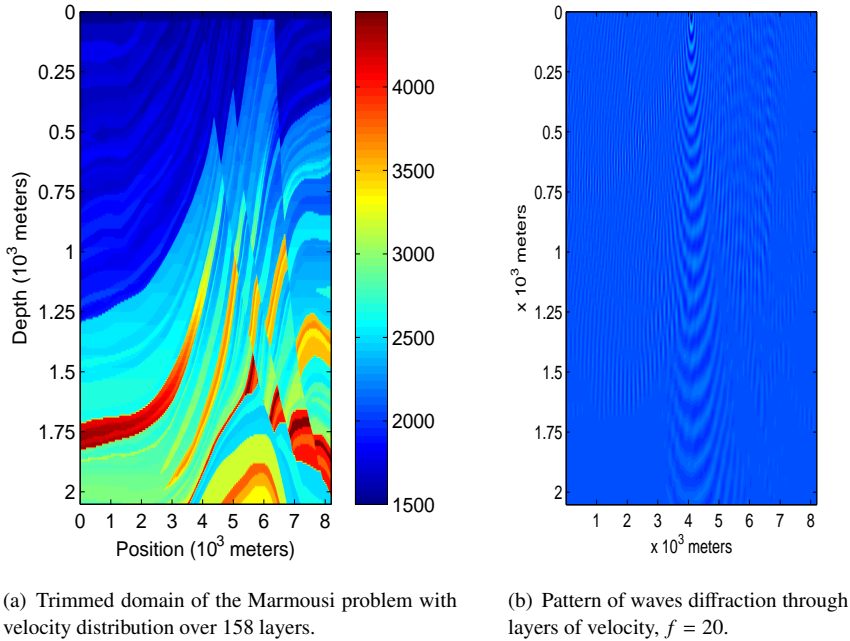


Figure 2.3: The Marmousi problem introduced in [116].

FDM discretizations. The problems with complex physical domains are commonly dealt with FEM, whereas FDM is well known for effortless discretization of simple geometries. The two important challenges faced by any discretization scheme for the Helmholtz problem are :

- Problems with large wave numbers have highly oscillatory solutions. Resolution immediately requires high refinement of the grid or the use of a higher order discretization method.
- In order to ensure the accuracy of the computed solution at very high wave numbers, as this solution is highly oscillatory, the grid is required to be more refined to capture oscillations. Suppose if “ h ” is the mesh size of discretization, this can be done by keeping kh to be small. However it has been observed in literature [12] and experiments that an increase in k deteriorates the accuracy even if kh is kept small enough. This indicates that the increase in grid size should be proportional to the characteristic of Helmholtz equation called wave number. To reduce the discretization error acceptably, the discretization should meet the stringent conditions such as imposing the condition $k^3 h^2 \leq \epsilon$ [12]. This problem is termed as *pollution* or nuisance in wave propagation.

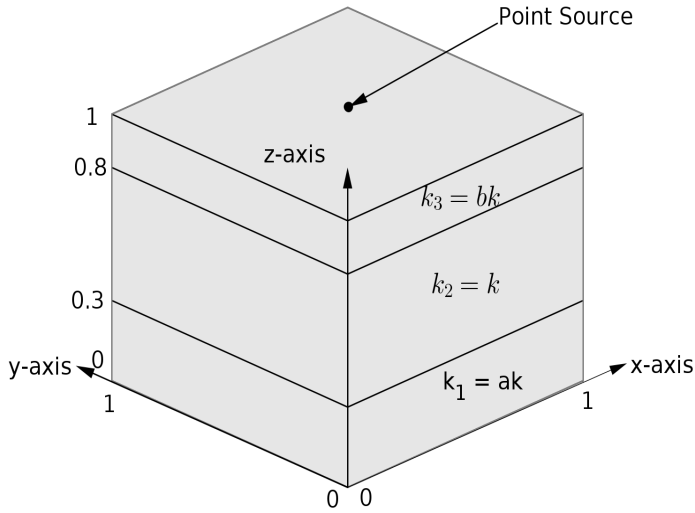


Figure 2.4: Layered wave number distribution and point source location in three dimensional unit cube domain.

The rule of thumb for the second order accurate $O(h^2)$ finite difference and linear FEM discretization is that at least 10 nodes per wavelength $\lambda = \frac{2\pi}{k}$ should be employed, which leads the restriction

$$kh \approx \frac{2\pi}{10} \approx 0.628. \quad (2.18)$$

However higher values of k imply shorter wavelengths and thus a fine grid to capture short wavelengths. If a certain domain encounters more wavelengths, the so-called *pollution* error accumulates. In some class of problems the accuracy might require more refined grids, say 20 or 30 grid points per wavelength (gp/wl) to overcome the pollution error [12]. For a given wave number k , we choose grid size h according to Condition (2.18) throughout this work, unless mentioned explicitly.

Having simple geometries in geophysical seismic imaging applications, second order FDM is used in this thesis. The discretization of the test problem with constant wave number Equation (2.10) in a unit square domain is explicated. Sommerfeld boundary conditions (2.11) are incorporated.

The finite difference discretization of the two dimensional problem with constant wave number

$$\left. \begin{aligned} -\Delta u(x, y) - k^2 u(x, y) &= f(x, y), \quad \text{on } \Omega = (0, 1) \times (0, 1) \\ f(x, y) &= \delta(x - \frac{1}{2}, y - \frac{1}{2}) \quad x, y \in \Omega, \end{aligned} \right\} \quad (2.19)$$

on a uniform mesh with mesh width $h_x = h_y = h$ in both x and y -direction yields the five point stencil

$$[A_h] = \frac{1}{h^2} \begin{bmatrix} & & -1 & & \\ & -1 & 4 - k^2 h^2 & -1 & \\ & & -1 & & \end{bmatrix}, \quad (2.20)$$

and vertex-center boundary elimination yields the stencil for lower-left grid point

$$[A_h] = \frac{1}{h^2} \begin{bmatrix} & & -2 & & \\ & 0 & 4 - k^2 h^2 & -2 & \\ & & 0 & & \end{bmatrix}. \quad (2.21)$$

If the grid is ordered lexicographically, the above stencil leads to a system of linear equations

$$A_h u_h = f_h, \quad (2.22)$$

where the discrete Helmholtz operator A_h is the sum of a stiffness matrix $-\Delta_h$, mass matrix $-k^2 I_h$ and boundary conditions matrix $-tkB_h$

$$A_h = -\Delta_h - k^2 I_h - tkB_h. \quad (2.23)$$

The FDM discretization of a three dimensional problem does not carry any complexities and is analogous to that of the two dimensional problem. It comes up as a 7-point stencil, as it is 5-point in the two dimensional case. The matrix in linear system (2.22) is complex-valued because of the Sommerfeld boundary conditions, indefinite in terms that eigenvalues of the matrix lies on both sides of the imaginary axis in the *complex plane*, symmetric but non-Hermitian, and ill conditioned.

Remark 2. Assume that the physical domain has a characteristic length \bar{l} and consider a non-dimensional domain $[0, 1]^3$. The non-dimensional length is determined as $\tilde{x} = x/\bar{l}$, $\tilde{y} = y/\bar{l}$ and $\tilde{z} = z/\bar{l}$. Thus

$$\frac{\partial}{\partial x} = \frac{1}{\bar{l}} \frac{\partial}{\partial \tilde{x}}, \quad \frac{\partial}{\partial y} = \frac{1}{\bar{l}} \frac{\partial}{\partial \tilde{y}}, \quad \text{and so on.}$$

Substituting these relations into Equation (2.17) results in

$$-\tilde{\Delta} u(\tilde{\mathbf{x}}) - (1 - \alpha i) \bar{k}^2(\mathbf{x}) u(\tilde{\mathbf{x}}) = f(\tilde{\mathbf{x}}), \quad \tilde{\mathbf{x}} = (\tilde{x}, \tilde{y}, \tilde{z}), \quad \text{in } \Omega = (0, 1)^3,$$

with the wavenumber in the non-dimensional domain, denoted by \bar{k} , can be related to the physical quantities in the physical domain by the relation

$$\bar{k} = 2\pi f \bar{l} / c. \quad (2.24)$$

Throughout this thesis, we will use the notation k for the wavenumber regardless of the domain we are considering. However, the meaning should be clear from the context. The wavenumber would be dimensionless if considered domain is unit.

2.6 Spectrum of the Helmholtz operator

An eigenvalue problem for operator L is, finding a pair (ϕ, λ) of eigenfunctions and eigenvalues respectively, such that

$$L\phi = \lambda\phi$$

and ϕ is not a zero function. The one-dimensional discrete eigenvalue problem for the given Helmholtz operator (2.23) with homogeneous Dirichlet boundary conditions (2.8) in a unit domain will read as

$$\left(-\Delta_h - k^2 I_h\right) \phi_h = \lambda_h \phi_h \quad (2.25)$$

If Ω_h is a given discrete domain with grid-size $h = \frac{1}{n}$ and $\underline{x} = [x_j]$ for $1 \leq j \leq n-1$ is the grid vector, then a simple derivation brings out the discrete eigenfunctions

$$\phi_h^\ell = \sin(\ell\pi\underline{x}) \text{ for } 1 \leq \ell \leq n-1 \quad (2.26)$$

and the corresponding eigenvalues

$$\lambda_h^\ell = \frac{1}{h^2}(2 - 2\cos(\ell\pi h) - k^2 h^2). \quad (2.27)$$

Similarly, with a step-size $h = \frac{1}{n}$ in both x- and y-directions on a unit square domain, the discrete eigen functions of a two-dimensional Helmholtz problem with Dirichlet boundary conditions given in Equation (2.13) are

$$\phi_h^{\ell_1, \ell_2} = \sin(\ell_1\pi\underline{x}) \sin(\ell_2\pi\underline{y}) \text{ for } 1 \leq \ell_1, \ell_2 \leq n-1 \quad (2.28)$$

where \underline{x} and \underline{y} are grid vectors in x- and y-direction. corresponding to the eigenvalues

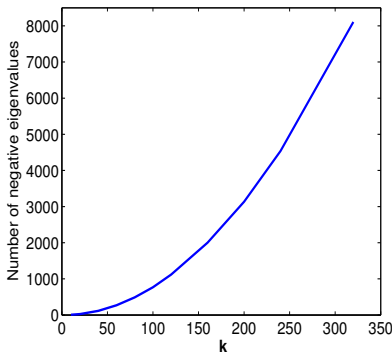
$$\lambda_h^{\ell_1, \ell_2} = \frac{1}{h^2}(4 - 2\cos(\ell_1\pi h) - 2\cos(\ell_2\pi h) - k^2 h^2). \quad (2.29)$$

By graphical interpretation of the above given discrete spectrum, it can easily be shown that the increasing wave number k causes severe indefiniteness in the problem i.e. more negative eigenvalues occur in the spectrum. The negative eigenvalues of the two dimensional Helmholtz problem from the Equation (2.29) are quantified in Figure 2.5(a), which shows the increasing number of negative eigenvalues against the increasing wave number k . The eigenvalues are plotted on a grid which takes approximately 20 grid-points to resolve one wavelength. It is also important to make

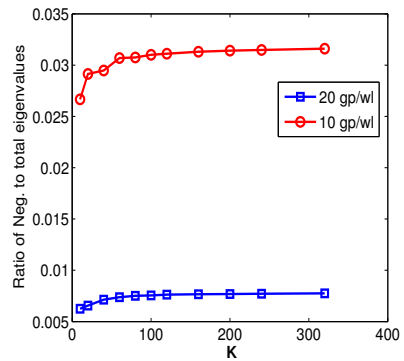
the observation that for a fixed wave number, making the grid step smaller does not influence this characteristic of the spectrum. This means that the number of negative eigenvalues does not increase when the grid is more refined for a fixed wave number, and the number of negative eigenvalues solely depends upon the wavenumber. However the the ratio of the negative eigenvalues to the total number of eigenvalues, which is defined as

$$\text{ratio} = \frac{\text{Number of negative eigenvalues}}{\text{Number of total eigenvalues}},$$

remains exactly the same for wave numbers $k > 100$ as long as kh is constant. This is evident in Figure 2.5(b).



(a) Number of negative eigenvalues for increasing k



(b) Ratio of negative to total eigenvalues

Figure 2.5: The trend of the increasing number of negative eigenvalues for the two-dimensional Helmholtz against the wavenumber. The Grid resolution is set such that there are 20 grid-points per wavelength. On the right, the ratio of the negative eigenvalues to the total eigenvalues is plotted.

Chapter 3

Preconditioner Survey

3.1 Preconditioners for Helmholtz: Survey

The efficient numerical solution of the discrete Helmholtz equation has long been an open problem. The Helmholtz system arising from a FDM discretization appears to be sparse, complex valued due to the Sommerfeld absorbing boundary conditions, symmetric but not Hermitian and carries negative eigenvalues, keeping it into the class of indefinite systems. Further, the elimination of the Sommerfeld (Neumann) boundary conditions makes it non-symmetric. Earlier, direct methods had been used for these problems. But since most practical problems are modeled from a three dimensional geometry and also certain physical constraints require a tight grid refinement, this leads to a very large discrete problem. Thus direct methods became of no use. Two classes of iterative methods were attractive. The first one is *multigrid*, the second one is a Krylov method.

Multigrid methods are considered efficient for elliptic problems [113, 63, 118, 22, 20]. A careful construction of the multigrid components, which are smoother and coarse grid correction, leads to convergence often independent of grid size h . The traditional *smoothers* like Jacobi and Gauss-Seidel do not perform well when applied to indefinite problems, particularly the Helmholtz equation with increasing wave number k [64, 113]. The coarse grid correction process may change the sign of the eigenvalues close to origin [48]. In this case, rather than giving convergence acceleration, coarse grid correction may cause divergence [61]. For more details, the reader is referred to Section 3.4.

The second class consists of Krylov subspace methods [98]. Krylov subspace methods are attractive for sparse linear systems. There are various types of Krylov

subspace methods and the different variants are classified for different systems with certain characteristics [97]. The ill conditioned systems, in particular an indefinite linear system, restrict the choice of Krylov subspace method. For a non-symmetric, indefinite system, well established GMRES [99] and Bi-CGSTAB [115] are the methods of choice. The convergence analysis have been explored significantly for GMRES but the method uses long recursions, which lead to huge memory and CPU time if the number of iterations is large. Bi-CGSTAB has been preferred for using short recursions. The recently proposed IDR(s) method [106] can also be employed. In general, the convergence of Krylov subspace methods depends upon the spectrum of the coefficient matrix. Thus it is necessary for any Krylov subspace method to have a system with a favorable spectrum of the matrix in order to converge fast [98]. The linear system is typically preconditioned to have a favorable spectrum for respective Krylov iterations. Preconditioning means, a Krylov subspace method is applied to the left preconditioned system

$$M_h^{-1}A_h\bar{u}_h = M_h^{-1}b_h \quad (3.1)$$

instead of the system given in Equation 2.22. The preconditioning matrix M_h is chosen such that spectrum of the preconditioned matrix $M_h^{-1}A_h$ has a more favorable spectrum than A_h in order to reduce the number of Krylov iterations. If we take $M_h = A_h$ as preconditioner, this would be the ideal preconditioner as a Krylov method will need only one iteration for the identity matrix. An instant and rough approximation of coefficient matrix A_h can be obtained by picking diagonal entries of A_h , and is called a diagonal preconditioner. Many other preconditioners have been developed and used for various problems depending upon the properties of coefficient matrix A_h . For certain problems, a preconditioner arised from a continuous operator, instead of the coefficient matrix can be used. In the subsequent section, we discuss preconditioners used for the Helmholtz problem particularly.

3.2 Standard Preconditioners for Helmholtz

As we know, the performance of Krylov solvers typically depends upon the choice of the preconditioner. Therefore, many preconditioners have been proposed for Helmholtz, both matrix based and operator based. A class of preconditioners is based upon the factorization of the system matrix. This class of preconditioners for Helmholtz is obtained by Incomplete LU factorization of matrix A_h . An incomplete LU factorization preconditioner (ILU) has been examined in [102, 40] with a variation in fill in. Approximation of LU depends upon the fill-in allowed in the factors L and U . More fill-in produces a better accuracy. The incomplete LU factors are obtained by Gaussian elimination and then adapting fill-in. Most well known type of fill-in is keeping the same sparsity pattern of A_h . This is generally denoted as $ILU(0)$. Accuracy can be increased by allowing more fill-in. $ILU(tol)$ is a variant where the entries in factors L and U valued less than the defined tolerance tol are dropped. For more variants,

the reader is referred to [96, 98]. Since the matrix obtained from the problem of our interest, the Helmholtz problem, is an indefinite one, not all factorizations may be stable. This can be evident in the experiments performed with $ILU(0)$ and $ILU(tol)$ preconditioners. A brief performance of $ILU(0)$ and $ILU(0.01)$ has been presented in Table 3.1, where the number of iterations taken by GMRES without preconditioner and GMRES preconditioned by $ILU(0)$ and $ILU(0.01)$ are compared. These two preconditioners reduced the number of Krylov iterations, particularly $ILU(0.01)$. Though it can be deduced that for large wave number problem, ILU type preconditioners are not practical since $ILU(tol)$ will have a more dense pattern. Conclusively, ILU type preconditioners have only been a good choice for the Helmholtz problem with small wave number.

k	No Prec.	ILU(0)	ILU(0.01)
10	36	21	8
20	82	43	16
30	143	71	24
40	231	99	38
50	341	120	52

Table 3.1: Number of GMRES iterations without preconditioner, with $ILU(0)$ and $ILU(0.01)$ preconditioners.

An analytical approach is considered in [57] for obtaining a factorization preconditioner A_{ILU} . The references [56, 55] show the application of A_{ILU} for Helmholtz problem. An approximate factorization preconditioner has been obtained by separation of variable technique in [91]. This preconditioner is constrained to constant wave number problems and diverges for non-constant wave number problems.

Operator based preconditioners are more problem specific and appear to be more efficient. The early work on operator dependent preconditioners for Helmholtz has been reported in [11, 73] in which a Laplace operator and a Laplace operator with a positive real shift, respectively, are proposed. Both preconditioners lead to good results for medium size wave numbers. For large wave numbers numerical results on the contrary show a steep increase in the number of iterations.

3.3 Historic Development of the Shifted Laplace Preconditioner

Laplace preconditioners for Helmholtz problems were extended with an incorporation of a complex shift in [43]. Later a generalized complex shift has been proposed and

studied in [46, 47]. For $\beta_1, \beta_2 \in \mathbb{R}$ the complex shifted Laplace preconditioner (CSLP), denoted by $M_h(\beta_1, \beta_2)$, can be constructed by discretization of the operator

$$M(\beta_1, \beta_2) = -\Delta - (\beta_1 - i\beta_2)k^2, \quad (3.2)$$

where β_1 and β_2 are real and imaginary shifts respectively. Usually, the operator is obtained by a discretization with the same boundary conditions provided for the original problem. Development of CSLP has been a breakthrough in preconditioning for Helmholtz problem. The Krylov solver preconditioned by CSLP requires a number of iterations that grows only linearly as the wave number increases. Further, with the appearance of the complex shift in CSLP, a computationally feasible solution has become available. As the complex shift introduces damping and renders the preconditioned system amenable to approximate inversion using multigrid or modified factorization methods [77]. More recently algebraic multigrid has been used to invert the preconditioner [17, 1]. Inspired by advances in CSLP, a number of generalizations of the work appeared shortly afterwards in [42, 93, 17, 1, 44] together with applications in different industrial contexts in [120, 3, 94, 89, 90, 114, 2, 87]. Along-with its variations, CSLP preconditioner has been observed as most effective and robust one for Helmholtz.

The 2D constant wavenumber problem defined in Section 2.2 has been solved in Matlab to make a comparison between the ILU and CSLP with different shifts. Table 3.2 presents the number of iterations taken by GMRES preconditioned by $ILU(0)$, ILU with 0.01 fill-in tolerance i.e. $ILU(0.01)$ and various CSLP preconditioners. All preconditioners are inverted exactly. The CSLP preconditioner $M_h(1, 0.5)$ is apparently more close to the original operator and is the optimal one in the list of preconditioners given in Table 3.2. $M(1, 1)$ is also a competitor and can be more useful when approximation of preconditioner is taken into considerations instead of exact inversion.

K	ILU(0)	ILU(0.01)	$M_h(0, 0)$	$M_h(-1, 0)$	$M_h(0, 1)$	$M_h(1, 1)$	$M_h(1, 0.5)$
10	21	8	9	12	10	10	8
20	43	16	19	22	19	18	13
30	71	24	37	38	30	26	17
40	99	38	62	58	40	30	19
50	120	52	96	84	51	33	21
60	143	76	136	107	59	37	23

Table 3.2: A comparison of the number of iterations for ILU type preconditioners and CSLP preconditioners using different shifts for various wave numbers.

3.4 Multigrid approximation of CSLP

Linear systems with a CSLP-like coefficient matrix can be solved with many methods including Krylov subspace and multigrid methods. Multigrid methods are well known to be efficient solvers for elliptic problems. Further, with the appearance of the imaginary shift (β_1, β_2) in CSLP, CSLP becomes more attractive for the multigrid method. The imaginary shift introduces damping and renders preconditioned system amenable to approximate inverse using multigrid [40].

Our focus application are of Geophysical wave imaging, in which usually the domain is either a square or a rectangle. Therefore, the following discussion of multigrid is limited to the geometric multigrid approximation of CSLP, whereas AMG has been reported to approximate CSLP for more complicated geometries in [17, 1].

Multigrid methods combine two basic techniques, a smoother and a coarse grid correction. Since basic iterative methods are observed to have good smoothing characteristics in general, therefore, the high-frequency modes of the error are smoothed by application of basic iterative methods like Jacobi or Gauss Seidel iteration. The process is known as *pre-smoothing*. Whereas the low-frequency modes needs more attention. These low frequency modes can be expressed better on a coarse grid. Thus they are transformed to a coarse grid, smoothed there and then subsequently interpolated back to the fine grid. The fine grid approximation is updated by interpolated error components. The process is known as a *coarse grid correction*. The fine grid approximation encounters the high-frequency error due to the prolongation procedure used to transfer the correction. Therefore, a corrected solution on the fine grid is smoothed, which is called *post-smoothing*. This gives a two-grid method. The size of the coarse problem is substantially smaller than the size of the fine grid problem, still the coarse problem is too large to solve directly. However the coarse problem has the same form as of the fine level. Therefore, the coarse problem can further be iterated by an other two-grid cycle, thus introducing an other coarser problem. This process can be repeated recursively until a coarsest level problem with a very few grid-points is reached. This comes up as a *multigrid cycle* [107, 113]. Different cycle strategies can be implemented for various problems.

In the start of the development of CSLP, only an imaginary shift was incorporated with a Laplace operator, which was reported in [43]. The convergence factor of multigrid methods with V-cycle and W-cycle and with different smoothing strategies are presented in Figures (3.1) and (3.2). The CSLP defined in Equation (3.2) along with Sommerfeld boundary Conditions (2.5) is used for the experiment, for wave number $k = 40$ and $\beta_1 = 0$ and with a range of $0 \leq \beta_2 \leq 1$. The discretized unit square domain turns up a grid of size 64×64 abiding Condition (2.18). Figure (3.1(a)) gives the convergence factor (CF) for a multigrid V-cycle. Figure 3.1(b) shows the CF for

a multigrid W-cycle. In both methods, Gauss-Seidel has been used as smoother. Approximation of CSLP is comparable with both multigrid V- and W-cycles. However a W-cycle is computationally expensive. Next Figures (3.2(a)) and (3.2(b)) show the convergence factor for V- and W-cycles respectively, whereas Jacobi iterations are used as smoothers. The figures show different number of pre- and post-smoothing. For e.g. $V(v_1, v_2) = V(1, 1)$ is a V-cycle with one pre- and one post-smoothing step. One can observe that a multigrid approximation of CSLP is efficient when the real part of the shift is zero i.e. $\beta_1 = 0$.

Later on, generalized complex shifts are proposed in [106, 46]. It has been shown that inclusion of the real shift $\beta_1 = 1$ reduces the efficiency of multigrid approximation, however the CSLP with both real and imaginary shifts decreases the number of iterations of the global method. It is straightforward to see that by increasing the imaginary part, CSLP become more favorable for multigrid approximation. This may deteriorate the global convergence of Krylov iterations, as an increase in the imaginary shift makes CSLP less resembling to the original operator, but inclusion of real shift is still preferred since it causes significant reduction in global number of iteration.

An advancement in this regard has been proposed in [24], where different relaxation parameters have been used at different levels. Briefly, the choice of shifts is sensitive and multigrid approximation is efficient if the imaginary shift is increased however the global convergence is slower because the preconditioner is a worse approximation of the original matrix. An analysis on the choice of shift for CSLP has been performed in [30]. The issue of a flexible imaginary shift is further discussed in Chapter 4 where the possibilities of an increasing imaginary shift are suggested, in order to have a better approximation of CSLP.

3.5 Spectrum of the CSLP/Convergence Obstacles

Generally, preconditioners are used to cluster the spectrum of the coefficient matrix from the system-to-be-solved. Applying CSLP $M_{h,(\beta_1, \beta_2)}^{-1}$ to the Helmholtz operator A_h , the spectrum of $M_{h,(\beta_1, \beta_2)}^{-1} A_h$ becomes clustered in the complex plane near one and bounded above by 1 in absolute value [59]. Further spectral properties of CSLP are also elaborated in [59, 100, 79]. Analysis also shows that the smallest eigenvalues of the preconditioned operator rush to zero as the wave number increases. This has been shown in Figure (3.3). This figure illustrates that the spectral radius is bounded by 1 and that the spectrum has more eigenvalues around the origin as the wave number increases, as the left-hand side figure shows few eigenvalues near zero, where more eigenvalues occur to be around the origin in the right-hand side figure where the wave number increases from $k = 30$ to $k = 120$.

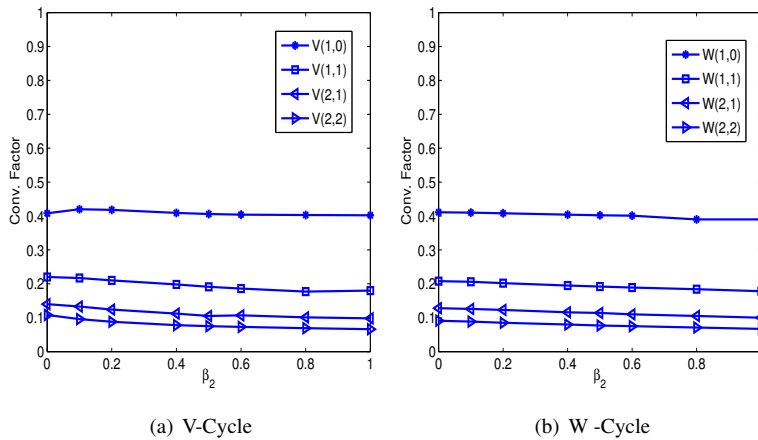


Figure 3.1: Convergence factor for $M_h(0, \beta_2)$ with the Gauss Seidel smoother versus imaginary shift β_2

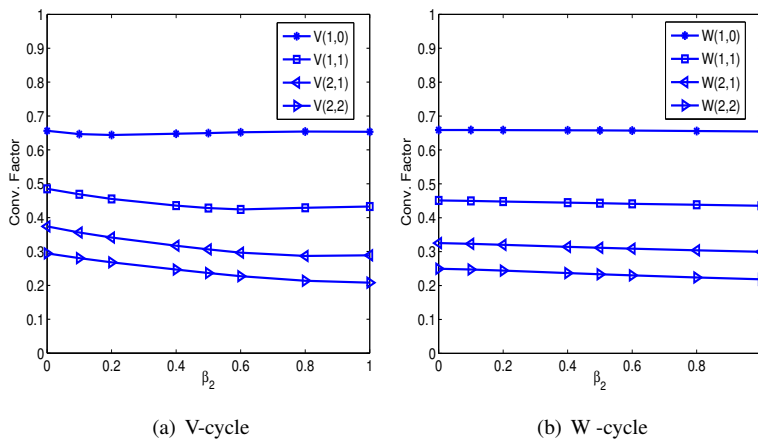


Figure 3.2: Convergence factor for $M_h(0, \beta_2)$ with the Jacobi smoother versus imaginary shift β_2 .

3.6 Down-sides: Need of Projection

The increasing wave number makes the smallest eigenvalues in the spectrum of $M_h^{-1}A_h$ to rush to the origin in complex plane. For certain large wave numbers, it is even inconceivable to apply CSLP. The small eigenvalues need to be taken care of. *Deflation* is a technique commonly used to get rid of a certain part of the spectrum, and to force the “bad” eigenvalues not to participate in the Krylov iterations. This technique has

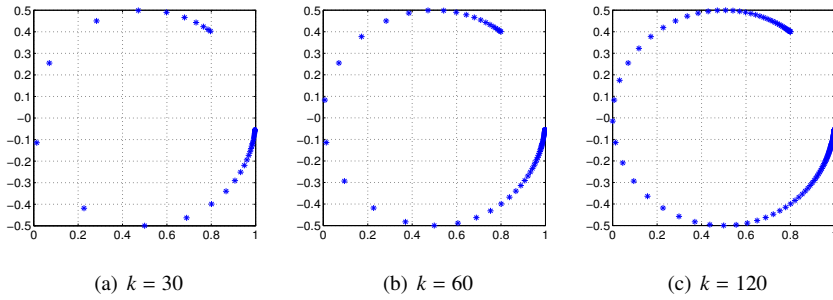


Figure 3.3: Spectrum of the $M_{h,(1,0.5)}^{-1}A_h$ for different values of the wave number k .

been used for many problems [109, 25, 69]. For Helmholtz problems, deflation has been proposed in combination with the shifted Laplace preconditioner in [45, 42]. This can be seen as a second level preconditioner that removes small eigenvalues.

The next chapter details more about deflation in general, and deflation for Helmholtz problem followed by the analysis of the deflation.

Chapter 4

Deflation

Convergence of the Krylov subspace method are typically adversely affected by small eigenvalues. Deflation is the technique that aims at dealing with the undesired part of spectrum in the (un)preconditioned linear system. Deflation on a system-to-be-iterated can be performed either by projecting the spectrum by applying the *projection preconditioner* P [84] or by augmenting eigenvectors corresponding to the convergence-hampering eigenvalues into the Krylov subspace [80, 81]. As eigenvectors are usually costly to construct, therefore the projection preconditioner has often been used to deflate unwanted eigenvalues to make its influence void on Krylov subspace [52, 111]. Usually, but not necessarily, the stagnated convergence is caused by eigenvalues of small absolute value. The next sections elaborate on projection deflation. Development of a projection preconditioner (deflation) can be considered the same as of the coarse grid correction of standard multigrid method [103] while multigrid inter-grid operator are taken as deflation matrices.

In the next section, the deflation is explained, the subsequent sections detail the characteristics of the deflation preconditioner. In the final sections, we discuss the deflation preconditioner in context of the Helmholtz problem.

4.1 Deflation as Preconditioner

The basic idea of deflation is, by means of the projection preconditioner, to bring eigenvalues of small absolute value to zero, in general for complex valued linear systems. The deflation preconditioner can be defined as

$$P = I - AQ \text{ where } Q = ZE^{-1}Z^T \text{ and } E = Z^T AZ \quad (4.1)$$

where $Z \in \mathbb{R}^{n \times r}$ is introduced as a *deflation matrix* whose $r < n$ columns are called deflation vectors.

The deflation preconditioner 4.1 can be characterized based on the method it is constructed, and the choice of deflation vectors. We discuss these characteristics of the deflation preconditioner later on in subsequent sections in details. The characteristics are :

- Construction method
- Shifting (deflated spectrum) to 1
- Choice of deflation vectors i.e. Z
- Implementation/Parallel Implementation

Essentially, the construction method is base to differentiate the types of preconditioner. Where as the choice of deflation vectors influences the effectiveness of deflation preconditioner. Subsequent sections discuss the above listed characteristics respectively.

4.2 Variants of Deflation Preconditioner

Due to different choices in the deflation vectors, there are many types of deflation preconditioners. Though in context of our target application Helmholtz, we will consider five deflation variants. We subdivide those into two groups. The first group consists of two members and performs deflation based on the CSLP preconditioned Helmholtz operator, instead of the un-preconditioned one. This group gives rise to complicated preconditioners. They have been used in [42, 45]. The second group consists of three members and performs deflation based on the un-preconditioned Helmholtz operator. The distinguishment of these variants in both groups is based on the construction of the operator E .

To cast five deflation variants into the same framework we will introduce the following notation. We use this notation only in the current section. We will denote by \mathcal{A}_h the system matrix to which deflation will be applied. In the following \mathcal{A}_h will be set equal to A_h as defined in 2.23 or to (some approximation of) $M_{h,(\beta_1,\beta_2)}^{-1} A_h$, where $M_{h,(\beta_1,\beta_2)}$ is defined by 3.2. Let n denoted the size of \mathcal{A}_h . We define the matrix $Z_h \in \mathbb{R}^{n \times r}$ whose $r < n$ columns are the deflation vectors. These vectors should be chosen such that the matrix Z_h has full rank. Given a choice for \mathcal{A}_h , we will denote by $E_{2h} \in \mathbb{C}^{r \times r}$ its approximation on the coarser grid. In the following E_{2h} will be constructed by either Galerkin coarsening, approximate Galerkin coarsening or re-discretization. Construction of operator E_{2h} will be the defining factor for the different variants. With the notation defined, the deflation preconditioner 4.1 can be written as

$$\mathcal{P}_h = I - \mathcal{A}_h Q_h \text{ where } Q_h = Z_h E_{2h}^{-1} Z_h^T \text{ and } E_{2h} = Z_h^T \mathcal{A}_h Z_h \quad (4.2)$$

In the following sections, we describe the five deflation variants in more detail.

4.2.1 Deflation using the CSLP Preconditioned Operator: Idealized Variant (V1)

In the first of the five deflation variants that we consider in this thesis, the coarse grid operator E_{2h} is defined by Galerkin coarsening of the CSLP preconditioned Helmholtz operator, i.e., we set

$$\mathcal{A}_h = M_{h,(\beta_1,\beta_2)}^{-1} A_h \text{ and } E_{2h} = Z_h^T \mathcal{A}_h Z_h. \quad (4.3)$$

One can observe that for every coarse grid iteration, an application of $M_{h,(\beta_1,\beta_2)}^{-1}$ is needed. This makes the method infeasible and impractical to implement. Since merely construction of E_{2h} costs the inversion of $M_{h,(\beta_1,\beta_2)}$. Application of this deflation variant is performed on \mathcal{A}_h , which is the CSLP preconditioned Helmholtz operator. The spectral properties for the variant are explored in the Chapter 5. Later in this chapter, the complexity of the implementation for this variant is discussed.

4.2.2 Deflation using the CSLP Preconditioned Operator: Practical Variant (V2)

The first variant is too expensive to use. It costs inversion of CSLP at the fine grid for construction of operator E_{2h} on the coarse grid. The second deflation variant renders the first variant computationally feasible by introducing an approximation. We define the coarse grid operators $M_{2h,(\beta_1,\beta_2)}$ and A_{2h} as $M_{2h,(\beta_1,\beta_2)} = Z_h^T M_{h,(\beta_1,\beta_2)} Z_h$ and $A_{2h} = Z_h^T A_h Z_h$ respectively. The approximation of the Galerkin coarse grid operator is computed as follows. For sake of simplified derivation, we consider the deflation matrices X_h and X_h^T and we assume matrices $X_h \in \mathbb{R}^{n \times n}$ and X_h^T are invertible. Then the approximation of the coarse grid operator follows as:

$$\begin{aligned} E_{2h} &= X_h^T \mathcal{A}_h X_h \\ &= X_h^T (M_{h,(\beta_1,\beta_2)}^{-1} A_h) X_h \\ &= X_h^T X_h (X_h^{-1} M_{h,(\beta_1,\beta_2)}^{-1} (X_h^T)^{-1} X_h^T A_h) X_h \\ &= X_h^T X_h (X_h^{-1} M_{h,(\beta_1,\beta_2)}^{-1} (X_h^T)^{-1}) (X_h^T A_h X_h) \\ &= \Theta_h M_{2h,(\beta_1,\beta_2)}^{-1} A_{2h}. \end{aligned} \quad (4.4)$$

Though our deflation matrices Z_h and Z_h^T are not invertible in general. However in consideration of above derivations, the approximation of the coarse grid operator can be done as

$$E_{2h} = Z_h^T \mathcal{A}_h Z_h \approx \Theta_h M_{2h,(\beta_1,\beta_2)}^{-1} A_{2h} \quad (4.5)$$

where $\Theta_h = Z_h^T Z_h$ is an approximation term. This approximation makes the preconditioner feasible to use, however the resulting convergence can be slower than the convergence for the ideal variant. We will discuss this in Chapter 5 with analysis.

Apart from using operator \mathcal{A}_h for construction of the deflation preconditioner, this variant of deflation is also supposed to be applied on \mathcal{A}_h straightforwardly. Further implementation-related issues are discussed in Section 4.7 and 4.7.1.

4.2.3 Deflation using the Helmholtz Operator (V3)

The third deflation variant deflates the Helmholtz operator, i.e., we set

$$\mathcal{A}_h = A_h \text{ and } E_{2h} = Z_h^T \mathcal{A}_h Z_h. \quad (4.6)$$

Here E_{2h} is the natural Galerkin operator. This variant is most elegant and refined one amongst all the variants, which are discussed in this document. The development of this variant of deflation is based on the Helmholtz operator A_h . The preconditioner can be applied with combination of standard preconditioner. For the Helmholtz operator, application of this variant is performed in combination with CSLP $M_{h(\beta_1, \beta_2)}$. We will elaborate the implementation and approach to multilevel extension of this variant in Sections 4.6 and 4.6.1. Similar to earlier variants, an extensive spectral analysis shall be performed in Chapter 5.

4.2.4 Deflation using the Complex Shifted Helmholtz Operator (V4)

The fourth variant deflates the complex shifted Helmholtz operator, i.e., we set

$$\mathcal{A}_h := \Delta_h - (\hat{\beta}_1 - i\hat{\beta}_2)k^2 I \text{ and } E_{2h} = Z_h^T \mathcal{A}_h Z_h. \quad (4.7)$$

This variant can be seen as a follow-up of the third variant. Where for multilevel application of the third variant requires the iterative solution of the coarse grid operator. The motivation comes from the results of spectral analysis, which indicates instability in the iterative coarse grid solve in the third variant. In this variant the coarse grid solve is stabilized by incorporating real and imaginary shifts. Immediate consequence of these shifts is that this variant is not a projection. Analysis verifies this fact. As the third variant, this variant of deflation can also be applied in combination with a standard preconditioner, which in case of Helmholtz we consider CSLP. Naturally the shifts used in development of this variant ($\hat{\beta}_1, \hat{\beta}_2$) are different from those used in CSLP. For further analysis, we refer to Chapter 5.

4.2.5 Deflation using Re-discretization (V5)

In all previous defined variants of the deflation preconditioner, we use the Galerkin coarse grid operator. In a two dimensional problem, in contrast to the five-point stencil, the Galerkin coarse grid operator turns out to be a nine-point stencil. Similarly the Galerkin CSLP has also a nine-point stencil. Re-discretization of the coarse grid

avoids the stencil growth. The fifth variant defines a non-deflation operator by defining the coarse grid operator E_{2h} by a re-discretization approach. We will consider re-discretizing the Helmholtz operator, i.e., we set

$$E_{2h} = \text{re-discretization}(A_h). \quad (4.8)$$

This choice is motivated by the wish to preserve the sparsity pattern of the fine grid operator on the coarser grids. Details are discussed via analysis in Chapter 5.

Typically, the standard deflation preconditioner defined in Equation 4.1 projects a certain part of the spectrum to zero. For many application, one might need to project at other values, rather than to zero. Essentially, for multilevel approaches of deflation, deflation to zero might cause serious problems of scattering eigenvalues around zero, as the coarse grid operator is approximated in multilevel approaches. We discuss this in the next section, how one can shift the deflated eigenvalues to a certain value not equal to zero.

4.3 Shifting Deflated Spectrum

The deflation, described above, shifts troublesome eigenvalues to zero unlike multigrid methods that push such eigenvalues to unity. Also we note that in the deflation preconditioner, the operator A_{2h} appears to be inverted. The choice of deflation matrices decides the sparsity pattern of A_{2h} . If the columns of the deflation matrix are the eigenvectors of A_h , then A_{2h} simply is a diagonal matrix of eigenvalues and is easy to invert. But eigenvectors are not usually used as deflation vectors, as they are expensive to compute. Some sparse vectors, which span, approximate the span of the “bad eigenvectors” can be used. As in the case of Helmholtz, the prolongation matrix 4.13 of multigrid has been used as deflation matrix, which is sparse. Even with this choice of sparse deflation matrix, which results in a relatively sparse A_{2h} , the exact inversion of A_{2h} is impractical for large problems and one has to resort to approximate solvers instead. Without proper care, this will however lead to the occurrence of close-to-zero eigenvalues in the preconditioned systems, resulting in slow Krylov convergence. This is evident in Figure 4.1, where the exact eigenvalues of the deflated preconditioned one-dimensional operator $M_{h,(\beta_1,\beta_2)}^{-1} P_{h,2h} A_h$, with parameters wavenumber k , grid-size h and (β_1, β_2) valued at 100, $\frac{1}{160}$ and (1, 0.5) respectively, are plotted with exact and in-exact inversion of the coarse grid operator A_{2h} respectively. The in-exact inversion of A_{2h} is meant by approximation of A_{2h}^{-1} by few GMRES iterations. Figure 4.1(a) shows the eigenvalues which involves exact inversion of the coarse grid operator A_{2h} where we see that the eigenvalues are nicely projected to the origin. On the other hand, in Figure 4.1(b) in the highlighted box, we see a spread of eigenvalues around 0 caused by in-exact inversion of A_{2h} . The CSLP has been inverted exactly in both figures.

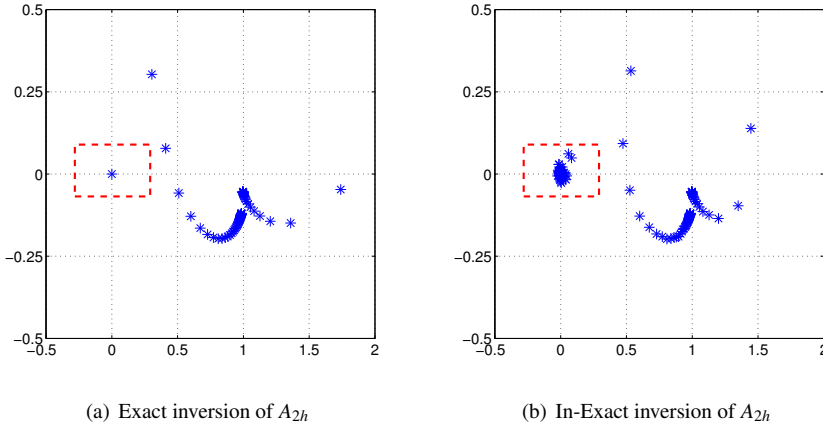


Figure 4.1: Approximation of the coarse grid operator in Equation 4.1 disperse the eigenvalues in the vicinity of origin in the spectrum of the deflated preconditioned Helmholtz operator given on the left-hand side of Equation 4.17 .

This indicates the sensitivity of approximate solve of A_{2h} . This can be avoided by deflating to the largest eigenvalue (or certain value, where the spectrum is expected to be clustered) of the preconditioned system, instead of deflating to zero. The disperse of the eigenvalues near the clusters away from the origin would be more favorable than that around origin. This can be performed by adding a term, which shifts the *deflated eigenvalues* from the origin to the value γ [83, 52]. With this additive term in 4.16, we obtain the preconditioner of the type

$$P_{h,\gamma} = P_h + \gamma Q_h = (I_h - A_h Q_h) + \gamma Q_h. \quad (4.9)$$

Some deflation preconditioner shifts the deflated spectrum without any trouble to the value assigned. Whereas other deflation preconditioners, particularly those constructed for certain applications, are reluctant to shift the deflated spectrum in a uncomplicated way. One of the examples is variant **V2** deflation preconditioner which is elaborated in Section 4.7.1.

In the next section, we discuss the choice of deflation vectors for Helmholtz equation.

4.4 Choice of Deflation Vectors

Deflation vectors can be seen as basic ingredients of the deflation preconditioner. The choice of the deflation matrix Z is related with the invertibility of the matrix E therefore the deflation matrix Z should at least be a full rank matrix. Zero and one are the

only eigenvalues of the deflation preconditioner 4.1, assuring it to be a projection. The choice of deflation vectors is an open question for different problems. Relative in terms of the background of the problem. For e.g. in [69], the approximations to rigid body modes of composites has been chosen as deflation vectors. Also in the problems with a linear system with multiple right-hand sides can benefit by using extracted residuals from one right-hand side as deflation vectors for subsequent right-hand sides. These strategies for symmetric as well as non-symmetric matrices have been reported in [104, 31]. Generally, by taking eigenvectors of matrix A as the deflation vectors, i.e. say;

$$Z = [v_1 v_2 \dots v_r] \quad (4.10)$$

are the eigenvectors corresponding to eigenvalues $\{\lambda_1 \lambda_2 \dots \lambda_r\}$ of A , whereas the spectrum of A is given by

$$\phi(A) = \{\lambda_i \mid 1 \leq i \leq n\} \quad (4.11)$$

then it can be easily verified [111] that the eigenvalues of the deflated operator PA will be

$$\phi(PA) = \{0 \dots 0 \lambda_{r+1} \lambda_{r+2} \dots \lambda_n\} \quad (4.12)$$

In this case, the deflation preconditioner P simply deflates eigenvalues corresponding to those eigenvectors to 0. The eigenvectors are often computationally expensive. Therefore, it is impractical to construct the deflation preconditioner using eigenvectors as deflation vectors. Subject to availability of eigenvectors Z , Z is a dense matrix, and subsequently the whole deflation preconditioner will also be dense. This may be computationally expensive. Approximation of the eigenvectors corresponding to (unwanted) eigenvalues can be considered. We notice that the deflation preconditioner involves the inversion of operator E . The dense deflation leads to a dense E and ultimately a dense the deflation preconditioner 4.1. Therefore the dense (approximation of eigenvectors) deflation vectors are avoided in practice. The approximations of the eigenvectors can be useful where sparsity is taken into consideration, as the sparsity of deflation vectors plays a vital role in construction of the deflation operator. For details of possible choices of deflation vectors, the reader is referred to [109, 108].

4.4.1 Deflation Vectors for Helmholtz

In this section we concentrate on the construction of the deflation preconditioner particularly for the linear system arising from the Helmholtz equation. The base component in the deflation preconditioner is the deflation matrix Z . As there is no established theory to choose deflation vectors, however a better and necessarily sparse approximation of the eigenvectors would be theoretically preferred. The dense deflation vectors lead to a dense deflation preconditioner and this might not be computationally acceptable. Talking particularly about deflation vectors for Helmholtz system, we noticed that the analysis of the Helmholtz matrix preconditioned by CSLP

shows that the preconditioned operator contains many small eigenvalues particularly when the wavenumber is increased. Thus we perform deflation with geometrically constructed multigrid vectors. This idea originated from multigrid framework. As it is well known that the multigrid inter-grid operators highlight the small frequencies and projects them on coarser levels, which is helpful to treat them on coarser levels. They are also sparse, hence computationally cheap to implement. For clarity, before setting up the deflation subspace matrix, we assume that p is a non-zero natural number and the given domain Ω is discretized on an uniform mesh with $n = 2^p$ elements and mesh width $h = 1/n$, in each direction for the two- and three-dimensions.

We also consider the Helmholtz system given in Equation 2.22, assumed to be of size n . Now in a follow-up of the notation defined in an earlier section, for the Helmholtz problem we set the deflation subspace in 4.1 equal to the coarse to fine grid interpolation operator i.e.

$$Z = I_{2h}^h \quad (4.13)$$

with the standard coarsening strategy $h \rightarrow 2h$. Here we symbolize the things in context of a two-dimensional problem. Corresponding equivalents for one-dimensional as well as three dimensional problems will be used. Comprehensively for two-dimensional setting, the bilinear interpolation operator such that for fine grid points not belonging to the coarse grid has the stencil

$$[I_{2h}^h] = \frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}_{2h}^h. \quad (4.14)$$

With the choice $Z = I_{2h}^h$, the Z^T will be full-weighting restriction operator

$$I_h^{2h} = c(I_{2h}^h)^T \quad (4.15)$$

with c a constant scalar. With this choice, the deflation preconditioner Equation 4.1 can be rewritten as

$$P_h = I_h - A_h Q_h \text{ where } Q_h = I_{2h}^h A_{2h}^{-1} I_h^{2h} \text{ and } A_{2h} = I_h^{2h} A_h I_{2h}^h. \quad (4.16)$$

With this choice the deflation operator P_h defined by (4.16) coincides with the transpose of the two-grid correction operator (in multigrid terminology) with a coarse grid operator A_{2h} build by Galerkin coarsening.

The operator Q_h in Equation 4.16 inherits the complex symmetry from A_h , which is proved in Lemma 1.

Lemma 1. *If A_h be a symmetric matrix, I_h^{2h} and I_{2h}^h as defined in Equation 4.14 and 4.15 then A_{2h} and Q_h as given in Equation 4.16 are also symmetric.*

Proof. $(A_{2h})^T = (I_{2h}^h A_h I_{2h}^h)^T = (I_{2h}^h)^T (A_h)^T (I_{2h}^h)^T = I_h^{2h} (A_h) I_{2h}^h$
and
 $Q_h^T = (I_{2h}^h A_{2h}^{-1} I_{2h}^h)^T = (I_{2h}^h)^T (A_{2h}^{-1})^T (I_{2h}^h)^T = I_h^{2h} (A_{2h}^{-1}) I_{2h}^h.$ \square

It is also easy to verify that P_h is a projection and thus the spectrum of $P_{h,2h}$ consists of 0 and 1. If it is assured that the Helmholtz operator A_h is non-singular, then the choice of *restriction* and *prolongation* operators as deflation matrix also assures the non-singularity of matrix A_{2h} .

4.5 Implementation

In this section, we discuss the pros and cons of the implementation of deflation preconditioner. The development of the deflation preconditioner is motivated by the spectrum of the (preconditioned) operator. The deflation operator can be applied along-with a standard preconditioner. The deflation preconditioner can be considered as second level preconditioner, where the deflation preconditioner is used to transform the spectrum of a system preconditioned by some trendy preconditioner more favorable for a Krylov subspace method. Suppose for a given linear system

$$Au = b,$$

which is preconditioned by some stereotype preconditioner, denoted by M_h , then the deflation operator can be considered as second level preconditioner and the linear system preconditioned by two preconditioners can be written as

$$M^{-1}PAu = M^{-1}P.$$

In our case the linear system is the Helmholtz system $A_h u_h = b_h$, the first level preconditioner is CSLP, as defined in Equation (3.2). Also the motivation comes from the spectrum of the Helmholtz operator preconditioned by CSLP. Therefore, we explain the procedure of the application of the deflation preconditioner on the CSLP preconditioned Helmholtz operator M_h

$$M_h^{-1} A_h u_h = M_h^{-1} b_h$$

and it reads as

$$M_h^{-1} P_h A_h u_h = M_h^{-1} P_h b_h \quad (4.17)$$

It is worth to mention here that the preconditioned System (4.17) is equivalent of solving the system

$$P_h^T M_{h,(\beta_1, \beta_2)}^{-1} A_h u_h = P_h^T M_{h,(\beta_1, \beta_2)}^{-1} b_h \quad (4.18)$$

The same has been discussed in [112, 110] and subsequently has been proved [111] that both systems have the same spectrum, i.e.

$$\sigma(P_h^T M_{h,(\beta_1, \beta_2)}^{-1} A_h) = \sigma(M_{h,(\beta_1, \beta_2)}^{-1} P_h A_h) \quad (4.19)$$

The implementation of 4.18 limits to choose the starting vector as $Q_h b_h + P_h^T x_o$, where x_o is an arbitrary starting vector. Where with the implementation of 4.17, the approximated solutions x_{j+1} at iteration $(j + 1)^{th}$ is suggested to be updated i.e.

$$x_{j+1} = Q_h b_h + P^T x_{j+1}$$

However the deflation preconditioner combined with "shifting term" Q_h avoids this complexity. We discuss this later in Section 4.6.1. We perform the application of the deflation, the way it is defined in 4.17. For sake for elaboration, we present it in Algorithm 1 where the deflation preconditioner in combination with CSLP is applied on the Helmholtz system and solved by GMRES. In short, Algorithm 1 shows how GMRES is employed to solve the two level preconditioned system 4.17. One can notice at the end that the iterated solution is updated as discussed above. It is necessary to mention that it becomes more convenient to analyze 4.18, it allows to perform analysis in framework of multigrid fashion. Therefore, we use the 4.18 for analysis. We explore further about this in the next chapter.

Algorithm 1 Preconditioned Deflated GMRES for system $Au = b$

- 1: Choose u_0 and compute $r_0 = b - Au_0$, $b_0 = \|r_0\|$ and $v_1 = r_0/b_0$
 - 2: **for** $j:=1,2,\dots$ k or until convergence **do**
 - 3: $\tilde{v}_j = Pv_j$
 - 4: $w = M^{-1}A\tilde{v}_j$
 - 5: **for** $i:=1,2,\dots$ j **do**
 - 6: $h_{i,j} := w^T v_i$
 - 7: $w := w - h_{i,j}v_i$
 - 8: **end for**
 - 9: $h_{j+1,j} := \|w\|$
 - 10: $v_{j+1} := w/h_{j+1,j}$
 - 11: **end for**
 - 12: Store $V_k = [\tilde{v}_1, \dots, \tilde{v}_k]$; $H_k = \{h_{i,j}\}$, $1 \leq i \leq j + 1$, $1 \leq j \leq m$
 - 13: Compute $y_k = \operatorname{argmin}_y \|b_0 - H_k y\|$ and $u_k = u_0 + V_k y_k$
 - 14: The entries of upper $k + 1, k$ Hessenberg matrix H_k are the scalars $h_{i,j}$
 - 15: Update approximated solution $u_k = Qb + P^T u_k$.
-

In next part of the chapter, we choose the deflation variants, discussed earlier, which are more suitable for Helmholtz systems. We elaborate these variants to the extent of multilevel extensions.

4.6 ADEF1 preconditioner

The term ADEF1 was first coined in [82, 83] for the preconditioner which is primarily the third variant **V3** but it also combines the shifting term, which is defined in Sec-

tion 4.3. As the deflation preconditioner is set to be applied to the preconditioned Helmholtz problem, and recalling the shift-preconditioner from Section 4.3, we get the preconditioner

$$P_{h,ADEF1} = M_{h,(\beta_1,\beta_2)}^{-1}P_h + \gamma Q_h,$$

which is called *ADEF1*. We note that the basic deflation preconditioner P_h is applied to the preconditioned Helmholtz and we also know from [59] that the spectrum of preconditioned Helmholtz by CSLP is known to be bounded above by 1. Thus for case of the Helmholtz problem, this helps in choosing that particular value of γ as 1. Thus this two level preconditioner ADEF1, a combination of the CSLP and the deflation preconditioner with shift-of-spectrum is written as

$$P_{h,ADEF1} = M_{h,(\beta_1,\beta_2)}^{-1}P_h + Q_h \quad (4.20)$$

As we mentioned earlier that this preconditioner shifts the corrected or “deflated low eigen frequencies of the spectrum” to the value $\gamma = 1$. This technique immediately tolerates the approximation of coarse grid operator A_{2h} , as the deflated part of spectrum is merely shifted to the value $\gamma = 1$ without affecting the remaining part of the spectrum. Approximate solve of the coarse grid operator is not possible, when the shifting-of-spectrum strategy is not adapted. In the earlier case, the spectrum is deflated to 0 and the approximate solve of A_{2h} could have caused the dispersion of the small eigen values around zero. The linear system preconditioned by ADEF1 preconditioner reads as

$$P_{h,ADEF1}A_h u_h = P_{h,ADEF1}b_h \quad (4.21)$$

The issue of controlling the approximate solve with A_{2h} are further discussed in the next section.

4.6.1 ADEF1 Multilevel

The algorithm described in the previous section can be viewed as a two-level preconditioner for an outer Krylov iteration in which the CSLP preconditioner 3.2 acting on the first level is combined with the deflation algorithm (4.16) acting on the second and in addition with the scaling of the deflated spectrum. Figure 4.2 also explains this application of two-level preconditioner excluding the scaling part of the deflated spectrum. Conventionally, the first level preconditioner, CSLP preconditioner can be approximated by using standard multigrid cycle [47] or with some adaptation of multigrid cycle ingredients as required by problems with heterogeneity. The deflation part of the ADEF1 requires the solution of a coarse grid problem with A_{2h} as coefficient matrix. Of course A_{2h} is impractical to invert explicitly, specially for large wave number problem. To solve A_{2h} iteratively, a Krylov subspace method can be used. To understand this, we consider the preconditioned step inside the Krylov subspace

method, i.e.

$$\begin{aligned}
 z &= (M_{h,(\beta_1,\beta_2)}^{-1}P_h + Q_h)v \\
 &= (M_{h,(\beta_1,\beta_2)}^{-1}(I - A_hQ_h) + Q_h)v, \\
 &= M_{h,(\beta_1,\beta_2)}^{-1}(v - A_hQ_hv) + Q_hv, \\
 &= M_{h,(\beta_1,\beta_2)}^{-1}(v - A_hv') + v',
 \end{aligned} \tag{4.22}$$

where

$$v' = Q_hv = (I_{2h}^h A_{2h}^{-1} I_h^{2h})v \tag{4.23}$$

The vector v is the vector to be preconditioned at every iteration of the Krylov subspace method. We get the vector v' by the matrix-vector product in 4.23. The vector v is restricted by I_h^{2h} , as seen in 4.22,

$$v_R = I_h^{2h}v. \tag{4.24}$$

With v_R , now we have the Galerkin system with coefficient matrix A_{2h} given in 4.22, and that can be written as

$$v'_R = A_{2h}^{-1}v_R \iff A_{2h}v'_R = v_R \text{ (solve for } v'_R) \tag{4.25}$$

The system $A_{2h}v'_R = v_R$ given in Equation 4.25 can be solved iteratively. The approximate solve of A_{2h} does not influence the effectiveness of the preconditioner at large, as we deflate to 1 instead of 0. However this solve can be sensitive for global convergence and may need tight accuracy. To achieve acceptable accuracy, the Krylov solve on A_{2h} may be proven more costly in terms of iterations. We know that this coarse matrix has the same properties of the original Helmholtz operator. This immediately suggests that the coarse grid problem can be preconditioned by CSLP. This will substantially reduce the number of iteration for coarse grid Krylov solve. Also the deflation preconditioner 4.1 can be applied to this Galerkin system. Precisely, the ADEF1 preconditioner can be applied to Galerkin system with matrix A_{2h} , while being iteratively solved by Krylov subspace method. This application of ADEF1 at coarse level $2h$ will require solving another coarser system at level $4h$. The *Krylov method preconditioned by ADEF1* can be applied recursively on subsequent Galerkin systems. By doing this, we will have a variable preconditioner at each iteration of outer Krylov method. A flexible outer subspace Krylov method will be required in order to precondition the vector at each iteration with a variable preconditioner. Krylov methods such as Flexible-GMRES [95], GCR [36] and flexible IDR [60] can be employed. Note however that the convergence of the outer Krylov method critically depends on the accuracy to which the linear system with A_{2h} as coefficient matrix is solved. We will further discuss the issue of accuracy requirement for approximate solve of different level Galerkin systems while presenting numerical experiments in Chapter 6.

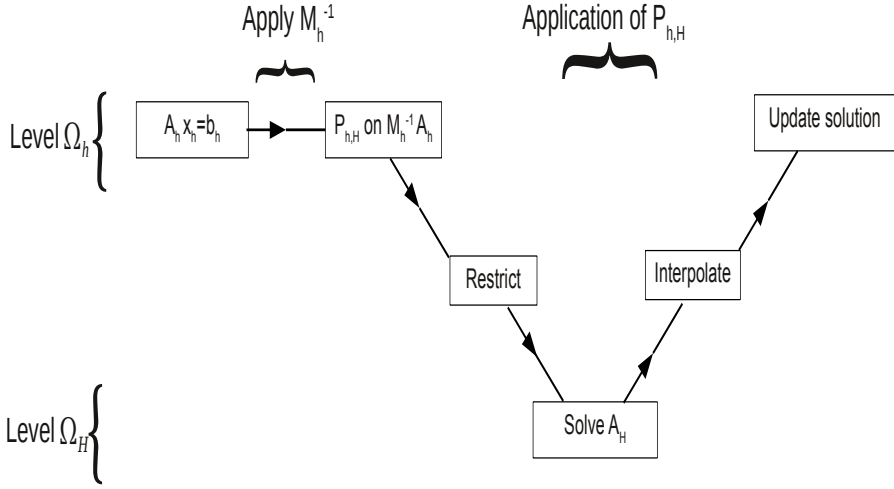


Figure 4.2: A schematic representation of the two level ADEF1 algorithm considered.

4.6.2 ADEF1 Multilevel Algorithm Implementation

Though the implementation of ADEF1 preconditioner involves a complex procedure of approximating Galerkin operator while using ADEF1 preconditioner at subsequent coarser levels. To understand this vital part of multilevel algorithm we present the implementation multilevel preconditioner in Algorithm 2. One can observe that it is indeed possible to implement ADEF1 preconditioner in simplified way. To present the Algorithm, we introduce some symbols for the involved operators as follows:

- The (Galerkin, except on level $j = 1$) Helmholtz operator on all possible levels $j = 1, 2, \dots, A_{jh}$ is denoted by $A^{(j)}$.
- The (Galerkin, except on level $j = 1$) CSLP preconditioner on all possible levels $j = 1, 2, \dots, M_{jh}$ is denoted by $M^{(j)}$.
- The deflation subspace matrices I_h^{2h} and I_{2h}^h operators are denoted by $Z^{(1,2)}$ and $Z^{(1,2)T}$ on level $j = 1$. And for the rest of levels, $j = 2, 3, \dots$, they are denoted by $Z^{(2,3)}$ & $Z^{(2,3)T}$, $Z^{(3,4)}$ & $Z^{(3,4)T}$, ...
- The ADEF1 preconditioner $P_{h,ADEF1}$ is denoted by P^j where subscript distinguishes the level at which preconditioner is applied.
- For rest of the operators and vectors, the subscript identifies the level of them.

Making use of these simplified notations for the operators involved in ADEF1 preconditioner, we now write the multilevel implementation of ADEF1 as Algorithm 2.

4.7 Two-Level Krylov Method (TLKM)

Theoretically, the deflation preconditioner should be based on the linear system whose coefficient matrix carries unfavorable eigenvalues, which cause slow convergence. Before focusing on such a type of preconditioner, which is reported in [42, 41], we consider the preconditioned Helmholtz operator, the problem of our interest,

$$\tilde{A}_h = M_{h,(\beta_1,\beta_2)}^{-1}A_h. \quad (4.26)$$

Recalling from the previous chapter, we observed that the spectrum of $M_{h,(\beta_1,\beta_2)}^{-1}A_h$ encounters small eigenvalues around origin particularly for large wave numbers. The deflation preconditioner is meant to treat those small eigenvalues. In view of theoretical characteristics of the deflation preconditioner, theoretically the deflation preconditioner should be based on preconditioned operator $M_{h,(\beta_1,\beta_2)}^{-1}A_h$ instead of the operator A_h . The development of this type of preconditioner can be seen by replacing A_h by \tilde{A}_h in the ADEF1 preconditioner 4.20. The resulting preconditioner looks like

$$\tilde{P}_h = I_h - \tilde{A}_h \tilde{Q}_h \text{ where } \tilde{Q}_h = I_h^{2h} \tilde{A}_{2h}^{-1} I_h^{2h} \text{ and } \tilde{A}_{2h} = I_h^{2h} \tilde{A}_h I_h^{2h}. \quad (4.27)$$

The preconditioner (4.27) deflates the the troublesome eigenvalues to 0 effectively. But the need of shifting of deflated-spectrum from the origin to a certain value γ , as discussed in previous sections, is inevitable for large problems. Therefore, we perform a shift of the deflated spectrum. With the shifting term, the preconditioner is given by

$$\tilde{P}_{h,\lambda_n} = \tilde{P}_h + \gamma \tilde{Q}_h = (I_h - \tilde{A}_h \tilde{Q}_h) + \gamma \tilde{Q}_h \quad (4.28)$$

As the construction of preconditioner (4.28) suggests, it is intended to be applied on Helmholtz system already preconditioned by CSLP i.e.

$$\tilde{P}_{h,\lambda_n} \tilde{A}_h u_h = \tilde{P}_{h,\lambda_n} \tilde{b}_h \quad (4.29)$$

where $\tilde{b}_h = M_{h,(\beta_1,\beta_2)}^{-1} b_h$. If we look exclusively at the original Helmholtz system, the preconditioner application will be read as

$$P_{h,TLKM} = \tilde{P}_{h,\lambda_n} M_{h,(\beta_1,\beta_2)}^{-1} \quad (4.30)$$

Essentially, this is the first variant **V1** as briefly defined earlier. From here onwards, we will call this preconditioner as the two level Krylov method (TLKM) preconditioner.

Algorithm 2 Implementation ADEF1 Preconditioner $P_{h,ADEF1} = M_{h,(\beta_1, \beta_2)}^{-1} P_h + Q_h$

1: Construct $A^{(1)}, M^{(1)}$

2: **for** $i:=1,2,\dots, m$, the coarsest level **do** • m is number of levels

3: Construct $Z^{(i,i+1)}$ and $Z^{(i,i+1)T}$,

4: Compute $A^{(i+1)} = Z^{(i,i+1)} A^{(i)} Z^{(i,i+1)T}$

5: Compute $M^{(i+1)} = Z^{(i,i+1)} M^{(i)} Z^{(i,i+1)T}$

6: **end for**

7: Start: $i = 1$;

8: **Solve:** $A^{(1)} u^{(1)} = b^{(1)}$ with Krylov preconditioned by $P^{(1)}$ • $P^{(i)} = (M^{(i)})^{-1} (I - A^{(i)} Q^{(i)}) + Q^{(i)}$

9: • $Q^{(i)} = Z^{(i,i+1)} (A^{(i)})^{-1} Z^{(i,i+1)T}$

10: $u^{(1)}$, vector to be preconditioned

11: **Restriction:** $\hat{u}^{(2)} = Z^{(1,2)T} u^{(1)}$

12: **if** $m = 1$ **then**

13: $u^{(2)} = (A^{(2)})^{-1} \hat{u}^{(2)}$ with direct solver

14: **else**

15: $i=2$

16: **Solve:** $A^{(2)} u^{(2)} = \hat{u}^{(2)}$ with Krylov preconditioned by $P^{(2)}$

17: $u^{(2)}$, vector to be preconditioned

18: **Restriction:** $\hat{u}^{(3)} = Z^{(2,3)T} u^{(2)}$

19: **if** $m = 2$ **then**

20: $u^{(3)} = (A^{(3)})^{-1} \hat{u}^{(3)}$ with direct solver

21: **else**

22: $i=3$

23: **Solve:** $A^{(3)} u^{(3)} = \hat{u}^{(3)}$ with Krylov preconditioned by $P^{(3)}$

24: ...

25: ...

26: ...

27: **end if**

28: **Interpolation:** $q^{(2)} = Z^{(2,3)} u^{(3)}$

29: $\hat{t}^{(2)} = u^{(2)} - A^{(2)} q^{(2)}$

30: $t^{(2)} = (M^{(2)})^{-1} \hat{t}^{(2)}$ by multigrid

31: $w^{(2)} = t^{(2)} + q^{(2)}$

32: $p^{(2)} = A^{(2)} w^{(2)}$

33: **end if**

34: **Interpolation:** $q^{(1)} = Z^{(1,2)} u^{(2)}$

35: $\hat{t}^{(1)} = u^{(1)} - A^{(1)} q^{(1)}$

36: $t^{(1)} = (M^{(1)})^{-1} \hat{t}^{(1)}$ by multigrid

37: $w^{(1)} = t^{(1)} + q^{(1)}$

38: $p^{(1)} = A^{(1)} w^{(1)}$

39: **End**

4.7.1 Multi-Level Krylov Method (MLKM)

With the same set of arguments used while extending the ADEF1 preconditioner to multilevel, the TLKM preconditioner can also be extended to multilevel. Unfortunately, unlike the extension of ADEF1 to multilevel, the extension of the TLKM is more complicated. The idea of using CSLP preconditioned Helmholtz \tilde{A}_h instead of A_h for the development of the MLKM preconditioner, particularly for Galerkin coarse grid operator \tilde{A}_{2h} , increases its complexity. To get insights of how the operator $\tilde{A}_{2h} = I_h^{2h} \tilde{A}_h I_h^h$ is applied on a vector, we will consider the application of the entire MLKM preconditioner on the (Arnoldi) vector inside Krylov method by considering the preconditioning step;

$$\begin{aligned}\tilde{w} &= \tilde{P}_{h,\gamma} \tilde{v} \\ &= \left(I - \tilde{A}_h I_{2h}^h \tilde{A}_{2h}^{-1} I_h^{2h} + \gamma I_{2h}^h \tilde{A}_{2h}^{-1} I_h^{2h} \right) \tilde{v}, \\ &= \left(\tilde{v} - (\tilde{A}_h - \gamma I) \tilde{v}' \right),\end{aligned}\tag{4.31}$$

where

$$\tilde{v}' = \left(I_{2h}^h A_{2h}^{-1} I_h^{2h} \right) \tilde{v}\tag{4.32}$$

Vector \tilde{v} is the vector to be preconditioned at every iteration of the Krylov subspace method. We get vector \tilde{v}' by application of matrix $\tilde{Q}_h = I_{2h}^h A_{2h}^{-1} I_h^{2h}$ in 4.32. Initially, vector \tilde{v} is restricted by I_h^{2h} , as seen in Equation (4.31),

$$\tilde{v}_R = I_h^{2h} \tilde{v}.\tag{4.33}$$

With \tilde{v}_R , we have to solve the Galerkin system with coefficient matrix \tilde{A}_{2h} as given in (4.32), and that can be written as

$$\nu'_R = \tilde{A}_{2h}^{-1} \nu_R \iff \tilde{A}_{2h} \nu'_R = \tilde{v}_R \text{ (solve for } \nu'_R)\tag{4.34}$$

4.7.2 Coarse Grid Operator Complexity

To solve ν'_R from the coarse system

$$\tilde{A}_{2h} \nu'_R = \nu_R$$

the coefficient matrix \tilde{A}_{2h} is required, which is obtained by the Galerkin method using the preconditioned Helmholtz matrix \tilde{A}_h i.e.

$$\tilde{A}_{2h} = I_h^{2h} \tilde{A}_h I_h^h\tag{4.35}$$

It is apparent to observe that the explicit construction of \tilde{A}_{2h} requires the action of $M_{h,(\beta_1,\beta_2)}^{-1}$ on each column of A_h , which is computationally not practical. The inverse matrix $M_{h,(\beta_1,\beta_2)}^{-1}$ is implicitly available via a multigrid cycle. This can be utilized while employing a Krylov subspace method to solve ν'_R from the coarse system

$$\left(I_h^{2h} M_{h,(\beta_1,\beta_2)}^{-1} A_h I_h^h \right) \nu'_R = \nu_R,$$

where $I_h^{2h} M_{h,(\beta_1, \beta_2)}^{-1} A_h I_{2h}^h$ is considered as coefficient matrix. In this way, we can achieve the multilevel algorithm but at every Krylov iteration, the coarse solve will cost the approximation of $M_{h,(\beta_1, \beta_2)}^{-1}$, which is again impractical to use. Both approaches to construct \tilde{A}_{2h} make use of the inverse $M_{h,(\beta_1, \beta_2)}^{-1}$ implicitly or explicitly. With this choice of construction of \tilde{A}_{2h} , the computational complexity suggests that multilevel implementation is simply impractical in terms of cost per Krylov iteration. We call the TLKM preconditioner, which make use of \tilde{A}_{2h} , the *ideal variant* of the TLKM preconditioner.

In order to make the multilevel implementation suitable, an approximation of the inverse of CSLP operator $M_{h,(\beta_1, \beta_2)}^{-1}$ has been suggested in [42], $M_{h,(\beta_1, \beta_2)}^{-1}$ can be approximated by $I_{2h}^h (I_{2h}^{2h} M_{h,(\beta_1, \beta_2)} I_{2h}^h)^{-1} I_h^{2h}$. This results in an adaptation of the coarse grid matrix as follows

$$\begin{aligned}
 \tilde{A}_{2h} &= I_h^{2h} \tilde{A}_h I_{2h}^h \\
 &= I_h^{2h} M_h^{-1} A_h I_{2h}^h \\
 &\approx I_h^{2h} I_{2h}^h (I_h^{2h} M_h I_h^{2h})^{-1} I_h^{2h} A_h I_{2h}^h \\
 &= I_h^{2h} I_{2h}^h M_{2h}^{-1} A_{2h} \\
 &= I_h^{2h} I_{2h}^h \hat{A}_{2h} = \Theta_h \hat{A}_{2h}.
 \end{aligned} \tag{4.36}$$

where $\Theta_h = I_h^{2h} I_{2h}^h$ is an approximation term. Replacing \tilde{A}_{2h} with this approximation, the coarse system can be written as

$$\Theta_h M_{2h}^{-1} A_{2h} v'_R = v_R, \tag{4.37}$$

where A_{2h} , M_{2h} and Θ_h are the Galerkin matrices associated with matrices A_h , $M_{h,(\beta_1, \beta_2)}$ and I_h the identity matrix. One can notice that the approximation of the coarse grid operator in Preconditioner 4.30 as

$$\hat{A}_{2h} = I_h^{2h} \tilde{A}_h I_{2h}^h \approx I_h^{2h} I_{2h}^h M_{2h}^{-1} A_{2h}$$

turns the preconditioner practical to use and coincides with third variant **V3**

4.7.3 MLKM: Multilevel Implementation

The Krylov subspace method of choice can be used to solve the adapted coarse system given in Equation 4.37. Though the coefficient matrix in the coarse system is the Galerkin matrix A_{2h} preconditioned by Galerkin CSLP operator M_{2h} but it also involves the approximation term Θ_h . Thus, it does not necessarily have favorable spectrum for a Krylov method. To aim for fast Krylov convergence, the MLKM preconditioner 4.28 is applied. Which will give rise to an other coarse system at level Ω_{4h} . A recursive application of the deflation Preconditioner 4.28 at subsequent coarser levels immediately results in a *practical variant* of MLKM preconditioner. Implementation

of this MLKM Practical (MLKM-P) preconditioner is explained in Algorithm 3. For the archetype of the algorithm, the reader is referred to [42].

Comparatively, the MLKM preconditioner is more expensive and complex to implement than the ADEF1 preconditioner from Equation (4.20). We will explore how effective this approximation is by the analysis of the TLKM in the next chapter.

4.8 Complexity Analysis

In this section, we give a brief comparison of the deflation variants defined above. We claim that the ADEF1 preconditioner is relatively simple to implement when compared to both TLKM deflation variants; ideal and practical. In Table 4.8, we present an overview of the operations of an application of ADEF1 and TLKM ideal (TLKM-I) and practical (TLKM-P) preconditioners. A two-level application of these preconditioners is considered. The inverse of the CSLP operator M_h can be obtained by any means; say by multigrid or by a direct solver. Purpose is to highlight the costs of the operations for different variants of two-level applications. Apparently, the TLKM-I preconditioner costs three applications of M_h^{-1} . This also indicates that it is not feasible for multilevel extension, as for every coarse grid iteration, it is required to apply M_h^{-1} . The TLKM-P variant costs two applications of M_h^{-1} , which is a significant difference with ADEF1, which costs only one such application. Approximation of the coarse grid operator in TLKM-P also costs two operations as given in Table 4.8 and this can be more complicated while moving to a multilevel implementation. The additional costs for the two-level application of ADEF1, when comparing with the costs of the CSLP preconditioner, is justified with the results presented in Chapter 6. Meanwhile, we can say that the ADEF1 preconditioner is favorable one among the deflation variants in context of the application costs at two-levels. This is further illustrated in Figure 4.8. Iterations taken to solve a one dimensional Helmholtz problem are compared for three different deflation variants. TLKM-I takes less iterations, but as we stated this is only useful for theoretical purposes. For large problems, the two-level preconditioner is not feasible and multilevel extension is required. TLKM-I is not suitable for a multilevel extension where as TLKM-P can be extended to multilevel easily. The TLKM-P and ADEF1 preconditioners are fairly compared, while the latter is computationally cheaper than the earlier.

4.9 Concluding Remarks

In the earlier sections, the ADEF1 preconditioner is introduced for the Helmholtz problem. ADEF1 is a simple and user-friendly preconditioner. We have shown via Algorithm 2 that it is easy to implement. It is solely based on the Helmholtz operator

Operation	CSLP	CSLP with Deflation		
		ADEF1	TLKM-I	TLKM-P
Vector Sum	0	2	2	2
$A_h v_h$	0	1	1	1
$M_h^{-1} v_h$	1	1	3	2
$Q_h v_h$	$I_h^{2h} v_h$	0	1	1
	$I_{2h}^h v_{2h}$	0	1	1
	$A_{2h}^{-1} / M_{2h}^{-1}$	0	1	0
	$A_{2h} v_{2h}$	0	0	1
	$I_{2h}^h I_{2h}^{2h} v_h$	0	0	1

Table 4.1: Costs per application of a two-level preconditioner in one Krylov iteration.

A_h . It can be considered as a second level preconditioner, as it is applied in combination with CSLP. Later, the TLKM ideal variant preconditioner is introduced which is based on the preconditioned Helmholtz operator \tilde{A}_h , rather than A_h . The TLKM preconditioner is more extensive and complicated to implement. Indeed the ideal TLKM involves

$$\tilde{A}_{2h} = I_h^{2h} \left(M_{h,(\beta_1, \beta_2)}^{-1} A_h \right) I_{2h}^h$$

Explicit unavailability of M^{-1} makes it nearly impossible to implement the ideal variant. Every application of \tilde{A}_{2h} costs the implicit inversion of $M_{h,(\beta_1, \beta_2)}$, which can not be justified. With this costs, it is not a practical to implement. The coarse grid operator \tilde{A}_{2h} has been adapted with an approximation term

$$\tilde{A}_{2h} = I_h^{2h} I_{2h}^h M_{2h}^{-1} A_{2h}.$$

with this approximation, it is likely to extend the application of TLKM in a multilevel fashion.

In the next chapter, the performance analysis has be performed for these deflation preconditioners.

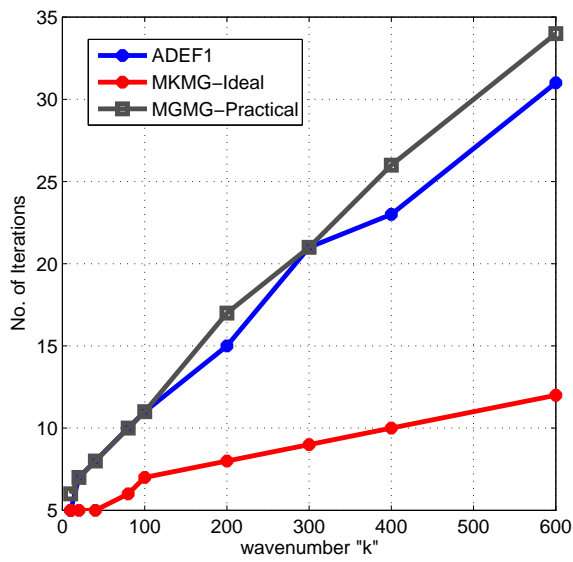


Figure 4.3: Comparing iterations to solve one dimensional Helmholtz with two-level solvers ADEF1 and TLKM ideal and practical variants.

Algorithm 3 Implementation MLKM-P Preconditioner $P_{h,TLKM} = M_{h,(\beta_1, \beta_2)}^{-1} P_h + Q_h$

- 1: Construct $A^{(1)}$, $M^{(1)}$ and $\Theta^{(1)} = I$
 - 2: Precondition matrix $\tilde{A}^{(1)} = (M^{(1)})^{-1}A^{(1)}$, and right-hand side vector $\tilde{b}^{(1)} = (M^{(1)})^{-1}b^{(1)}$
 - 3: **for** $i=1,2,\dots, m$, the coarsest level **do** • m is number of levels
 - 4: Construct $Z^{(i,i+1)}$ and $Z^{(i,i+1)T}$,
 - 5: Compute $A^{(i+1)} = Z^{(i,i+1)}A^{(i)}Z^{(i,i+1)T}$
 - 6: Compute $M^{(i+1)} = Z^{(i,i+1)}M^{(i)}Z^{(i,i+1)T}$
 - 7: Compute $\Theta^{(i+1)} = Z^{(i,i+1)}Z^{(i,i+1)T}$
 - 8: **end for**
 - 9: Compute $\tilde{A}^{(m)} = \Theta^{(m)}(M^{(m)})^{-1}A^{(m)}$
 - 10: Start: $i = 1$;
 - 11: Θ_h
 - 12: **Solve:** $\tilde{A}^{(1)}u^{(1)} = \tilde{b}^{(1)}$ with Krylov preconditioned by $\tilde{P}^{(1)}$ • $\tilde{P}^{(i)} = (I - \tilde{A}^{(i)}\tilde{Q}^{(i)}) + \tilde{Q}^{(i)}$
 - 13: • $\tilde{Q}^{(i)} = Z^{(i,i+1)}(\tilde{A}^{(i)})^{-1}Z^{(i,i+1)T}$
 - 14: $u^{(1)}$, vector to be preconditioned
 - 15:
 - 16: **Restriction:** $b^{(2)} = Z^{(1,2)T}u^{(1)}$
 - 17: **if** $m = 2$ **then**
 - 18: $u_c^{(2)} = (\tilde{A}^{(2)})^{-1}b^{(2)}$ with direct solver
 - 19: **else**
 - 20: $i=2$
 - 21: **Solve:** $\tilde{A}^{(2)}u_c^{(2)} = b^{(2)}$ with Krylov preconditioned by $P^{(2)}$
 - 22: $u^{(2)}$, vector to be preconditioned
 - 23: **Restriction:** $b^{(3)} = Z^{(2,3)T}u^{(2)}$
 - 24: **if** $m = 3$ **then**
 - 25: $u^{(3)} = (\tilde{A}^{(3)})^{-1}b^{(3)}$ with direct solver
 - 26: **else**
 - 27: $i=3$
 - 28: **Solve:** $\tilde{A}^{(3)}u^{(3)} = \hat{u}^{(3)}$ with Krylov preconditioned by $P^{(3)}$
 - 29:
 - 30: **end if**
 - 31: **Interpolation:** $q^{(2)} = Z^{(2,3)}u^{(3)}$
 - 32: $\hat{\gamma}^{(2)} = u^{(2)} - A^{(2)}q^{(2)}$
 - 33: $t^{(2)} = (M^{(2)})^{-1}\hat{\gamma}^{(2)}$ by multigrid
 - 34: $w^{(2)} = t^{(2)} + q^{(2)}$
 - 35: $p^{(2)} = A^{(2)}w^{(2)}$
 - 36: **end if**
 - 37: **Interpolation:** $t^{(1)} = Z^{(1,2)}u_c^{(2)}$
 - 38: $s^{(1)} = v^{(1)} - t_M^{(1)} + \gamma t^{(1)}$ • $t_M^{(1)} = (M^{(1)})^{-1}t_A^{(1)} = (M^{(1)})^{-1}(A^{(1)}t^{(1)})$
 - 39: $w^{(1)} = (M^{(1)})^{-1}s^{(1)}$ by multigrid
 - 40: $p^{(1)} = A^{(1)}w^{(1)}$
 - 41: **End**
-

Spectral Analysis

In this chapter we perform the rigorous Fourier analysis of the Helmholtz operator preconditioned by CSLP in combination with variants of the deflation preconditioner defined in earlier chapter. Recall that base deflation preconditioner is $P = I - AQ$, where $Q = ZE^{-1}Z^T$ and Z is the deflation matrix. The preconditioner projects the eigenvalues, related with the columns of the deflation matrix, to 0. Multilevel implementation of deflation preconditioner (in-exact inverse of matrix E) might cause a number of eigenvalues around 0. In order to make multilevel implementation possible, a shift term Q was introduced and the deflation preconditioner was integrated with that shift term as $P = I - AQ + Q$. This shift term is expected to shift the deflated part of the spectrum to 1 by another number. This shift can be adapted by replacing 1. We disregard the shift for analysis purpose. We perform analysis without including shift term unless mention explicitly, which we shall discuss to confirm the action of the shift term Q in later sections. Further, we discuss the variants ADEF1 and TLKM in details, where as in the end, we will highlight the rest of variants broadly.

5.1 Rigorous Fourier Analysis Framework

Traditionally Fourier analysis has been used to analyse the performance of multigrid methods. The deflation preconditioner

$$\mathcal{P}_h = I - \mathcal{A}_h Q_h \text{ where } Q_h = Z_h E^{-1} Z_h^T \text{ and } E = Z_h^T \mathcal{A}_h Z_h$$

defined in 4.2, coincides with the coarse correction operator of multigrid, i.e.

$$P_{h,2h} = I_h - A_h Q_h \text{ where } Q_h = I_h^{2h} A_{2h}^{-1} I_{2h}^h \text{ and } A_{2h} = I_h^{2h} A_h I_{2h}^h \quad (5.1)$$

if on a two-dimensional discrete domain Ω_h with a grid-step h in each direction, the deflation matrix Z_h is set as the prolongation operator I_h^{2h} , which is obtained from interpolation from grid Ω_{2h} to Ω_h . Application of this deflation preconditioner along with CSLP $M_{h,(\beta_1,\beta_2)}$ defined in Equation 3.2 results in a *two-grid method*. The resemblance of the two-grid method with a two-grid cycle motivates us to use Fourier mode analysis to analyze our CSLP and deflation preconditioners in terms of spectrum. Applying CSLP (3.2) to the deflated linear system requires solving a system with $M_{h,(\beta_1,\beta_2)}^{-1}P_{h,2h}A_h$ as coefficient matrix. Applying the deflation operator (5.1) to the CSLP-preconditioned system instead leads to a system with $P_{h,2h}^T M_{h,(\beta_1,\beta_2)}^{-1}A_h$ as coefficient matrix. The sequence of application of two preconditioners is not sensitive in the terms that two aforementioned coefficient matrices have the same spectrum i.e.

$$\sigma\left(M_{h,(\beta_1,\beta_2)}^{-1}P_{h,2h}A_h\right) = \sigma\left(P_{h,2h}^T M_{h,(\beta_1,\beta_2)}^{-1}A_h\right). \quad (5.2)$$

The ordering of the operators in the left-hand side is typically chosen in implementation in practical codes. In Fourier analysis, however, ordering, as appears in right-hand side, is used. For the proof of identity 5.2, the reader is referred to [111]. In this analysis, the deflated-preconditioned operator is denoted as

$$B_{h,2h,(\beta_1,\beta_2)} = P_{h,2h}^T M_{h,(\beta_1,\beta_2)}^{-1}A_h. \quad (5.3)$$

The analysis of the operator $B_{h,2h,(\beta_1,\beta_2)}$ can be seen as an analysis of a two-grid operator, where $M_{h,(\beta_1,\beta_2)}$ is considered as smoother and $P_{h,2h}^T$ as coarse grid correction. The rigorous Fourier analysis in sequel of this chapter shows that, in case that both the CSLP $M_{h,(\beta_1,\beta_2)}$ and the coarse grid operator A_{2h} are inverted exactly, a basis of discrete sine modes exists that allows us to diagonalize both A_h and $M_{h,(\beta_1,\beta_2)}$ and to block diagonalize $P_{h,2h}$. We will also make use of the following notations

$$\text{diag} \begin{bmatrix} d_1 \\ d_2 \end{bmatrix} = \begin{bmatrix} d_1 & 0 \\ 0 & d_2 \end{bmatrix} \quad (5.4)$$

and

$$\text{diag} \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{bmatrix} = \begin{bmatrix} d_1 & 0 & 0 & 0 \\ 0 & d_2 & 0 & 0 \\ 0 & 0 & d_3 & 0 \\ 0 & 0 & 0 & d_4 \end{bmatrix}. \quad (5.5)$$

5.2 One-Dimensional Rigorous Two-Grid Analysis of ADEF1

In this section we perform a rigorous two-grid Fourier analysis of the solver proposed above in a one-dimensional setting. We therefore consider the Helmholtz equation (2.8) supplied with homogeneous Dirichlet boundary conditions defined in 2.8 discretized on $\Omega_h = \{0, h, 2h, \dots, 1\}$, where the step-size $h = \frac{1}{n}$. For sake of convenience,

we will limit n as even integer. Dirichlet boundary conditions are easy to analyze. Further motivation is, as these boundary conditions do not introduce any damping, the analysis that follows can be considered to be a worst case analysis for the problem with Sommerfeld boundary conditions. For sake of the analysis, the discretization of the Helmholtz equation is done on a uniform mesh $\Omega_h \subset (0, 1)$ with mesh width $h = 1/n$ using the second order accurate stencil

$$[A_h] = \frac{1}{h^2} \begin{bmatrix} -1 & 2 - \kappa^2 & -1 \end{bmatrix} \text{ where } \kappa = kh, \quad (5.6)$$

results after elimination of the boundary conditions in the linear system

$$A_h x_h = b_h,$$

where $A_h \in \mathbb{C}^{(n-1) \times (n-1)}$. Given the preconditioner $M_{h,(\beta_1, \beta_2)}$ defined in (3.2) and the multigrid deflation operator $P_{h,2h}$ defined in (5.1), the goal of our analysis is to find all eigenvalues of the deflated preconditioned operator $B_{h,2h,(\beta_1, \beta_2)}$. In this analysis, it is assumed that both the preconditioner $M_{h,(\beta_1, \beta_2)}$ as well as the coarse grid matrix A_{2h} are inverted exactly.

If the vector $\underline{x} \in \mathbb{R}^{n-1}$ carries the components $x_i = ih$, then one can easily observe that the grid vectors

$$\phi_h^\ell = \sin(\ell\pi\underline{x}) \text{ for } 1 \leq \ell \leq n-1 \quad (5.7)$$

are eigenvectors of A_h corresponding to the eigenvalues

$$\lambda^\ell(A_h) = \frac{1}{h^2} (2 - 2c_\ell - \kappa^2), \quad (5.8)$$

where $c_\ell = \cos(\ell\pi h)$.

5.2.1 Basis Diagonalization

The eigenvectors $\{\phi_h^\ell = \sin(\ell\pi\underline{x}) \text{ for } 1 \leq \ell \leq n-1\}$ form a set of orthonormal vectors. The matrix V_h with these orthonormal vectors as its columns satisfies the relation $V_h^T = V_h^{-1}$. This orthonormal basis diagonalizes the operator 5.3 i.e. $B_{h,2h,(\beta_1, \beta_2)}$ and transforms it into block diagonal matrix as follows

$$B_{h,2h,(\beta_1, \beta_2)} \xrightarrow{\text{Diagonalization}} [B_{h,2h,(\beta_1, \beta_2)}^\ell]_{1 \leq n/2} \quad (5.9)$$

where each block $[B_{h,2h,(\beta_1, \beta_2)}^\ell]_{1 \leq n/2}$ is a combination of blocks of all involved matrices i.e.

$$B_{h,2h,(\beta_1, \beta_2)}^\ell = (P_{h,2h}^T)^\ell (S_h)^\ell$$

where S_h^ℓ is conveniently denotes the blocks of $M_{h,(\beta_1, \beta_2)}^{-1} A_h$ and are given by

$$S_h^\ell = (M_{h,(\beta_1, \beta_2)}^{-1} A_h)^\ell. \quad (5.10)$$

It is worth to mention that the operator $B_{h,2h,(\beta_1,\beta_2)}$ and blocks $[B_{h,2h,(\beta_1,\beta_2)}^\ell]_{1 \leq \ell \leq n/2}$ have the same spectrum i.e.

$$\sigma(B_{h,2h,(\beta_1,\beta_2)}) = \sigma\left([B_{h,2h,(\beta_1,\beta_2)}^\ell]_{1 \leq \ell \leq n/2}\right) \quad (5.11)$$

Diagonalization of A_h : In what follows we will start diagonalizing the Helmholtz operator A_h , CSLP $M_{h,(\beta_1,\beta_2)}$ and deflation operator $P_{h,2h}$ defined by (2.22), (3.2) and (5.1), respectively. The derivation of blocks will lead to closed form expressions for the eigenvalues of the two-grid preconditioner $B_{h,2h,(\beta_1,\beta_2)}$ defined by 5.3. We therefore proceed in the standard way [113] and reorder the eigenvectors according to

$$V_h = [\phi_h^1, \phi_h^{n-1}, \phi_h^2, \phi_h^{n-2}, \dots, \phi_h^{n/2-1}, \phi_h^{n/2}]. \quad (5.12)$$

This basis brings A_h into a block diagonal form that can be written as

$$A_h = [A_h^\ell]_{1 \leq \ell \leq n/2}, \quad (5.13)$$

where for $1 \leq \ell \leq n/2 - 1$, A_h^ℓ is the 2×2 diagonal block

$$A_h^\ell = \begin{bmatrix} \frac{1}{h^2}(2 - 2c_{\ell_1} - \kappa^2) & 0 \\ 0 & \frac{1}{h^2}(2 + 2c_{\ell_1} - \kappa^2) \end{bmatrix}, \quad (5.14)$$

and $A_h^{n/2}$ is the 1×1 block

$$A_h^{n/2} = \frac{2}{h^2} - \kappa^2. \quad (5.15)$$

Diagonalization of $M_{h,(\beta_1,\beta_2)}$ and $S_{h,(\beta_1,\beta_2)}$: The re-ordered grid vectors 5.12 are also the eigenvectors of the preconditioner $M_{h,(\beta_1,\beta_2)}$ and subsequently of the preconditioned operator (or smoother) $S_{h,(\beta_1,\beta_2)}$. As we already know the eigenvalues of Helmholtz operator A_h , the corresponding eigenvalues of the operator $M_{h,(\beta_1,\beta_2)}$ and $S_{h,(\beta_1,\beta_2)}$ will be

$$\lambda^\ell(M_{h,(\beta_1,\beta_2)}) = \frac{1}{h^2} \left(2 - 2c_\ell - \kappa^2(\beta_1 - \iota\beta_2) \right), \quad (5.16)$$

and

$$\lambda^\ell(S_{h,(\beta_1,\beta_2)}) = \frac{2 - 2c_\ell - \kappa^2}{2 - 2c_\ell - \kappa^2(\beta_1 - \iota\beta_2)}, \quad (5.17)$$

respectively.

The diagonalization of $M_{h,(\beta_1,\beta_2)}$ and $S_{h,(\beta_1,\beta_2)}$ with basis vectors given in 5.12 immediately resulted in blocks

$$M_{h,(\beta_1,\beta_2)} \xrightarrow{\text{Diagonalization}} [M_{h,(\beta_1,\beta_2)}^\ell]_{1 \leq \ell \leq n/2}, \quad (5.18)$$

and

$$S_{h,(\beta_1,\beta_2)} \xrightarrow{\text{Diagonalization}} [S_{h,(\beta_1,\beta_2)}^\ell]_{1 \leq \ell \leq n/2}. \quad (5.19)$$

The spectrum of operators $M_{h,(\beta_1,\beta_2)}$ and $S_{h,(\beta_1,\beta_2)}$ and their respective block representation in Equations 5.18 and 5.19 is same.

For $1 \leq \ell \leq n/2 - 1$, the respective blocks are given by

$$M_{h,(\beta_1,\beta_2)}^\ell = \frac{1}{h^2} \begin{bmatrix} \frac{1}{h^2}(2 - 2c_\ell - (\beta_1 - \iota\beta_2)\kappa^2) & 0 \\ 0 & \frac{1}{h^2}(2 + 2c_\ell - (\beta_1 - \iota\beta_2)\kappa^2) \end{bmatrix}, \quad (5.20)$$

and

$$S_{h,(\beta_1,\beta_2)}^\ell = \begin{bmatrix} \frac{2-2c_\ell-\kappa^2}{2-2c_\ell-(\beta_1-\iota\beta_2)\kappa^2} & 0 \\ 0 & \frac{2+2c_\ell-\kappa^2}{2+2c_\ell-(\beta_1-\iota\beta_2)\kappa^2} \end{bmatrix}. \quad (5.21)$$

For the value of $\ell = n/2$, the 1×1 blocks of $M_{h,(\beta_1,\beta_2)}$ and \hat{A}_h are

$$M_h^{n/2} = \frac{2}{h^2} - (\beta_1 - \iota\beta_2)\kappa^2$$

and

$$S_h^{n/2} = \frac{2 - \kappa^2}{2 - (\beta_1 - \iota\beta_2)\kappa^2}.$$

Diagonalization of inter-grid operators and coarse grid operator: We have made a choice for the deflation matrix. The prolongation operator, from the multigrid methods, defined in a stencil 4.14 in the previous chapter are considered. For a one dimensional problem, the prolongation operator I_{2h}^h will be linear interpolation and for the restriction operator I_h^{2h} we will use the full weighting operator, which is defined by the stencil

$$[I_h^{2h}] = \left[\frac{1}{4} \quad \frac{1}{2} \quad \frac{1}{4} \right], \quad (5.22)$$

Given the fact that the basis (5.12) diagonalizes the linear interpolation operator $I_h^{2h} \in \mathbb{R}^{n \times (\frac{n}{2}-1)}$ into blocks

$$(I_h^{2h})^\ell = \left[\frac{1}{2}(1 + c_\ell) \quad -\frac{1}{2}(1 - c_\ell) \right], \quad (5.23)$$

for $1 \leq \ell \leq n/2 - 1$ and $(I_h^{2h})^{n/2} = 0$, and that the inter-grid transfer operators are related by $I_{2h}^h = (I_h^{2h})^T$, the 1×1 diagonal blocks of the Galerkin coarse grid operator A_{2h} and M_{2h} can shown to be equal to

$$A_{2h}^\ell = (I_h^{2h})^\ell A_h^\ell (I_{2h}^h)^\ell = \frac{2(1 - c_\ell^2) - \kappa^2(1 + c_\ell^2)}{2h^2}, \quad (5.24)$$

and

$$M_{2h}^\ell = (I_{2h}^h)^\ell M_h^\ell (I_{2h}^h)^\ell = \frac{2(1 - c_\ell^2) - (\beta_1 - \iota\beta_2)\kappa^2(1 + c_\ell^2)}{2h^2}, \quad (5.25)$$

for $1 \leq \ell \leq n/2 - 1$ and $(A_{2h})^{n/2} = \frac{2-\kappa^2}{2h^2}$.

Subsequently the 1×1 block for $M_{2h}^{-1}A_{2h}$ will be

$$(M_{2h}^{-1}A_{2h})^\ell = \frac{2(1 - c_\ell^2) - \kappa^2(1 + c_\ell^2)}{2(1 - c_\ell^2) - (\beta_1 - \iota\beta_2)\kappa^2(1 + c_\ell^2)}, \quad (5.26)$$

for $1 \leq \ell \leq n/2 - 1$ and $(M_{2h}^{-1}A_{2h})^{n/2} = \frac{2 - \kappa^2}{2 - \kappa^2(\beta_1 - \iota\beta_2)}$.

Also the 1×1 block for the approximation term $B_h = I_h^{2h}I_{2h}^h$ can be simplified as

$$B_h^\ell = (I_h^{2h}I_{2h}^h)^\ell = (1 + c_\ell^2), \quad (5.27)$$

for $1 \leq \ell \leq n/2$.

The ADEF1 deflation (or coarse grid correction) operator $P_{h,2h}^T$ employs a coarser grid with mesh width $H = 2h$ with as inter-grid transfer and coarser operators the one-dimensional variants of those described above. By reordering the eigenvectors of A_h in a standard way in $(\ell, n - \ell)$ pairs [113], we obtain the basis

$$V_h = \{(\phi_h^\ell, \phi_h^{n-\ell}) \mid \ell = 1, \dots, n/2 - 1\} \cup \{\phi_h^{n/2}\} \quad (5.28)$$

in which $P_{h,2h}^T$ can be written in a block diagonal form, i.e., can be written

$$P_{h,2h}^T \xrightarrow{\text{Diagonalization}} [(P_{h,2h}^\ell)^T]_{1 \leq \ell \leq n/2}, \quad (5.29)$$

where the individual blocks are given by

$$P_{h,2h}^\ell = I - (I_{2h}^h)^\ell (A_{2h}^\ell)^{-1} (I_h^{2h})^\ell A_h^\ell. \quad (5.30)$$

A standard computation gives the 2×1 blocks of the bilinear interpolation where for $1 \leq \ell \leq n/2 - 1$

$$(I_{2h}^h)^\ell = \frac{1}{2} \begin{bmatrix} (1 + c_\ell) \\ -(1 - c_\ell) \end{bmatrix}, \quad (5.31)$$

and where

$$(I_{2h}^h)^{n/2} = 0. \quad (5.32)$$

The Galerkin coarsening then results in the 1×1 blocks where for $1 \leq \ell \leq n/2 - 1$

$$A_{2h}^\ell = (I_h^{2h})^\ell A_h^\ell (I_{2h}^h)^\ell = \frac{1}{2h^2} [2(1 - c_\ell^2) - \kappa^2(1 + c_\ell^2)] \quad (5.33)$$

and where

$$A_{2h}^{n/2} = 0. \quad (5.34)$$

A straightforward computation subsequently allows to obtain for $1 \leq \ell \leq n/2 - 1$ the following 2×2 blocks of $(P_{h,2h}^\ell)^T$

$$(P_{h,2h}^\ell)^T = \frac{1}{\mathbb{C}} \begin{bmatrix} -(c_\ell + 1)(c_\ell^2 - 1) + \frac{1}{2}\kappa^2(c_\ell^2 - 1) & \frac{1}{2}(c_\ell^2 - 1)(2 + 2c_\ell - \kappa^2) \\ \frac{1}{2}(c_\ell^2 - 1)(-2 + 2c_\ell + \kappa^2) & (c_\ell^2 - 1)(3 + c_\ell) + \frac{1}{2}\kappa^2(c_\ell^2 + 3) \end{bmatrix}$$

where $\zeta = \frac{1}{2(1 - c_\ell^2) + \kappa^2(c_\ell^2 + 1)}$,
and 1×1 block

$$(P_{h,2h}^{n/2})^T = 1. \quad (5.35)$$

5.2.2 Closed-form Expression for Eigenvalues

The basis V_h (5.28) can therefore be used to block diagonalize the deflated preconditioned operator, i.e., we can write

$$B_{h,2h,(\beta_1,\beta_2)} = [B_{h,2h,(\beta_1,\beta_2)}^\ell]_{1 \leq \ell \leq n/2} \quad (5.36)$$

where for $1 \leq \ell \leq n/2 - 1$, $B_{h,2h,(\beta_1,\beta_2)}^\ell$ is the 2×2 matrix

$$B_{h,2h,(\beta_1,\beta_2)}^\ell = (P_{h,2h}^\ell)^T \text{diag} \left[\begin{array}{c} \lambda^\ell(S_{h,(\beta_1,\beta_2)}) \\ \lambda^{n-\ell}(S_{h,(\beta_1,\beta_2)}) \end{array} \right], \quad (5.37)$$

and where

$$B_{h,2h,(\beta_1,\beta_2)}^{n/2} = \lambda^{n/2}(S_{h,(\beta_1,\beta_2)}) = \frac{2 - \kappa^2}{2 - \kappa^2(\beta_1 - i\beta_2)}. \quad (5.38)$$

This block diagonal form renders an analytical computation of the eigenvalues of $B_{h,2h,(\beta_1,\beta_2)}$ feasible and results in the conclusion that $B_{h,2h,(\beta_1,\beta_2)}$ has a zero eigenvalue of multiplicity $n/2 - 1$, the eigenvalue (5.38) and $n/2 - 1$ eigenvalues of the form

$$\lambda^\ell(B_{h,2h,(\beta_1,\beta_2)}) = \frac{a_\ell + ib_\ell}{c_\ell + id_\ell} \text{ for } 1 \leq \ell \leq n/2 - 1, \quad (5.39)$$

where a_ℓ , b_ℓ , c_ℓ and d_ℓ are third order polynomials in κ^2 and given by

$$\begin{aligned} a_\ell &= (-1 - c_\ell^2)\beta_1\kappa^6 + (4\beta_1 + 2 - 2c_\ell^2 + 4c_\ell^2\beta_1)\kappa^4 \\ &\quad + (8c_\ell^2 - 4\beta_1 - 8 + 4c_\ell^4\beta_1)\kappa^2 + (8 - 16c_\ell^2 + 8c_\ell^4) \\ b_\ell &= (1 + c_\ell^2)\beta_2\kappa^6 + (-4c_\ell^2\beta_2 - 4\beta_2)\kappa^4 + (4\beta_2 - 4c_\ell^4\beta_2)\kappa^2 \\ c_\ell &= (\beta_2^2 - \beta_1^2 + c_\ell^2\beta_2^2 - c_\ell^2\beta_1^2)\kappa^6 \\ &\quad + (4\beta_1 - 2c_\ell^2\beta_1^2 + 2c_\ell^2\beta_2^2 + 2\beta_1^2 - 2\beta_2^2 + 4c_\ell^2\beta_1)\kappa^4 \\ &\quad + (8\beta_1c_\ell^2 - 8\beta_1 - 4 + 4c_\ell^4)\kappa^2 + 8c_\ell^4 + 8 - 16c_\ell^2 \\ d_\ell &= (2\beta_1\beta_2 + 2c_\ell^2\beta_1\beta_2)\kappa^6 + (4c_\ell^2\beta_1\beta_2 - 4\beta_1\beta_2 - 4\beta_2 - 4c_\ell^2\beta_2)\kappa^4 \\ &\quad + (8\beta_2 - 8c_\ell^2\beta_2)\kappa^2. \end{aligned}$$

In the remainder of the section, the above expressions will be used to investigate the behavior of the spectrum of $B_{h,2h,(\beta_1,\beta_2)}$ in three parameter studies: varying wave number, varying the number of grid points per wavelength and varying the imaginary part in the shift β_2 . We observe that the zero eigenvalue of $B_{h,2h,(\beta_1,\beta_2)}$ does not influence the convergence of the outer Krylov subspace acceleration, and is therefore left out of

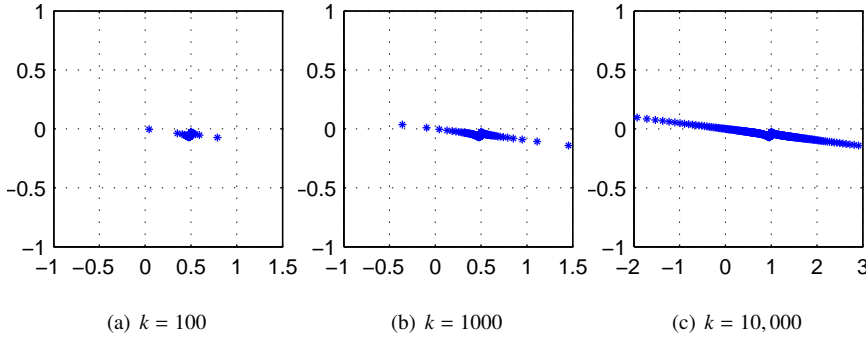


Figure 5.1: Non-zero part of $\sigma(B_{h,2h,(1,1)})$ in the one-dimensional problem for various wavenumbers k satisfying $\kappa = 0.3215$.

our considerations below.

Analysis is performed on a unit domain either in one dimension or two dimensions, where wavenumber k is dimensionless. However, the industrial problems use the parameter frequency f in certain dimensions. The dimensionless wave number \bar{k} is related with the frequency f through the relation defined in Equation 2.24 which is

$$\bar{k} = 2\pi f \bar{l}/c$$

where \bar{l} is the characteristic length, c is the speed of sound. The characteristic length of the physical domain considered for analysis is 1. The presentation of results in wavenumber should be clear to understand. For example if we consider the problem with unit domain and $k = 1000$, then this can convincingly be seen as equivalent to the benchmark Marmousi problem on physical domain of length $l = 9000$ with maximum frequency $f = 100$. In order to clear all ambiguities, we also look into the spectrum for wavenumbers higher than 1000.

Figure (5.1) shows the non-zero part of the spectrum of $B_{h,2h,(\beta_1=1,\beta_2=1)}$ for $k = 100$, $k = 1000$ and $k = 10,000$ using 20 grid points per wavelength (and thus $\kappa = 0.3215$). This figure shows that the spectrum is clustered around one and that the number of small eigenvalues increases as the wavenumber increases. While such a spectrum is favorable for the convergence of an outer Krylov subspace acceleration [88], the increasing number of small eigenvalues prevents the solver from being scalable. This non-scalability becomes even more pronounced in two dimensions as will be confirmed by further analysis and numerical experiments.

Figure (5.2) shows the non-zero spectrum of $B_{h,2h,(\beta_1=1,\beta_2=1)}$ for $k = 2000$ and for $\kappa = 0.625$, $\kappa = 0.312$, and $\kappa = 0.2015$. This figures shows that the number of small eigenvalues decreases, that the spectrum becomes more clustered and the deflated

preconditioned system becomes easier to solve by Krylov subspace methods as the number of grid points per wavelength is increased.

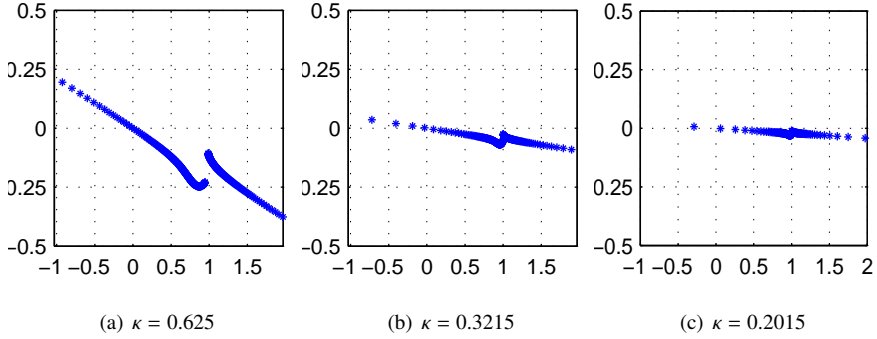


Figure 5.2: Non-zero part of $\sigma(B_{h,2h,(1,1)})$ in the one-dimensional problem for $k = 2000$ and for various values of κ .

Figure (5.3) shows the non-zero spectrum of $B_{h,2h,(\beta_1=1,\beta_2)}$ for $\kappa = 0.625$, for $k = 2000$, and for $\beta_2 = .5$, $\beta_2 = .75$ and $\beta_2 = 1$. This figure shows that the spectrum remains virtually unchanged as β_2 increases. As the CSLP preconditioner becomes more diagonally dominant as β_2 increases, this result opens promising perspectives on obtaining a very good approximation of the preconditioner at low cost.

Figure 5.4 finally shows a bar plot of the modulus of non-zero eigenvalues of $B_{h,2h,(\beta_1=1,\beta_2=1)}$ for $\kappa = 0.625$ and $\kappa = 0.3215$ and for various values of k . Figures does not present all the eigenvalue but part of the eigenvalues which are close to zero in magnitude. These figures clearly show that the number of small eigenvalues grows with k and that this effect is more pronounced in case if a smaller number of grid points

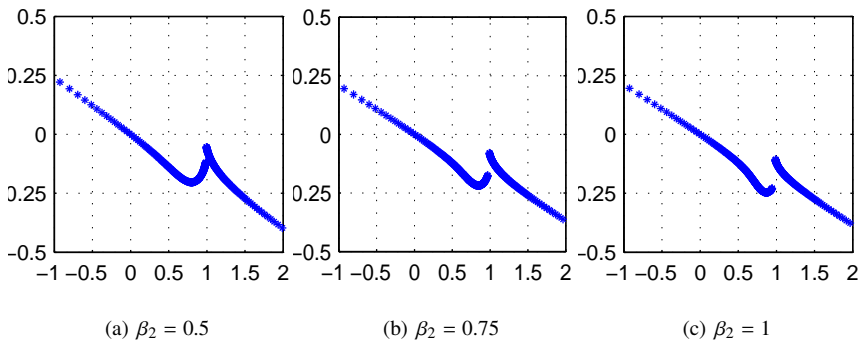


Figure 5.3: Non-zero part of $\sigma(B_{h,2h,(1,\beta_2)})$ in the one-dimensional problem for $k = 2000$ and for various values of β_2 satisfying $\kappa = 0.625$.

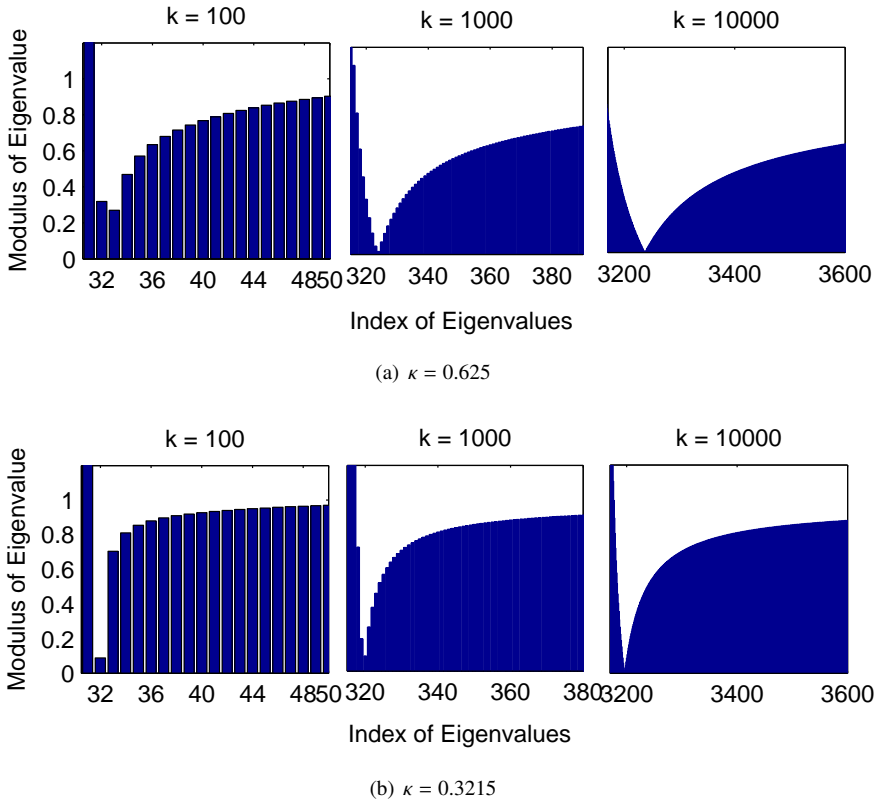


Figure 5.4: Magnitude of non-zero part of $\sigma(B_{h,2h,(1,1)})$ in the one-dimensional problem for $\kappa = 0.625$ and $\kappa = 0.3215$ and various values of k .

per wavelength is used (which corresponds to a larger value of κ). However Figure 5.4 also signify the fact that the ratio of small eigenvalues to the large ones remains constant. This can be noticed while considering spread of the x-axis for different wave numbers k in the same Figure.

5.3 ADEF1 Simplified Analysis

We have broadly analyzed the effect of the CSLP exclusively and along with deflation preconditioner. Now we make a simplification in the analysis in order to emphasize the exclusive influence of deflation preconditioner on the spectrum of the Helmholtz operator. The simplified analysis can also be seen as a complement of the analysis above and also of the analysis in [101]. We study here the spectrum of the deflated

preconditioned operator, i.e., the spectrum of the operator

$$\widehat{B}_{h,2h,(\beta_1,\beta_2)} = P_{h,2h}^T A_h. \quad (5.40)$$

It appears that $\widehat{B}_{h,2h,(\beta_1,\beta_2)}$ has the essential features of the spectrum of $B_{h,2h,(\beta_1,\beta_2)}$, specifically the eigenvalues near null space behave similar in both. The application of the CSLP preconditioner does not alter the spectrum in a significant manner. Vital effect of CSLP, while applied in combination with deflation preconditioner, can be concluded as it reduces the magnitude of the eigenvalues and it gathers the spectrum in a circle with small radius (≈ 1). However CSLP is not well suited to eliminate the near kernel nodes.

A Rigorous Fourier analysis yields that the eigenvalues of this operator are given by

$$\lambda^\ell \left(\widehat{B}_{h,H,(\beta_1,\beta_2)} \right) = - \frac{(c_\ell^2 + 1)\kappa^4 + (-4 c_\ell^2 - 4)\kappa^2 - 4(c_\ell^4 - 1)}{((c_\ell^2 + 1)\kappa^2 + 2(c_\ell^2 - 1))h^2}. \quad (5.41)$$

The denominator of this expression is the constant factor of the ℓ -th eigenvalue of the coarse grid operator A_{2h} defined by (5.24) also shown by Equation (5.12) in [101]. Since the Galerkin operator A_{2h} inherits the characteristic of yielding the near-null space modes from the Helmholtz operator, therefore eigenvalues $\lambda^\ell \left(\widehat{B}_{h,H,(\beta_1,\beta_2)} \right)$ can be expected to be large in magnitude for indexes ℓ corresponding to the those near-null space modes of A_{2h} .

In Figure (5.5) we plot the real part of the eigenvalues of $\widehat{B}_{h,2h,(\beta_1,\beta_2)}$ versus the index ℓ . In the top and bottom row of this figure we plotted the whole range of indices and those close to the near-kernel, respectively. This figure confirms that the scattering of the eigenvalues to the left and right of $(1, 0)$ in the complex plane is caused by the instability of the coarse grid solve.

It is important to clarify that the above expressions for eigenvalues do not include the “shift term” Q_h . Another important aspect which we want to investigate is that how smoothly adding Q_h shifts the “deflated spectrum”. Figure 5.6 demonstrates the exclusive effect of the shift term Q_h , which we have been discussed in the start of this chapter. Figures 5.6(a) and 5.6(b) present the spectrum of $\widehat{B}_{h,2h,(\beta_1,\beta_2)}$ and $P_{h,2h}^T A_h + Q_h A_h$. The difference in these two figures is that the later includes the shift term, thus one can notice that the deflated part of the spectrum (which is 0) is shifted to 1. The real part of the eigenvalues of operators $\widehat{B}_{h,2h,(\beta_1,\beta_2)}$ and $P_{h,2h}^T A_h + Q_h A_h$ has a large magnitude without CSLP hence one can not distinguish with the current axis. Including CSLP highlights this fact. In Figures 5.6(c) and 5.6(d) we plot the spectrum from Equation (5.39) and the same with addition of the shift term. In these two, one can observe the shifting of the deflated spectrum from 0 to 1.

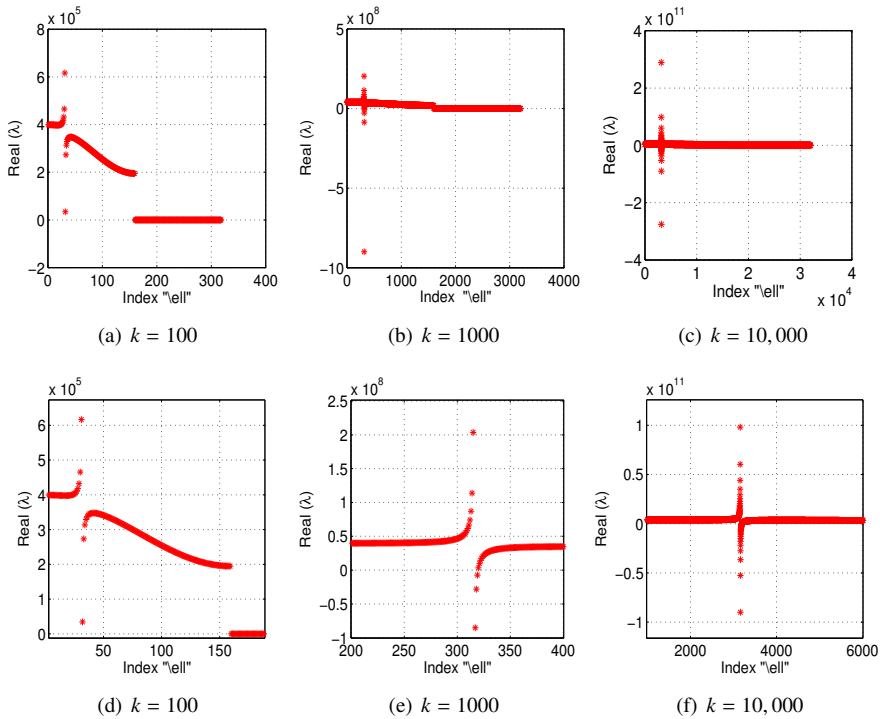


Figure 5.5: Real part of eigenvalues of the deflated unpreconditioned operator $\widehat{B}_{h,2h,(\beta_1,\beta_2)}$ versus the index ℓ for various values of the wavenumber. The top and bottom show all and a selection of indexes ℓ close to the near-null space, respectively.

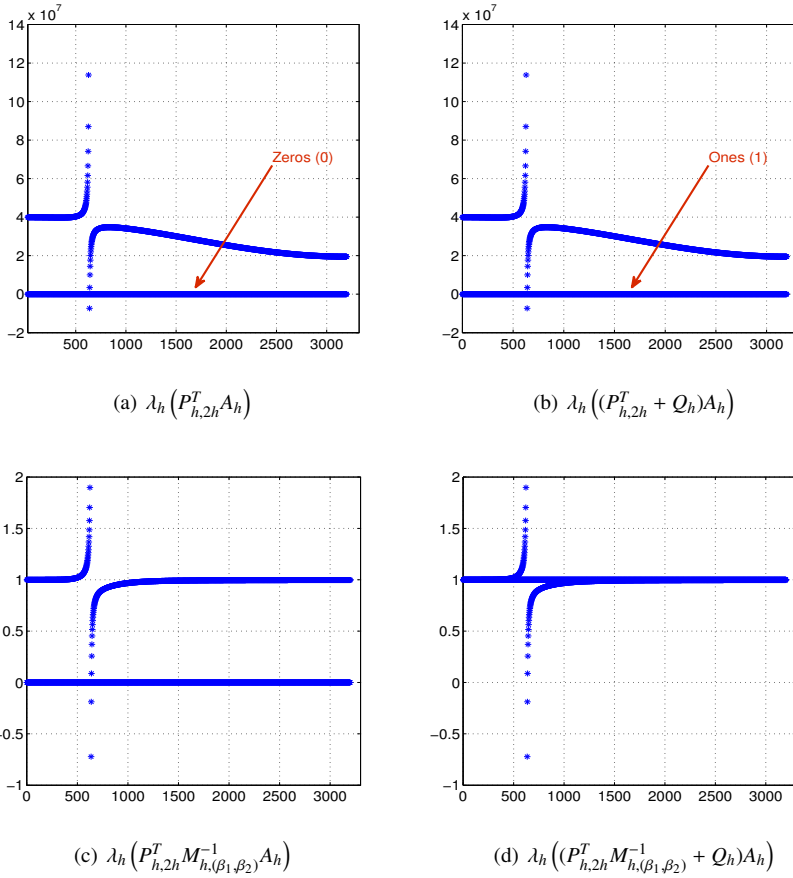


Figure 5.6: Real part of spectrum; ADEF1 preconditioner applied at \mathbf{A}_h . Wavenumber considered is $k = 10000$ with grid size is set such that there are 20 grid points per wavelength

5.4 Two-Dimensional Rigorous Two-Grid Analysis of ADEF1

In this section we extend the analysis of the previous section to two dimensional geometries. We therefore consider the Helmholtz equation (2.22) supplied with homogeneous Dirichlet boundary conditions discretized on $\Omega_h = (0, 1) \times (0, 1)$. As before we assume the preconditioner $M_{h,(\beta_1,\beta_2)}$ and the coarse grid operator A_{2h} to be inverted exactly. The second order finite difference discretization using the stencil (2.20) results after elimination of the boundary conditions in the linear system

$$A_h x_h = b_h$$

where $A_h \in \mathbb{C}^{(n-1)^2 \times (n-1)^2}$.

We denote by $\underline{x}, \underline{y} \in \mathbb{R}^{n-1}$ the vectors with components $x_i = ih$ and $y_i = ih$ and observe that the grid vectors

$$\phi_h^{\ell_1, \ell_2} = \sin(\ell_1 \pi \underline{x}) \sin(\ell_2 \pi \underline{y}) \text{ for } 1 \leq \ell_1, \ell_2 \leq n-1 \quad (5.42)$$

are eigenvectors of A_h corresponding to the eigenvalues

$$\lambda^{\ell_1, \ell_2}(A_h) = \frac{1}{h^2} (4 - 2c_{\ell_1} - 2c_{\ell_2} - \kappa^2), \quad (5.43)$$

where $c_{\ell_1} = \cos(\ell_1 \pi h)$ and $c_{\ell_2} = \cos(\ell_2 \pi h)$. The eigenvalues of the preconditioner $M_{h,(\beta_1,\beta_2)}$ and the preconditioned operator $S_{h,(\beta_1,\beta_2)} = M_{h,(\beta_1,\beta_2)}^{-1} A_h$ are therefore given by

$$\lambda^{\ell_1, \ell_2}(M_{h,(\beta_1,\beta_2)}) = \frac{1}{h^2} [4 - 2c_{\ell_1} - 2c_{\ell_2} - (\beta_1 - i\beta_2)\kappa^2], \quad (5.44)$$

and

$$\lambda^{\ell_1, \ell_2}(S_{h,(\beta_1,\beta_2)}) = \frac{4 - 2c_{\ell_1} - 2c_{\ell_2} - \kappa^2}{4 - 2c_{\ell_1} - 2c_{\ell_2} - \kappa^2(\beta_1 - i\beta_2)}, \quad (5.45)$$

respectively.

5.4.1 Basis Diagonalization

To block-diagonalize the multigrid deflation operator $P_{h,2h}^T$, we again reorder the basis of eigenvectors of A_h in a standard way in

$((\ell_1, \ell_2), (n - \ell_1, n - \ell_2), (\ell_1, n - \ell_2), (n - \ell_1, \ell_2))$ 4-tuples [113] to obtain the basis

$$\begin{aligned} V_h &= \{(\phi_h^{\ell_1, \ell_2}, \phi_h^{n-\ell_1, n-\ell_2}, \phi_h^{\ell_1, n-\ell_2}, \phi_h^{\ell_1, n-\ell_2}) \mid \ell_1, \ell_2 = 1, \dots, n/2 - 1\} \\ &\cup \{(\phi_h^{\ell_1, n/2}, \phi_h^{n-\ell_1, n/2}) \mid \ell_1 = 1, \dots, n/2 - 1\} \\ &\cup \{(\phi_h^{n/2, \ell_2}, \phi_h^{n/2, n-\ell_2}) \mid \ell_2 = 1, \dots, n/2 - 1\} \cup \{\phi_h^{n/2, n/2}\} \end{aligned} \quad (5.46)$$

In this basis, $P_{h,2h}^T$ can be written in a block diagonal form, i.e., we can write

$$P_{h,2h}^T = [(P_{h,2h}^{\ell_1, \ell_2})^T]_{1 \leq \ell_1, \ell_2 \leq n/2}, \quad (5.47)$$

where the individual blocks are given by

$$(P_{h,2h}^{\ell_1,\ell_2})^T = I - (I_{2h}^h)^{\ell_1,\ell_2} (A_{2h}^{\ell_1,\ell_2})^{-1} (I_h^{2h})^{\ell_1,\ell_2} A_h^{\ell_1,\ell_2}. \quad (5.48)$$

A standard computation results in the 4×1 blocks of the bilinear interpolation I_{2h}^h where for $1 \leq \ell_1, \ell_2 \leq n/2 - 1$

$$(I_{2h}^h)^{\ell_1,\ell_2} = \frac{1}{4} \begin{bmatrix} (1 + c_{\ell_1})(1 + c_{\ell_2}) \\ (1 - c_{\ell_1})(1 - c_{\ell_2}) \\ (1 + c_{\ell_1})(1 - c_{\ell_2}) \\ (1 - c_{\ell_1})(1 + c_{\ell_2}) \end{bmatrix} \quad (5.49)$$

and for other values of ℓ_1 and ℓ_2

$$(I_{2h}^h)^{\ell_1,\ell_2} = 0. \quad (5.50)$$

The Galerkin coarsening gives the 1×1 blocks where for $1 \leq \ell_1, \ell_2 \leq n/2 - 1$

$$A_{2h}^{\ell_1,\ell_2} = \frac{1}{h^2} \left[4 \left(1 - 2 c_{\ell_1} c_{\ell_2} + c_{\ell_1}^2 + c_{\ell_2}^2 + c_{\ell_1}^2 c_{\ell_2}^2 - c_{\ell_1}^3 c_{\ell_2} - c_{\ell_1} c_{\ell_2}^3 \right) - \kappa^2 \left(1 + c_{\ell_1}^2 + c_{\ell_2}^2 + c_{\ell_1}^2 c_{\ell_2}^2 \right) \right] \quad (5.51)$$

and for other values of ℓ_1 and ℓ_2

$$A_{2h}^{\ell_1,\ell_2} = 0. \quad (5.52)$$

The 4×4 blocks of the multigrid deflation operator $P_{h,2h}^T$ can subsequently be computed for $1 \leq \ell_1, \ell_2 \leq n/2 - 1$. For others values of ℓ_1 and ℓ_2 , these blocks are of size either 2×2 or 1×1 and equal to the identity.

The basis V_h (5.46) can therefore be used to block-diagonalize the deflated preconditioned operator, i.e., we can write

$$B_{h,2h,(\beta_1,\beta_2)} = [B_{h,2h,(\beta_1,\beta_2)}^{\ell_1,\ell_2}]_{1 \leq \ell_1,\ell_2 \leq n/2} \quad (5.53)$$

where for $1 \leq \ell_1, \ell_2 \leq n/2 - 1$, $B_{h,2h,(\beta_1,\beta_2)}^{\ell_1,\ell_2}$ is the 4×4 block

$$B_{h,2h,(\beta_1,\beta_2)}^{\ell_1,\ell_2} = (P_{h,2h}^{\ell_1,\ell_2})^T \text{diag} \begin{bmatrix} \lambda^{\ell_1,\ell_2} (S_{h,(\beta_1,\beta_2)}) \\ \lambda^{n-\ell_1,n-\ell_2} (S_{h,(\beta_1,\beta_2)}) \\ \lambda^{n-\ell_1,\ell_2} (S_{h,(\beta_1,\beta_2)}) \\ \lambda^{\ell_1,n-\ell_2} (S_{h,(\beta_1,\beta_2)}) \end{bmatrix}, \quad (5.54)$$

and where for other values of ℓ_1 and ℓ_2 , $B_{h,2h,(\beta_1,\beta_2)}^{\ell_1,\ell_2}$ is either a 2×2 block, e.g.,

$$B_{h,2h,(\beta_1,\beta_2)}^{\ell_1,n/2} = \text{diag} \begin{bmatrix} \lambda^{\ell_1,n/2} (S_{h,(\beta_1,\beta_2)}) \\ \lambda^{n-\ell_1,n/2} (S_{h,(\beta_1,\beta_2)}) \end{bmatrix} \quad (5.55)$$

or the 1×1 block

$$B_{h,2h,(\beta_1,\beta_2)}^{n/2,n/2} = \lambda^{n/2,n/2} (S_{h,(\beta_1,\beta_2)}) = \frac{4 - \kappa^2}{4 - \kappa^2(\beta_1 - i\beta_2)}. \quad (5.56)$$

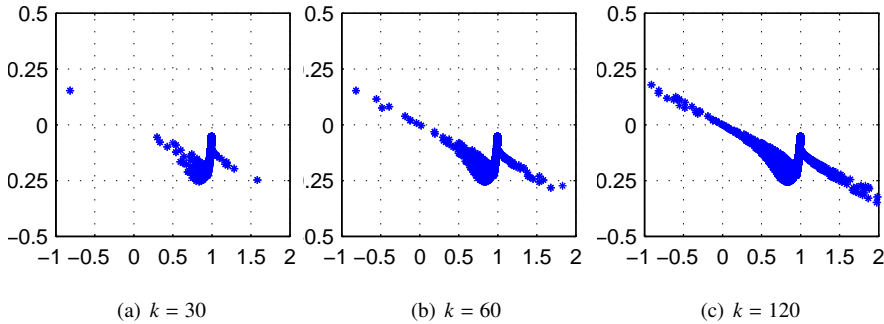


Figure 5.7: Non-zero part of $\sigma(B_{h,2h,(1,1)})$ for the two-dimensional problem for various values of k satisfying $\kappa = 0.625$.

The eigenvalues of $B_{h,2h,(\beta_1,\beta_2)}$ can therefore be computed numerically as the eigenvalues of separate blocks. In the remainder of this section we will present results of such computations that confirm properties of the solver revealed in the previous section. Figure (5.7) shows the non-zero part of the spectrum of $B_{h,2h,(\beta_1=1,\beta_2=1)}$ for $k = 30$, $k = 60$ and $k = 120$ using 10 grid points per wavelength (and thus $\kappa = 0.625$). As in the one-dimensional problem, the spectrum is clustered around one and the number of small eigenvalues increases as the wavenumber increases. The number of unresolved near null-space eigen-modes grows however substantially faster than in the one-dimensional case. These unresolved modes hamper the solver from being scalable.

Figure (5.8) shows the magnitude of the five eigenvalues of $\sigma(B_{h,H,(1,1)})$, smallest in size, versus the wave number for $\kappa = 0.625$ (left) and $\kappa = 0.3215$ (right) on a logarithmic scale. This figures serves to further illustrate the observation made earlier for Figure (5.7). From this figure, one can deduce that the magnitude of the smallest eigenvalues (5, in case of this figure) reduces when the wave number is increased and eigenvalues come closer to origin as the wave number increases. This effect is more pronounced in case when a small number of grid points per wavelength is used.

Figure (5.9) shows a bar plot of part of the eigenvalues of $B_{h,2h,(\beta_1=1,\beta_2=1)}$ for $\kappa = 0.625$ and $\kappa = 0.3215$ and for various values of k . These bar plots are aimed to highlight the frequency of eigenvalues close to zero in modulus. The frequency of eigenvalues close to 1 is higher in Figure 5.9(b) than that in Figure 5.9(a), as the mesh size is halved in the earlier figure. The y-axis are kept the same in both Figures 5.9(a) and Figure 5.9(b), in order to compare the frequency of near-zero part of the eigenvalues while considering different number of grid points. Therefore the *bar* corresponding t eigenvalues 1 or close to 1 are trimmed. These figures confirm earlier findings that the number of small eigenvalues grows with k and that this effect is more pronounced if a smaller number of grid points per wavelength is used.

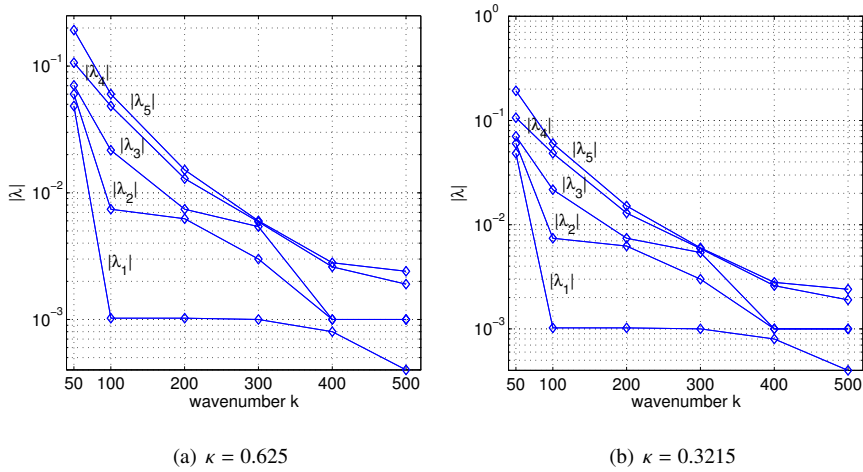


Figure 5.8: Magnitude of the five eigenvalues smallest in size of $\sigma(B_{h,H_s(1,1)})$ versus the wave number in the two-dimensional problem for $\kappa = 0.625$ (left) and $\kappa = 0.3215$ (right).

5.5 One-Dimensional Rigorous Two-Grid Analysis of TLKM

In this section, we advance to the TLKM deflation preconditioner. TLKM deflation is classified into the ideal (some times referred as theoretical) and the practical versions. The two versions are significantly different in the choice of the coarse grid operator. Details can be found in Chapter 4. First of all, we will perform a two-grid Fourier analysis of the ideal version of the TLKM deflation preconditioner. An aspect of the analysis of the ideal variant can also be found in [92]. In the subsequent section, we will discuss the analysis of the corresponding practical version.

Helmholtz and CSLP: Alike in case of ADEF1, we make use of Dirichlet’s boundary conditions, for convenience and as they can be seen as the worst case since they lack damping. We consider a one dimensional Helmholtz problem on the unit domain $\Omega_h = (0, 1)$. The stencil of the Helmholtz is given by

$$[A_h] = \frac{1}{h^2} \begin{bmatrix} -1 & 2 - \kappa^2 & -1 \end{bmatrix} \text{ where } \kappa = kh. \quad (5.57)$$

From previous sections, it is straightforward to write the eigenvectors and eigenvalues of the discrete Helmholtz equation as

$$\phi_h^\ell = \sin(\ell\pi x) \text{ for } 1 \leq \ell \leq n - 1 \quad (5.58)$$

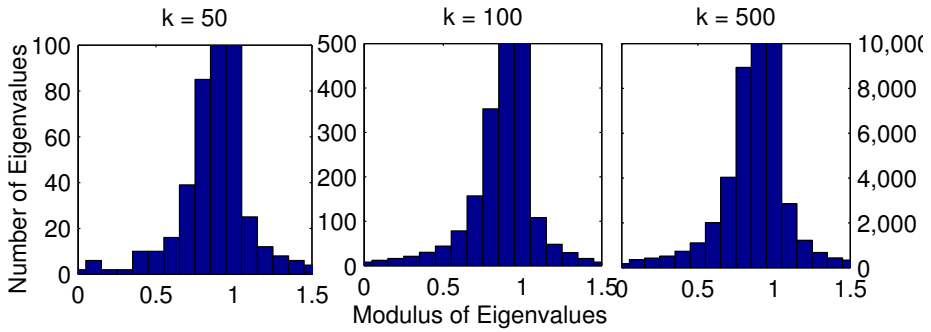
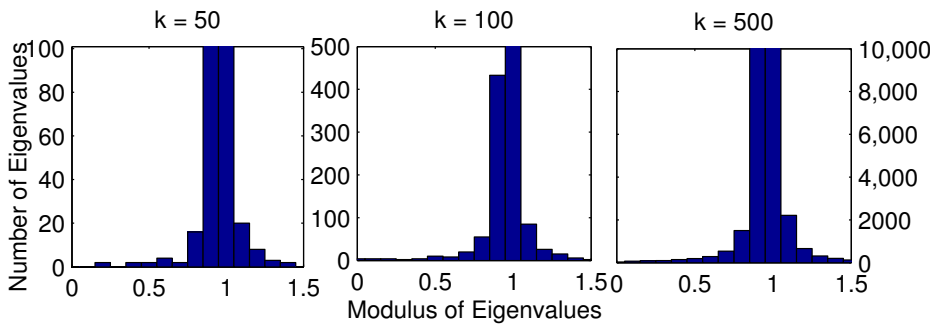
(a) $\kappa = 0.625$ (b) $\kappa = 0.3215$

Figure 5.9: Magnitude of the non-zero part of $\sigma(B_{h,2h,(1,1)})$ in the two-dimensional problem for $\kappa = 0.625$ and $\kappa = 0.3215$ for various values of k .

and

$$\lambda^\ell(A_h) = \frac{1}{h^2}(2 - 2c_\ell - \kappa^2), \quad (5.59)$$

where $c_\ell = \cos(\ell\pi h)$, as defined in previous sections.

As we know, the Helmholtz operator A_h and CSLP $M_{h,(\beta_1,\beta_2)}$ carry the same eigenvectors, the eigenvalue of $M_{h,(\beta_1,\beta_2)}$ are

$$\lambda^\ell(M_{h,(\beta_1,\beta_2)}) = \frac{1}{h^2}\left(2 - 2c_\ell - \kappa^2(\beta_1 - i\beta_2)\right). \quad (5.60)$$

We know that the TLKM preconditioner is applied to $\hat{A}_h = M_{h,(\beta_1,\beta_2)}^{-1}A_h$ instead of A_h . Thus we present the spectrum of \hat{A}_h as

$$\lambda^\ell(\hat{A}_h) = \frac{2 - 2c_\ell - \kappa^2}{2 - 2c_\ell - \kappa^2(\beta_1 - i\beta_2)}. \quad (5.61)$$

5.5.1 Basis Diagonalization

By making use of basis V_h defined in Equation 5.12, which is obtained by reordering the eigenvectors of A_h in the standard way $(\ell, n - \ell)$ pairs [113], the eigenvalues of \hat{P}_h can be transformed into block diagonal formulated as

$$\hat{P}_h = [\hat{P}_h^\ell]_{1 \leq n/2} \quad (5.62)$$

where each 2×2 block \hat{P}_h^ℓ is algebraic combination of blocks (of all operators forming \hat{P}_h);

$$\begin{aligned} \hat{P}_h^\ell &= I - \hat{A}_h^\ell \hat{Q}_h^\ell + \hat{Q}_h^\ell \\ &= I - \hat{A}_h^\ell \left(I_{2h}^h (\hat{A}_{2h}^\ell)^{-1} I_h^{2h} \right) + I_{2h}^h (\hat{A}_{2h}^\ell)^{-1} I_h^{2h} \hat{Q}_h^\ell \end{aligned} \quad (5.63)$$

where \hat{A}_{2h}^ℓ is 1×1 block and has been approximated as

$$\hat{A}_{2h}^\ell = (M_{2h}^{-1} A_{2h} B_h)^\ell \quad (5.64)$$

where $B_h = I_{2h}^h I_h^{2h}$ and thus the above block-equation can be rewritten as

$$\hat{P}_h^\ell = I - (M_h^{-1})^\ell A_h^\ell \left(M_{2h}^{-1} A_{2h} B_h \right)^\ell \left((I_{2h}^h)^\ell (I_h^{2h})^\ell \right) + \hat{A}_h^\ell \left(M_{2h}^{-1} A_{2h} B_h \right)^\ell \left((I_{2h}^h)^\ell (I_h^{2h})^\ell \right) \quad (5.65)$$

Diagonalization of A_h , $M_{h,(\beta_1-i\beta_2)}$ and \hat{A}_h The diagonalization of A_h , $M_{h,(\beta_1,\beta_2)}$ and \hat{A}_h with basis vectors given in 5.12 immediately results in:

$$A_h \xrightarrow{\text{Diagonalization}} [A_h^\ell]_{1 \leq \ell \leq n/2}, \quad (5.66)$$

$$M_h \xrightarrow{\text{Diagonalization}} [M_h^\ell]_{1 \leq \ell \leq n/2}, \quad (5.67)$$

and

$$\hat{A}_h \xrightarrow{\text{Diagonalization}} \left[\hat{A}_h^\ell \right]_{1 \leq \ell \leq n/2}, \quad (5.68)$$

where for $1 \leq \ell \leq n/2 - 1$, the respective blocks are

$$A_h^\ell = \frac{1}{h^2} \begin{bmatrix} (2 - 2c_\ell - \kappa^2) & 0 \\ 0 & (2 + 2c_\ell - \kappa^2) \end{bmatrix}, \quad (5.69)$$

$$M_{h,(\beta_1, \beta_2)}^\ell = \frac{1}{h^2} \begin{bmatrix} \frac{1}{h^2}(2 - 2c_\ell - (\beta_1 - \iota\beta_2)\kappa^2) & 0 \\ 0 & \frac{1}{h^2}(2 + 2c_\ell - (\beta_1 - \iota\beta_2)\kappa^2) \end{bmatrix}, \quad (5.70)$$

and

$$\hat{A}_h^\ell = \begin{bmatrix} \frac{2-2c_\ell-\kappa^2}{2-2c_\ell-(\beta_1-\iota\beta_2)\kappa^2} & 0 \\ 0 & \frac{2+2c_\ell-\kappa^2}{2+2c_\ell-(\beta_1-\iota\beta_2)\kappa^2} \end{bmatrix}. \quad (5.71)$$

For value of $\ell = n/2$, the 1×1 blocks of A_h , $M_{h,(\beta_1, \beta_2)}$ and \hat{A}_h are $A_h^{n/2} = \frac{2}{h^2} - k^2$, $M_h^{n/2} = \frac{2}{h^2} - (\beta_1 - \iota\beta_2)k^2$ and $\hat{A}_h^{n/2} = \frac{2 - k^2}{2 - (\beta_1 - \iota\beta_2)k^2}$.

$$A_{2h}^\ell = (I_h^{2h})^\ell A_h^\ell (I_{2h}^h)^\ell = \frac{2(1 - c_\ell^2) - \kappa^2(1 + c_\ell^2)}{2h^2}, \quad (5.72)$$

Diagonalization of grid transfer operators and coarse grid operator For restriction operator I_h^{2h} we will use the full weighting operator with stencil

$$[I_h^{2h}] = \begin{bmatrix} \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \end{bmatrix}, \quad (5.73)$$

and as prolongation operator, we use the linear interpolation I_{2h}^h . Recalling from previous section that by diagonalization, the restriction operator with this stencil can be transformed in into 1×2 blocks

$$(I_h^{2h})^\ell = \begin{bmatrix} \frac{1}{2}(1 + c_\ell) & -\frac{1}{2}(1 - c_\ell) \end{bmatrix}, \quad (5.74)$$

for $1 \leq \ell \leq n/2 - 1$ and $(I_h^{2h})^{n/2} = 0$. Since the inter-grid transfer operators are related by $I_{2h}^h = (I_h^{2h})^T$, therefore the linear interpolation operator $I_h^{2h} \in \mathbb{R}^{n \times (\frac{n}{2}-1)}$ would be read as, for $1 \leq \ell \leq n/2 - 1$

$$(I_{2h}^h)^\ell = \frac{1}{2} \begin{bmatrix} (1 + c_\ell) \\ -(1 - c_\ell) \end{bmatrix}, \quad (5.75)$$

while evaluating $(I_{2h}^h)^\ell$ at $\ell = n/2$, we have

$$(I_{2h}^h)^{n/2} = 0. \quad (5.76)$$

The 1×1 diagonal blocks of the Galerkin coarse grid operator \hat{A}_{2h} can be shown to be equal to

$$\begin{aligned}\hat{A}_{2h}^\ell &= (I_h^{2h})^\ell \hat{A}_h^\ell (I_{2h}^h)^\ell \\ &= (I_h^{2h})^\ell (M_h^{-1})^\ell A_h^\ell (I_{2h}^h)^\ell\end{aligned}\quad (5.77)$$

which, after few derivations, will be read as

$$\hat{A}_{2h}^\ell = \left[\frac{(2-2c_\ell - \kappa^2)(1/2+1/2c_\ell)^2}{2-2c_\ell - \kappa^2\beta_1 + i\kappa^2\beta_2} + \frac{(2+2c_\ell - \kappa^2)(-1/2+1/2c_\ell)^2}{2+2c_\ell - \kappa^2\beta_1 + i\kappa^2\beta_2} \right] \quad (5.78)$$

for $1 \leq \ell \leq n/2 - 1$ and \hat{A}_{2h}^ℓ valued at $\ell = n/2$ is given by

$$(\hat{A}_{2h})^{n/2} = \frac{2 - \kappa^2}{2(2 - (1 - \iota)\kappa^2)}$$

5.5.2 Closed-form Expression for Eigenvalues

The TLKM ideal variant turns up a deflation operator which is still a projection. Therefore deflated preconditioned operator i.e. two-grid operator $B_{h,2h,(\beta_1,\beta_2)}$ developed by using this variant of deflation has a multiple of zero eigenvalues. The above rigorous Fourier analysis yields that the zero eigenvalue has multiplicity $n/2 - 1$. The rest of the eigenvalues can be represented in the formula

$$\lambda^\ell(B_{h,2h,(\beta_1,\beta_2)}) = \frac{D_\ell(c_\ell, \kappa^2, \beta_1, \beta_2)}{F_\ell(c_\ell, \kappa^2, \beta_1, \beta_2) + \iota G_\ell(c_\ell, \kappa^2, \beta_1, \beta_2)} \quad \text{for } \ell = 1, \dots, n/2 - 1, \quad (5.79)$$

where D_ℓ , F_ℓ and G_ℓ are second order polynomials in κ^2 given by

$$D_\ell(c_\ell, \kappa^2, \beta_1, \beta_2) = (1 + c_\ell^2)\kappa^4 - 4(1 + c_\ell^2)\kappa^2 + 4(1 - c_\ell^4) \quad (5.80)$$

$$F_\ell(c_\ell, \kappa^2, \beta_1, \beta_2) = -\beta_1(1 + c_\ell^2)\kappa^4 + 2(1 + \beta_1 - \beta_1 c_\ell^2 + 3c_\ell^2)\kappa^2 - 4(1 - c_\ell^4)$$

$$G_\ell(c_\ell, \kappa^2, \beta_1, \beta_2) = \beta_2(1 - c_\ell^2)\kappa^4 - 2\beta_2(1 - c_\ell^2)\kappa^4,$$

and eigenvalue for $\ell = n/2$ is given by

$$\lambda^{n/2}(B_{h,2h,(\beta_1,\beta_2)}) = \frac{2 - \kappa^2}{2 - (\beta_1 - \iota\beta_2)\kappa^2}. \quad (5.81)$$

In Figure 5.10, we plot the spectrum from Formula 5.79 for various values of the wavenumber, i.e. $k = 100$, $k = 1000$ and $k = 10000$. The value of κ is chosen as $\kappa = 0.3125$ representing one wavelength of the wave over 20 grid points. This ideal variant of TLKM is remarkably good for low wavenumbers, as the spectrum of the two grid operator for wave number $k = 100$ and $k = 1000$ can be seen well clustered around 1 in the complex plane. But for large wavenumbers, it is not that efficient and seems to fail to justify the implementation cost. Figure 5.10(c) specifies this fact where scattering of eigenvalues from the cluster around 1 can be noticed, when k is

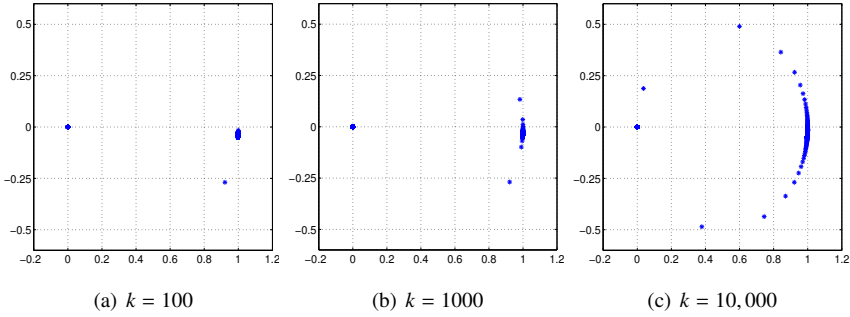


Figure 5.10: Spectrum of the TLKM ideal variant with CSLP for various wave numbers resolved over a grid with 20 gridpoint per wavelength.

increased to $k = 10000$. It is illustrative to look into the real and imaginary part of these eigenvalues denoted by $\text{Re}[\lambda^\ell(B_{h,2h,(\beta_1,\beta_2)})]$ and $\text{Im}[\lambda^\ell(B_{h,2h,(\beta_1,\beta_2)})]$, respectively. We have that for $\ell = 1, \dots, n/2 - 1$

$$\text{Re}[\lambda^\ell(B_{h,2h,(\beta_1,\beta_2)})] = \frac{p(c_\ell, \kappa^2, \beta_1, \beta_2)}{q(c_\ell, \kappa^2, \beta_1, \beta_2)} \quad (5.82)$$

$$\text{Im}[\lambda^\ell(B_{h,2h,(\beta_1,\beta_2)})] = \frac{r(c_\ell, \kappa^2, \beta_1, \beta_2)}{q(c_\ell, \kappa^2, \beta_1, \beta_2)} \quad (5.83)$$

where $p(c_\ell, \kappa^2, \beta_1, \beta_2)$, $q(c_\ell, \kappa^2, \beta_1, \beta_2)$ and $s(c_\ell, \kappa^2, \beta_1, \beta_2)$ are fourth order polynomials in κ^2 given by

$$\begin{aligned}
p(c_\ell, \kappa) &= (2\beta_1 c_\ell^2 + c_\ell^4 \beta_1 + \beta_1) \kappa^8 \\
&\quad + (-8\beta_1 c_\ell^2 - 2 c_\ell^4 \beta_1 - 6\beta_1 - 6 c_\ell^4 - 8 c_\ell^2 - 2) \kappa^6 \\
&\quad + (12\beta_1 + 4\beta_1 c_\ell^2 - 12 c_\ell^4 \beta_1 - 4 c_\ell^6 \beta_1 + 20 c_\ell^4 + 36 c_\ell^2 + 12 - 4 c_\ell^6) \kappa^4 \\
&\quad + (-24 - 40 c_\ell^2 + 40 c_\ell^6 - 8\beta_1 - 8 c_\ell^6 \beta_1 + 24 c_\ell^4 + 8 c_\ell^4 \beta_1 + 8\beta_1 c_\ell^2) \kappa^2 \\
&\quad + 16 - 32 c_\ell^4 + 16 c_\ell^8 \\
q(c_\ell, \kappa) &= (\beta_2^2 + \beta_1^2 + 2\beta_1^2 c_\ell^2 + 2 c_\ell^2 \beta_2^2 + c_\ell^4 \beta_1^2 + c_\ell^4 \beta_2^2) \kappa^8 \\
&\quad + (4 c_\ell^4 \beta_2^2 - 4\beta_1 - 4\beta_2^2 - 12 c_\ell^4 \beta_1 + 4 c_\ell^4 \beta_1^2 - 16\beta_1 c_\ell^2 - 4\beta_1^2) \kappa^6 \\
&\quad + (4 c_\ell^4 \beta_2^2 + 36 c_\ell^4 + 24 c_\ell^2 - 8 c_\ell^2 \beta_2^2 + 4\beta_2^2 + 16\beta_1 + 4 + 4\beta_1^2 \dots \\
&\quad + 4 c_\ell^4 \beta_1^2 - 8 c_\ell^6 \beta_1 - 32 c_\ell^4 \beta_1 - 8\beta_1^2 c_\ell^2 + 24\beta_1 c_\ell^2) \kappa^4 \\
&\quad + (-16 + 16\beta_1 c_\ell^2 - 48 c_\ell^2 - 16\beta_1 + 48 c_\ell^6 + 16 c_\ell^4 \beta_1 + 16 c_\ell^4 - 16 c_\ell^6 \beta_1) \kappa^2 \\
&\quad + 16 - 32 c_\ell^4 + 16 c_\ell^8 \\
r(c_\ell, \kappa) &= (2 c_\ell^2 \beta_2 + \beta_2 c_\ell^4 + \beta_2) \kappa^8 + (-6\beta_2 - 8 c_\ell^2 \beta_2 - 2\beta_2 c_\ell^4) \kappa^6 \\
&\quad + (12\beta_2 + 4 c_\ell^2 \beta_2 - 4\beta_2 c_\ell^6 - 12\beta_2 c_\ell^4) \kappa^4 \\
&\quad + (-8\beta_2 + 8 c_\ell^2 \beta_2 + 8\beta_2 c_\ell^4 - 8\beta_2 c_\ell^6) \kappa^2
\end{aligned}$$

In addition to the illustration of the spectrum in the complex plane, we take an alternative approach in Figure 5.11, where we plot the real part of the non-zero eigenvalues i.e. $\text{Re}[\lambda^\ell(B_{h,2h,(\beta_1,\beta_2)})]$ versus the index ℓ for three values of the wavenumber k , namely $k = 100$, $k = 1000$ and $k = 10,000$. The formulae of $\text{Re}[\lambda^\ell(B_{h,2h,(\beta_1,\beta_2)})]$ as well as $\text{Im}[\lambda^\ell(B_{h,2h,(\beta_1,\beta_2)})]$ are given in Equations (5.82) and (5.83) respectively. These figures shows that the real part of the eigenvalues is close to 1 for $k = 100$ and for $k = 1000$. However, for $k = 10,000$, the near-kernel eigenvalues tend to shift towards the origin. We therefore expect the TLKM ideal variant ceases to be scalable for such high wavenumbers despite its high computational cost. Figure 5.10 clearly demonstrate this difference, where increasing the wave number k from 1000 to 10000 shows severance of eigenvalues from the cluster. To gain insights into this distribution of non-zero eigenvalues with respect to increase in wavenumber, we consider a particular case where the parameters are specified as $\kappa = 0.625$, $\beta_1 = 1$ and $\beta_2 = 1$. The expressions for $\text{Re}[\lambda^\ell(B_{h,2h,(\beta_1,\beta_2)})]$ and $\text{Im}[\lambda^\ell(B_{h,2h,(\beta_1,\beta_2)})]$ for $\ell = 1, \dots, n/2 - 1$ then reduce to the following rational expressions in c_ℓ^2

$$\text{Re}[\lambda^\ell(B_{h,2h,(1,1)})] = \frac{16 c_\ell^8 + 11.2792 c_\ell^6 - 18.7327 c_\ell^4 - 7.3036 c_\ell^2 + 6.7086}{16 c_\ell^8 + 11.2793 c_\ell^6 - 17.8608 c_\ell^4 - 8.4778 c_\ell^2 + 7.1038} \quad (5.84)$$

$$\text{Im}[\lambda^\ell(B_{h,2h,(1,1)})] = \frac{-3.7354 c_\ell^6 + 1.198 c_\ell^4 + 3.3051 c_\ell^2 - 1.6283}{16 c_\ell^8 + 11.2793 c_\ell^6 - 17.8608 c_\ell^4 - 8.4778 c_\ell^2 + 7.1038} \quad (5.85)$$

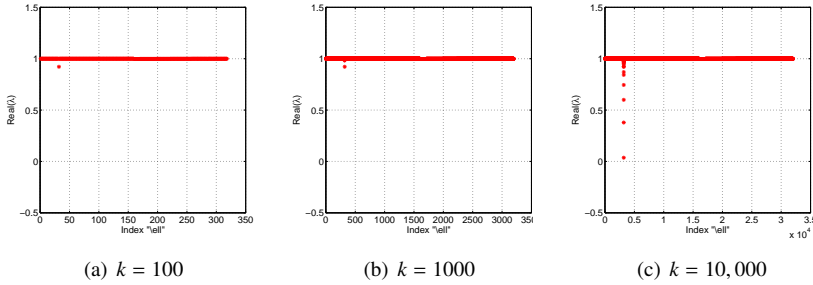


Figure 5.11: Real part of the non-zero eigenvalues of $B_{h,2h,(\beta_1,\beta_2)}$ versus the index ℓ for TLKM ideal variant.

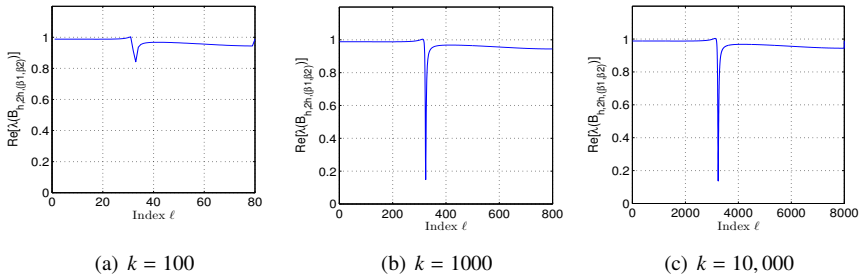


Figure 5.12: Real part of the non-zero eigenvalues of $B_{h,2h,(\beta_1,\beta_2)}$ versus the index ℓ for TLKM ideal variant.

Since $c_\ell = \cos(\ell\pi h)$, therefore formula $\text{Re}[\lambda^\ell(B_{h,2h,(1,1)})]$ can be seen as polynomial in index ℓ , say $f(\ell)$. The numerator and denominator of $\text{Re}[\lambda^\ell(B_{h,2h,(\beta_1,\beta_2)})]$ slightly differ in the low order terms of c_ℓ only. The zeroth order is profoundly different in numerator and denominator. Thus the function $f(\ell)$ can be seen as an approximation of constant function with range 1 for almost every index ℓ , except the certain indexes corresponding to which (real part of) eigenvalues tends to be near zero. Such behavior is confirmed by Figure 5.13, where the real part of the eigenvalues from Formula 5.84 are plotted against index ℓ . We also plot the \log_{10} of the real part given in Equation 5.84 versus the index $\ell = 1, \dots, n/2 - 1$ values of wave number $k = 100$, $k = 1000$ and $k = 10000$ in solid line. The dashed line represents the real part of non-zero eigenvalues of the CSLP preconditioner operator $\text{Re}[\lambda^\ell(M_{h(\beta_1,\beta_2)}^{-1}A_h)]$. Figure 5.13 thus highlights the action of the deflation preconditioner on the spectrum of CSLP preconditioned operator. The near-null space modes has sustained the deflation preconditioner and occur in the same indexes. The deflation reduces the adverse effect of the near-null space modes.

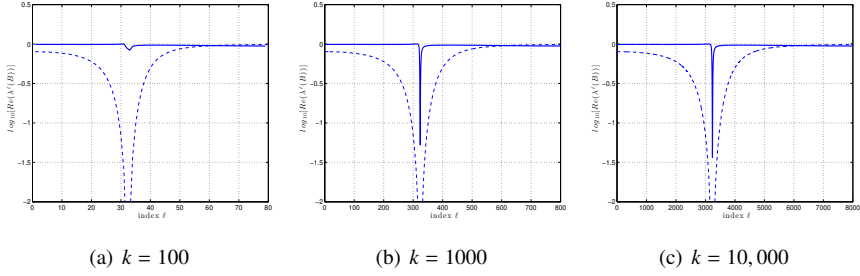


Figure 5.13: Real part of the non-zero eigenvalues of $B_{h,2h,(\beta_1,\beta_2)}$ versus the index ℓ for TLKM ideal variant.

5.6 TLKM Simplified Analysis

In this sections, simplification has been designated for the analysis of the TLKM ideal variant. We derive the formulae for eigenvalues without incorporating the CSLP. The CSLP preconditioner is excluded only on the fine level, but on coarser levels it is included as the coarser operator is part of the deflation preconditioner. The TLKM deflation preconditioner makes use of the CSLP preconditioned operator at fine as well as coarse level as usual. We will apply only the deflation preconditioner TLKM ideal variant to the Helmholtz operator instead of the CSLP preconditioned Helmholtz operator. This simplification of analysis will show the exclusive influence of the TLKM ideal preconditioner on the Helmholtz operator. Recall that the TLKM deflation preconditioner ideal variant is given by

$$\hat{P}_h^\ell = I - \hat{A}_h \hat{Q}_h, \quad (5.86)$$

where $\hat{A}_h = M_h^{-1} A_h$ and $\hat{Q}_h = I_h^{2h} \hat{A}_{2h} I_h^h$ are the same as earlier defined. As of now, we are interested in the spectrum of the operator

$$\widehat{B}_{h,2h,(\beta_1,\beta_2)} = \hat{P}_h^\ell A_h. \quad (5.87)$$

Adapting the analysis presented in earlier sections and some standard derivations will give the spectral formula for $\widehat{B}_{h,2h,(\beta_1,\beta_2)}$ which is

$$\lambda(\widehat{B}_{h,2h,(\beta_1,\beta_2)}) = -\frac{a(c_\ell, \kappa) + ib(c_\ell, \kappa)}{c(c_\ell, \kappa) + id(c_\ell, \kappa)} \quad (5.88)$$

where

$$\begin{aligned}
a(c_\ell, \kappa) &= (\beta_1 + \beta_1 c_\ell^2) \kappa^6 + (-2 - 6 c_\ell^2 - 4\beta_1 - 4\beta_1 c_\ell^2) \kappa^4 \\
&\quad + (8 + 24 c_\ell^2 + 4\beta_1 - 4\beta_1 c_\ell^4) \kappa^2 - (8 + 16 c_\ell^2 - 24 c_\ell^4) \\
b(c_\ell, \kappa) &= (-c_\ell^2 \beta_2 - \beta_2) \kappa^6 + (4\beta_2 + 4 c_\ell^2 \beta_2) \kappa^4 + (4 c_\ell^4 \beta_2 - 4\beta_2) \kappa^2 \\
c(c_\ell, \kappa) &= (-h^2 \beta_1 - h^2 \beta_1 c_\ell^2) \kappa^4 + (2h^2 + 2h^2 \beta_1 + 6h^2 c_\ell^2 - 2h^2 \beta_1 c_\ell^2) \kappa^2 - (4h^2 - 4h^2 c_\ell^4) \\
d(c_\ell, \kappa) &= (h^2 \beta_2 + h^2 c_\ell^2 \beta_2) \kappa^4 + (-2h^2 \beta_2 + 2h^2 c_\ell^2 \beta_2) \kappa^2.
\end{aligned}$$

Note that because of using \hat{A}_h (instead of A_h) for constructing \hat{P}_h^ℓ , we end up with the expression containing complex values. Also the coarse grid operator \hat{A}_{2h} carries out complex valued terms.

Further we like to explore the shifting process, as we did in case of analysis of the ADEF1 preconditioner. For this purpose, we look into the real part of the spectrum of $\widehat{B}_{h,2h,(\beta_1,\beta_2)}$, which is given in Equation 5.88 and the real part of the spectrum of $\widehat{B}_{h,2h,(\beta_1,\beta_2)} + \hat{Q}_h A_h$. In Figure 5.14(a), real part of the spectrum of $\widehat{B}_{h,2h,(1,1)}$ is plotted. The same is compared with the spectrum of $\widehat{B}_{h,2h,(1,1)} + \hat{Q}_h A_h$ plotted in Figure 5.14(b) where the deflated spectrum is shifted to 1 by applying the shift-term \hat{Q}_h . These figures indicate that shift term \hat{Q}_h does not perform a clear shifting of the zero eigenvalues (deflated eigenvalues) alike the case when we incorporate CSLP where shifting of the deflated eigenvalues is carried on without causing noise in the rest of the eigenvalues. This is made apparent in Figures 5.14(c) and 5.14(d), where the spectrum from Formula 5.82 is plotted. In these two figures, one can observe that zero eigenvalues are distinctly shifted to the maximum eigenvalue i.e. 1.

5.7 TLKM Practical Variant

In this section, we discuss the practical variant of TLKM preconditioner. If we recall the coarse grid operator \hat{A}_{2h} used to construct the TLKM operator earlier, it is

$$\begin{aligned}
\hat{A}_{2h}^\ell &= (I_h^{2h})^\ell \hat{A}_h^\ell (I_{2h}^h)^\ell \\
&= (I_h^{2h})^\ell (M_h^{-1})^\ell A_h^\ell (I_{2h}^h)^\ell
\end{aligned} \tag{5.89}$$

In Chapter 4, while exploring implementation possibilities, we discussed the approximation for the coarse grid operator \hat{A}_{2h} in Equation 4.36 which can be denoted by

$$\hat{A}_{2h} = \Theta_h \tilde{A}_{2h}, \tag{5.90}$$

where $\tilde{A}_{2h} = M_{2h}^{-1} A_{2h}$ and Θ_h is an approximation matrix. Now we know that A_{2h} and M_{2h} are simply the Galerkin operators therefore, eigenvalues are given below for certain indexes ℓ

$$A_{2h}^\ell = (I_h^{2h})^\ell \hat{A}_h^\ell (I_{2h}^h)^\ell = \frac{2(1 - c_\ell^2) - \kappa^2(1 + c_\ell^2)}{2h^2}, \tag{5.91}$$

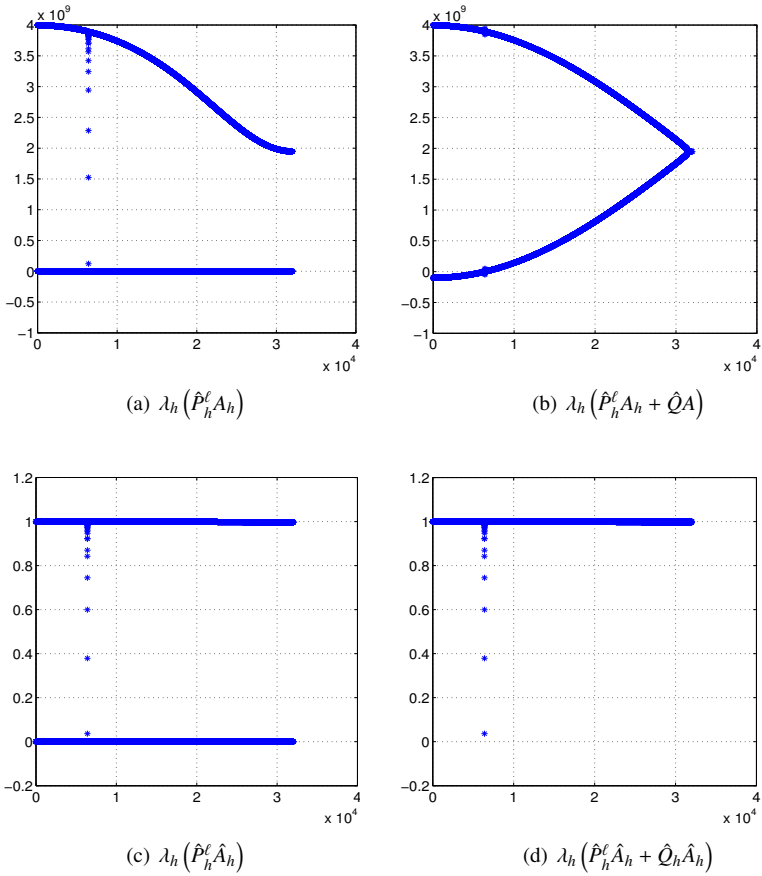


Figure 5.14: Real part of the spectrum; The TLKM ideal preconditioner applied to A_h . Wavenumber considered is $k = 10000$ with grid size is set such that there are 20 grid points per wavelength

and

$$M_{2h}^\ell = (I_h^{2h})^\ell M_h^\ell (I_h^h)^{\ell} = \frac{2(1 - c_\ell^2) - (\beta_1 - \iota\beta_2)\kappa^2(1 + c_\ell^2)}{2h^2}, \quad (5.92)$$

for $1 \leq \ell \leq n/2 - 1$ and $(A_{2h})^{n/2} = \frac{2-\kappa^2}{2h^2}$. Subsequently 1×1 block for $M_{2h}^{-1}A_{2h}$ will be

$$(M_{2h}^{-1}A_{2h})^\ell = \frac{2(1 - c_\ell^2) - \kappa^2(1 + c_\ell^2)}{2(1 - c_\ell^2) - (\beta_1 - \iota\beta_2)\kappa^2(1 + c_\ell^2)}, \quad (5.93)$$

for $1 \leq \ell \leq n/2 - 1$ and $(M_{2h}^{-1}A_{2h})^{n/2} = \frac{2 - \kappa^2}{2 - \kappa^2(\beta_1 - \iota\beta_2)}$. Also the 1×1 block for approximation term $\Theta_h = I_h^{2h}I_h^h$ can be simplified as

$$\Theta_h^\ell = (I_h^{2h}I_h^h)^\ell = (1 + c_\ell^2), \quad (5.94)$$

for $1 \leq \ell \leq n/2$. The algebraic combination of basis-diagonalized blocks of each of the corresponding operator given in Equation 5.65 can be written as

$$\hat{P}_h^\ell = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} \frac{2-2c_\ell-\kappa^2}{2-2c_\ell-(\beta_1-\iota\beta_2)\kappa^2} & 0 \\ 0 & \frac{2+2c_\ell-\kappa^2}{2+2c_\ell-(\beta_1-\iota\beta_2)\kappa^2} \end{bmatrix} \begin{bmatrix} \frac{2(1-c_\ell^2)-\kappa^2(1+c_\ell^2)}{2(1-c_\ell^2)-(\beta_1-\iota\beta_2)\kappa^2(1+c_\ell^2)} \\ \frac{1}{1+c_\ell^2} \end{bmatrix} \begin{bmatrix} (1+c_\ell)^2 & -(1-c_\ell^2) \\ -(1-c_\ell^2) & (1-c_\ell)^2 \end{bmatrix} \quad (5.95)$$

Unfortunately, we do not have a closed-form formula for the spectrum of $\mathcal{B}_{h,2h,(\beta_1,\beta_2)}$. Nevertheless, these algebraic combination of blocks can be considered as a reference to the figures and we can illustrate the spectral behavior graphically. It is also important to highlight that this variant defines a deflation operator that is no longer a projection. This is because of the approximation of \hat{A}_h as $\Theta_h\tilde{A}_{2h}$. This is further discussed below.

Recall that $\hat{P}_h = I_h - \hat{A}_h\hat{Q}_h$. We want to prove that \hat{P}_h is projection with assumption that $\hat{Q}_h = I_{2h}^h\hat{A}_{2h}I_h^{2h}$ and $\hat{A}_{2h} = I_h^{2h}M_h^{-1}A_hI_{2h}^h = I_h^{2h}\hat{A}_hI_{2h}^h$. For this proof, we only need to establish the identity $\hat{P}_h^2 = \hat{P}_h$.

$$\begin{aligned} \hat{P}_h^2 &= (I_h - \hat{A}_h\hat{Q}_h)(I_h - \hat{A}_h\hat{Q}_h) \\ &= I_h - \hat{A}_h\hat{Q}_h - \hat{A}_h\hat{Q}_h + \hat{A}_h\hat{Q}_h\hat{A}_h\hat{Q}_h \\ &= I_h - 2\hat{A}_h\hat{Q}_h + \hat{A}_h\hat{Q}_h \end{aligned}$$

where $\hat{Q}_h\hat{A}_h\hat{Q}_h = I_h^h\hat{A}_{2h}^{-1}I_{2h}^h\hat{A}_hI_{2h}^h\hat{A}_{2h}^{-1}I_h^{2h} = I_h^h\hat{A}_{2h}^{-1}\hat{A}_{2h}\hat{A}_{2h}^{-1}I_h^{2h} = I_h^h\hat{A}_{2h}^{-1}I_h^{2h} = \hat{Q}_h$. This is the case when $\hat{A}_{2h} = I_h^{2h}\hat{A}_hI_{2h}^h$, i.e. for the TLKM ideal variant. However if we approximate $\hat{A}_{2h} = \Theta_h\tilde{A}_{2h}$, then we no longer have the equality $\hat{Q}_h\hat{A}_h\hat{Q}_h = \hat{Q}_h$. Subsequently in this case, $\hat{P}_h^2 \neq \hat{P}_h$.

It appears that therefore adding the shift, i.e., the third term involving Q_h in the Expression (5.63) for $P_{h,2h,\lambda_n}$, has a strong influence on the spectrum of the deflated preconditioned operator. To illustrate this claim, we will consider two sub-variants, one with and the other without the shift term Q_h . The analytical expressions resulting from the Fourier analysis of the TLKM practical variant is too complex to be reported

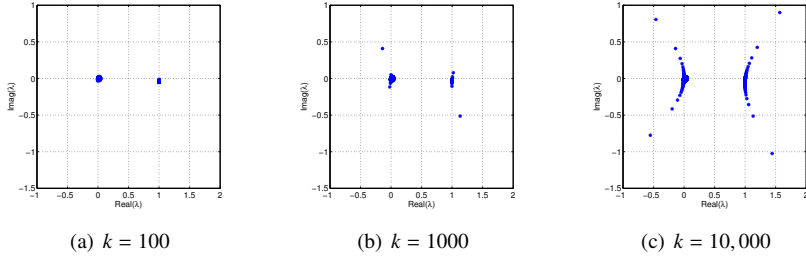


Figure 5.15: Eigenvalues of $B_{h,2h,(\beta_1,\beta_2)}$ in the complex plane considering TLKM practical variant without shift term Q_h for various values of the wavenumber.

here. We therefore resort to numerical computations. Numerical results showing the spectrum are given in Figure 5.15 and Figure 5.16.

In Figure 5.15 we plot the spectrum of $B_{h,2h,(\beta_1,\beta_2)}$ with $\lambda_n = 0$ in (5.63) in the complex plane for various values of the wavenumber k . This figure shows a spectrum consisting of two clusters, the first around the origin and the second around $(1, 0)$. The spread of both clusters appear to grow with the wavenumber k .

Figure 5.16 shows the real part of the eigenvalues of $B_{h,2h,(\beta_1,\beta_2)}$ while shifting the deflated spectrum to 1 in (5.63) versus the index ℓ . The top and bottom row of figures show all indexes ℓ and those close to the near-kernel, respectively. These figures show that the inclusion of the shift in the deflation operator causes the eigenvalues to be shifted both to the left and the right of $(1, 0)$ in the complex plane. This effect is caused by the coarse grid solve with A_{2h} in the matrix-vector multiplication with the matrix Q_h .

5.8 A Comparison: ADEF1 and TLKM

A global comparison of the three prominent variants of deflation analyzed above is presented in this Section. In Figures 5.17, 5.18 and 5.19, the spectrum of the Helmholtz operator preconditioned by ADEF1, TLKM Practical and TLKM ideal variant preconditioners is plotted in the complex plane respectively. The first two figures illustrate the idea that ADEF1 preconditioner is fairly comparable with TLKM practical preconditioner in context of spectrum. The TLKM ideal variant produces a nice spectrum, but it is impractical to use in a multilevel style, hence it remains not practical to use for large problems.

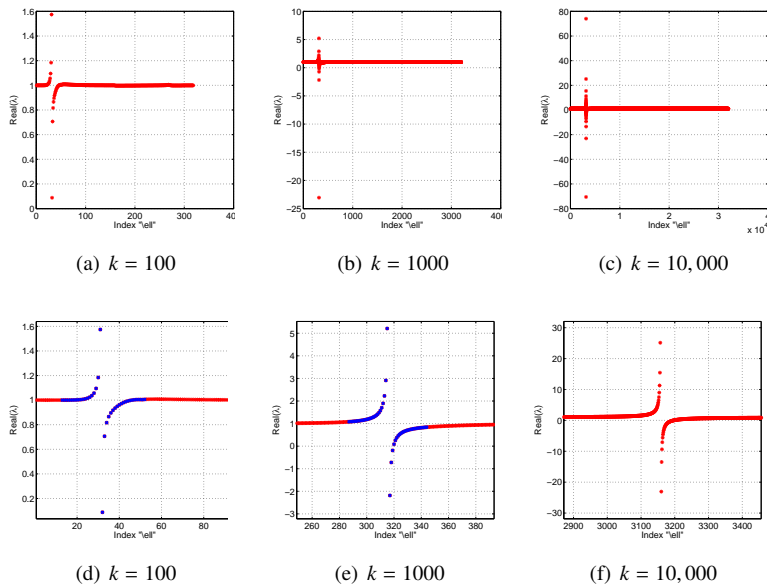


Figure 5.16: Real part of the eigenvalues of $B_{h,2h,(\beta_1,\beta_2)}$ versus the index ℓ using the TLKM practical variant with shift term Q_h for various values of the wavenumber. The top and bottom show all and selection of indices ℓ close to the near-null space, respectively.

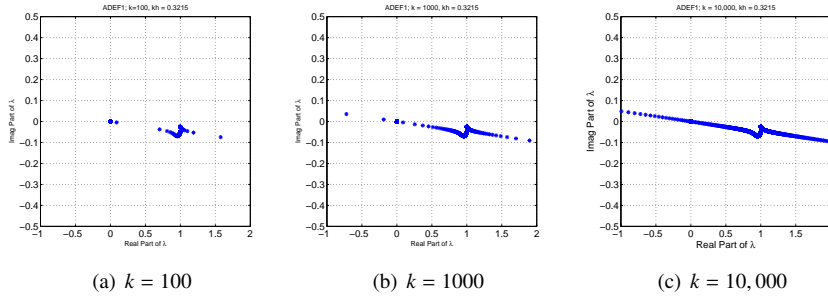


Figure 5.17: ADEF1; $\kappa = 0.3215$.

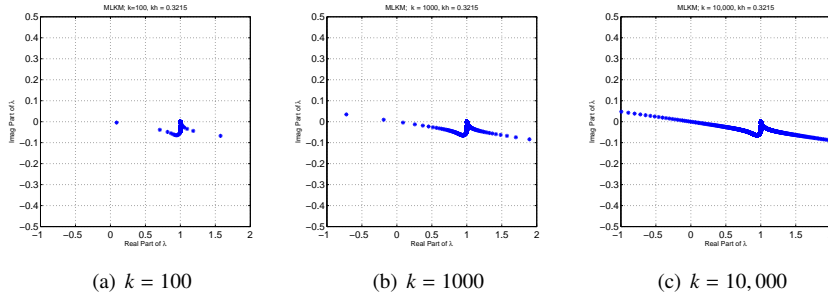


Figure 5.18: MLKM Practical; $\kappa = 0.3215$.

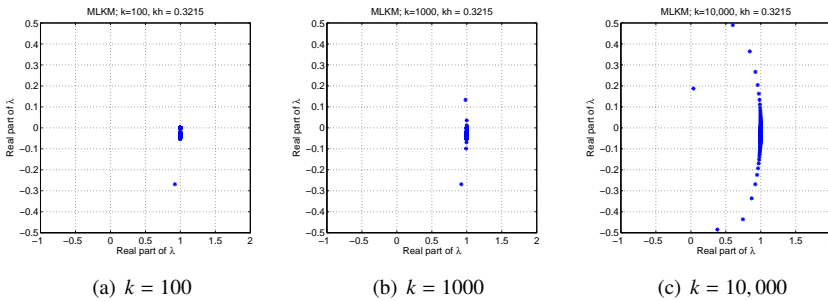


Figure 5.19: MLKM Ideal; $\kappa = 0.3215$.

5.9 Re-Discretized Coarse Grid Operator

In the deflation variants discussed so far, we have used the Galerkin coarse grid operator. Re-discretization has been an alternative to the Galerkin coarse grid operator. We will discuss the deflation variant which uses the re-discretized coarse grid operator. In multilevel algorithms, re-discretization addresses the problem of the stencil

growth at coarser levels in the Galerkin coarsening. It also allows to eliminate all matrices in coarser levels, which helps in reducing memory storage. The spectrum of re-discretized coarse grid operator can be written as

$$\lambda_h(A_{2h}) = \frac{1 - \cos(2l\pi h) - 2k^2}{2h^2} = \frac{\sin(l\pi h) - k^2}{h^2}. \quad (5.96)$$

Below we summarize a number of properties obtained from analysis of this variant of deflation:

- This deflation variant is not a projection. Hence the analysis produces two eigenvalues, which are too involved to be interpreted. Hence we rely upon the graphical interpretation of the analysis.
- The spectrum of the deflated preconditioned operator $B_{h,2h,(\beta_1,\beta_2)} = P_h M_{h,(\beta_1,\beta_2)}^{-1} A_h$ does not seem so convincing as shown in Figure (5.20(a)). However when a shift term Q_h is included, the spectrum is fairly comparable with that of ADEF1, which is shown in Figure (5.20(b)). This shows a prospective of using the re-discretized coarse grid operator for large problems in order to avoid large memory storage for the Galerkin coarse grid operator.
- The above discussed figures show the spectrum from analysis, where Dirichlet boundary conditions are used. However, inclusion of Sommerfeld boundary conditions visibly changes the spectrum, which is given in Figures (5.20(c)) and (5.20(c)).

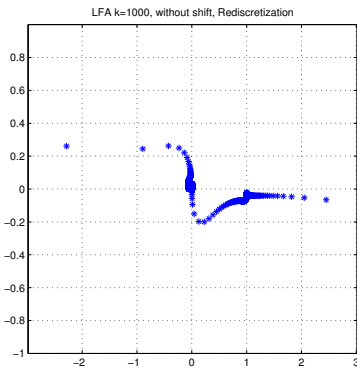
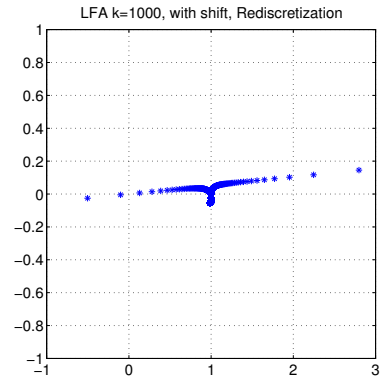
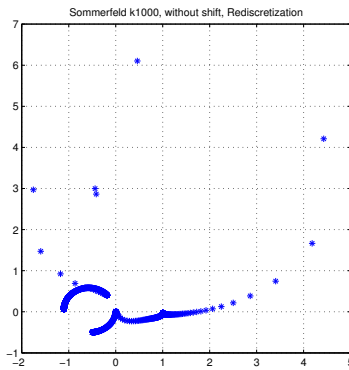
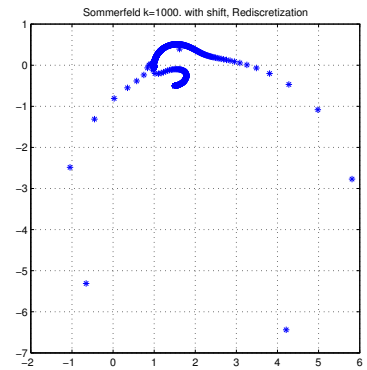
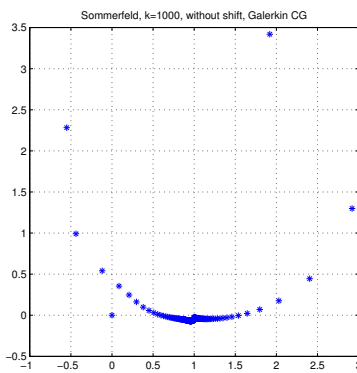
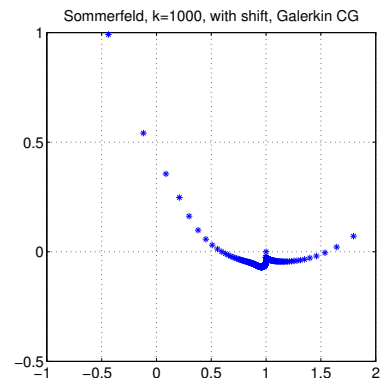
(a) $k = 1000$ (b) $k = 1000$ (c) $k = 1000$ (d) $k = 1000$ (e) $k = 1000$ (f) $k = 1000$

Figure 5.20: Spectrum of deflated operators using re-discretized coarse grid operator(a-d) and Galerkin coarse grid operator (e-f).

Numerical Experiments

This chapter is dedicated to numerical results for the problems defined in Chapter 2, which ranges from a one dimensional to three dimensional Helmholtz problems with simple heterogeneity. These numerical results will justify most of the theoretical findings presented in earlier chapters of this thesis. The numerical results give a detailed comparison of solvers with and without deflation. The details of solvers and corresponding results are presented in tables and figures in respective sections. Experiments have been performed in MATLAB and PETSc and are mentioned individually. With every choice of (flexible) Krylov method, a zero initial guess is used in all experiments. Krylov iterations are stopped if the scaled residual $\frac{\|r_n^k\|}{\|b_n\|}$ is reduced by a factor of 10^7 , except an exclusive mention is given.

6.1 One-Dimensional Constant Wave Number Problem

For discretization of wave scattering phenomena, it is often required to increase the grid size with increasing wave number. In other words the mesh size is kept as an increasing function of the wave number instead of constant function. Table 6.1 aims at testing the ADEF1 preconditioner at two levels for increasing wavenumber problems, where the grid size is such that k^3h^2 is kept constant, as recommended in [12]. The numerical experiment is carried out on one dimensional problem 2.7 with Sommerfeld boundary conditions 2.9, with a two-level ADEF1 preconditioner, where the coarse problem as well as CSLP are inverted exactly. Full GMRES has been chosen as Krylov solver. The shifts $(\beta_1, \beta_2) = (1, 0.5)$ are used. In Table 6.1, the difference is made clear between the choices of grid resolutions; $k^3h^2 \leq 0.625$ and $kh \leq 0.625$. Table 6.1 immediately shows constant number of iterations where as wave number ranges from 10 to 1000 while choosing criteria $k^3h^2 \leq 0.625$ and an increase in number of iterations has been observed while $kh \leq 0.625$.

k	$N(k^3 h^2 \leq 0.625)$	iter	$N(kh \leq 0.625)$	iter
10	44	4	16	4
20	116	4	32	5
40	320	4	64	5
80	800	4	128	6
100	1268	4	160	7
200	3572	4	320	8
400	10124	4	340	10
500	14144	4	800	12
600	18596	4	960	13
800	28628	4	1280	15
1000	40004	4	1600	17

Table 6.1: GMRES iterations preconditioned by Two-Level deflation preconditioner applied on one a dimensional Helmholtz problem with Sommerfeld boundary conditions.

Next we like to see how ADEF1 preconditioner performs for very large wavenumber problems. For the purpose, we choose one dimensional Helmholtz on interval $(0, 1)$ supplied with Sommerfeld boundary conditions. We use the shift $(\beta_1, \beta_2) = (1, 1)$ and 20 grid points per wavelength i.e. $kh = 0.3125$. Two-level deflation has been opted where coarse grid operator as well as CSLP has been inverted exactly. A comparison of the number of GMRES iterations preconditioned by CSLP and ADEF1 two-level preconditioner is given in Figure (6.1). In Figure (6.1(a)) and Figure (6.1(b)) we consider the wave number in the range of $0 \leq k \leq 800$ and $1000 \leq k \leq 20,000$, respectively. These figures show that in the wavenumbers range up to $k = 800$, the number of iterations is almost independent of k . Starting at $k = 1000$ however, we observe an increasing dependence of number of iterations on the wavenumber.

6.2 Two-Dimensional Constant Wave Number Problem

In this section, first we test our ADEF1 preconditioner at two-levels for a two dimensional problem with constant wave number. CSLP preconditioner involves shifts $(1, 0.5)$ and ADEF1 deflation is used at two-levels. In Table 6.2 and 6.3, we give the number of GMRES iterations required to solve the problem with Dirichlet and Sommerfeld boundary conditions for a range of wavenumbers and number of elements (for a fixed wavenumber) respectively. We compare the variants with and without deflation. In both tables, only the number of elements on and below the diagonal highlighted in bold suffice to meet the requirement of 20 grid points per wavelength, corresponding to the condition $kh = 0.3125$. The number of iterations with and with-

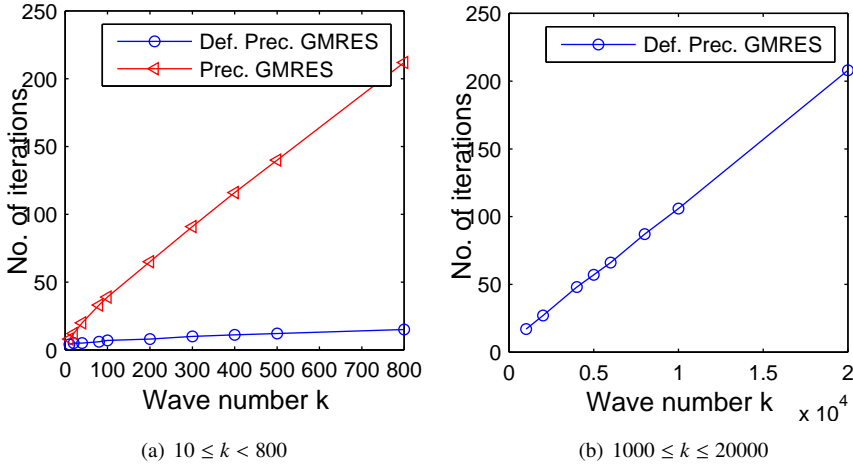


Figure 6.1: Number of $M_{h,(1,1)}$ preconditioned GMRES iterations with and without deflation for the one-dimensional constant wave number problem for k ranging between 10 and 800 (left) and between 1,000 and 20,000 (right).

	$k = 10$	$k = 20$	$k = 30$	$k = 40$	$k = 50$	$k = 100$
$n = 32$	3/10	8/17	17/31	35/50	52/80	13/14
$n = 64$	3/10	6/17	10/30	17/47	24/63	221/252
$n = 96$	3/10	5/17	7/30	11/46	15/62	209/220
$n = 128$	3/10	5/17	6/30	10/45	11/62	90/196
$n = 160$	3/10	4/17	5/30	8/45	9/62	65/194
$n = 320$	2/10	3/17	4/30	5/45	6/61	24/193

Table 6.2: Number of GMRES iterations for two dimensional constant wavenumber problem with Dirichlet boundary conditions for various wave numbers and grid resolutions using the SLP preconditioner $M_{h,(1,0.5)}$ with and without deflation.

out deflation are separated by the symbol “/” on left and right respectively. For both Dirichlet and Sommerfeld boundary conditions, the number of iterations for fixed k decreases with increasing grid points n , and thus increasing number of grid points per wavelength. This confirms findings from our Fourier analysis. The diagonal just above the highlighted one corresponds to $kh = 0.625$, which is 10 grid point per wavelength. For this often used discretization, the growth in the number of iterations is larger than for the case that $kh = 0.3125$. These tables are also aimed to compare with inclusion of damping term due to Sommerfeld boundary conditions. Comparing both tables confirms the claim that the problem with Dirichlet boundary conditions acts as a worst case for the problem with Sommerfeld boundary condition .

	$k = 10$	$k = 20$	$k = 30$	$k = 40$	$k = 50$	$k = 100$
$n = 32$	5/10	8/17	14/28	26/44	42/70	13/14
$n = 64$	4/10	6/17	8/28	12/36	18/45	173/163
$n = 96$	3/10	5/17	7/27	9/35	12/43	36/97
$n = 128$	3/10	4/17	6/27	7/35	9/43	36/85
$n = 160$	3/10	4/17	5/27	6/35	8/43	25/82
$n = 320$	3/10	4/17	4/27	5/35	5/42	10/80

Table 6.3: Number of GMRES iterations for two dimensional constant wavenumber problem with Sommerfeld boundary conditions for various wave numbers and grid resolutions using the SLP preconditioner $M_{h,(1,0.5)}$ with and without deflation.

In Table 6.4 we give the number of outer GCR [36] iterations required to solve the problem discussed using the multilevel algorithm with ADEF1 for various values of the wavenumber using 20 grid points per wavelength. Implementation of the ADEF1 preconditioner at multilevel requires to choose a flexible Krylov method, since we approximate the solution of the coarser problem at every level. We recursively apply coarsening until we obtain a single grid point on the coarsest level. The algorithm has been implemented in MATLAB [78]. In Table 6.4 the abbreviation TL stands for the two-level algorithm considered before while the indices n_1 , n_2 and n_3 in the notation ADEF1 (n_1, n_2, n_3) denote the number of GCR iterations on the first, second and third coarser levels, respectively. On the next coarser levels a single GCR iteration is used, except for the coarsest level where a direct solver is employed. A single standard V(1,1)-cycle with damped Jacobi using as damping parameter $\omega = 2/3$ as a smoother was used to approximate the CSLP preconditioner, which uses the shifts $(\beta_1, \beta_2) = (1, 1)$. Table 6.4 aims to discuss the number of iterations required at different coarser levels in order to achieve an optimized algorithm. Table 6.4 shows that reducing outer iterations for the range of wave numbers considered requires increasing n_1 (the number of GMRES iterations on the first coarser level). The number of iterations at subsequent coarse levels (other than n_1) are not as such sensitive as on finest level. The combination $(n_1, n_2, n_3) = (10, 2, 1)$ seems to be attractive however $(8, 2, 1)$ is a fair competitor, as 2 iterations might reduce more computations cost than the gain in global Krylov iterations.

6.3 Shifts in SLP

A good choice of the shifts (β_1, β_2) in CSLP is important, since it defined globally the convergence of preconditioned Krylov solver. The smaller the imaginary shift β_2 , the better the CSLP preconditioner resembles the original operator. However the proposed preconditioner solver (i.e. multigrid) does not approximate good enough for smaller β_2 . Optimal choice of the shifts has also been discussed in [30, 29]. How does

	$k = 10$	$k = 20$	$k = 40$	$k = 80$	$k = 160$
TL	6	7	11	15	25
ADEF1(4,2,1)	9	11	16	27	100+
ADEF1(6,2,1)	9	10	14	21	47
ADEF1(8,2,1)	9	10	13	20	38
ADEF1(8,3,2)	9	10	13	19	37
ADEF1(10,2,1)	9	10	14	19	32

Table 6.4: Number of outer GCR iterations preconditioner by ADEF-1 for the two-dimensional problem with Sommerfeld boundary conditions for various wave numbers and $kh = 0.3125$ using the shifts $(\beta_1, \beta_2) = (1, 1)$.

deflation influence the choice of shifts for CSLP? In order to see this, in Figure (6.2) we plot the required number of GMRES iterations to solve a two dimensional constant wavenumber and wedge problem with first order Sommerfeld boundary conditions as a function of the imaginary shift β_2 . In constant wavenumber and wedge problem, we used wavenumber $k = 50$ and frequency $f = 30$, respectively, and employed 10 grid points per wavelength. We have chosen $\beta_1 = 1$ and allowed β_2 to vary between 0 and 1. For $\beta_2 = 0$, the CSLP preconditioner coincides with the discrete Helmholtz operator, and the algorithm converges in a single iteration. The figures show that without deflation, the number of GMRES iterations increases with β_2 . This is due to the fact that the CSLP preconditioner differs more from the discrete operator as β_2 increases as observed in, e.g., [40]. More interestingly, the figures shows that with deflation, the required number of GMRES iterations initially increases but remains constant for $\beta_2 \geq 0.1$. These results confirm the Fourier analysis spectrum in Section (5.4) and opens promising perspectives on obtaining a good preconditioner at low cost.

6.4 Two-Dimensional Non-Constant Wave Number Problem

6.4.1 Wedge Problem

In this subsection we consider the two-dimensional wedge problem as defined in Chapter 2, where the frequency is a function of speed which is different in the three wedges ranging from 1500 m/s to 3000 m/s . Similar to Tables given in the Section 6.2, in Table 6.5 we give the required number of GMRES iterations for various frequencies and various mesh sizes with and without two-level deflation. We follow the same pattern and same solver type which is Krylov method GMRES is preconditioned with CSLP and deflation at two levels. CSLP is inverted exactly as well as the coarse

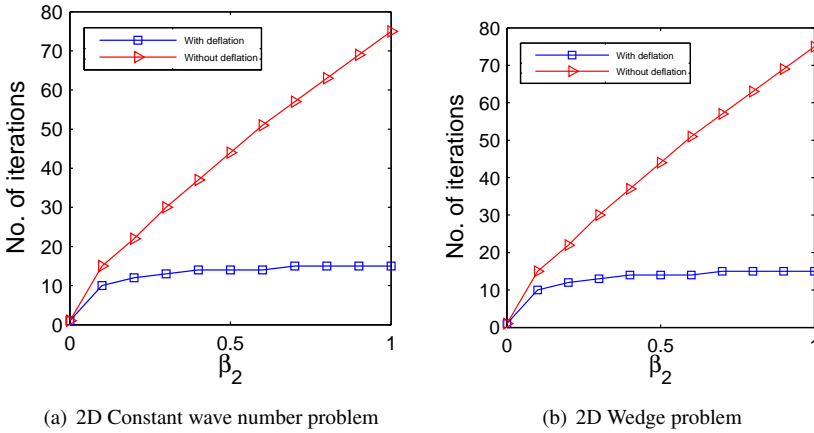


Figure 6.2: Number of $M_{h,(1,\beta_2)}$ preconditioned GMRES iterations with and without deflation versus β_2 for Problem (1) for $k = 50$ and Problem (2) for $f = 30$, both problems with Sommerfeld boundary conditions.

	$freq = 10$	$freq = 20$	$freq = 30$	$freq = 40$	$freq = 50$
74×124	7/33	20/60	79/95	267/156	490/292
148×248	5/33	9/57	17/83	42/112	105/144
232×386	5/33	7/57	10/81	25/108	18/129
300×500	4/33	6/57	8/81	12/105	18/129
374×624	4/33	5/57	7/80	9/104	13/128

Table 6.5: Number of $M_{h,(1,0.5)}$ preconditioned GMRES iterations with and without deflation for Problem (2) for various wave numbers and grid resolutions.

problem. CSLP uses shifts $(\beta_1, \beta_2) = (1, 0.5)$. MATLAB is used to perform experiments here. In this table the leftmost column indicates the number of grid points used in each coordinate direction. This table shows that while deflation is effective in reducing the number of iterations even in the case of non-constant wave numbers, not all near-null space of the system are sufficiently removed in order to obtain a fully scalable algorithm. But gain in number of iterations is attractive with the two-level deflation. In the next section, we will see if the sufficient reduction in number of iteration with deflation pays off the solve time.

6.4.2 Marmousi Problem

Next considered is the adapted Marmousi problem; i.e. the original domain (3000×9200) in meters has been trimmed into $\Omega_h = (2048 \times 8192)$ where the discrete ve-

Frequency	Solve Time		Iterations	
	CSLP-F	ADEF1-F	CSLP-F	ADEF1-F
1	1.22	5.07	13	7
10	10.18	9.43	112	13
20	72.16	60.32	189	22
40	550.20	426.79	354	39

Table 6.6: CSLP and ADEF1 performance comparison for Marmousi problem without damping i.e. $\alpha = 0$.

locity data can be coarsened geometrically and conveniently. The original velocity has also been adapted to make it less contrasted and the velocity $c(x,y)$ range is $2587.5 \leq c \leq 3325$. The grid is resolved for different frequencies such that for the maximum wavenumber k , $kh \leq 0.039$ for $f = 1$ and $kh \leq 0.39$ for $f = 10, 20$ and 40 . This scheme guarantees that one wavelength is resolved into more than 10 grid points. Without deflation, the Krylov method of choice is Bi-CGSTAB, where as multilevel implementation of ADEF1 deflation requires to choose flexible Krylov, therefore FGMRES has been used. The combination of number of FGMRES iterations at second, third and subsequent coarser levels is (8, 2, 1, ...) respectively. We choose two competitors in preconditioners i.e. CSLP and ADEF1. In both cases, a multigrid full cycle approximates the CSLP with standard coarsening in each direction and using weighted Jacobi smoother with $\omega = 2/3$. Also in both cases, the shifts in CSLP are such that $M_h(1, 1)$. Since we use standard multigrid components, therefore the CSLP with shifts (1, 1) is more accurately approximated than with shifts (1, 0.5). However if one adapt the multigrid, say to semi-coarsening then the CSLP with shifts (1, 0.5) should be preferred. Numerical experiments are performed in the C++ based tool PETSc [7] on Dell Precision machine with processor E8400 at 3.00GHz. Table 6.6 presents a comparison between iterations taken by CSLP and ADEF1 preconditioned Krylov, which confirms the trend that the use of ADEF1 deflation lowers the number of iterations significantly specially for high wavenumbers. And that reduced iteration count for ADEF1 results in a speedup at sufficient high wavenumber. This fact is concluded from the numbers showing solve time using both CSLP and ADEF1 preconditioner in the table under discussion.

In Table 6.7, we repeat the same but here we include the damping parameter $\alpha = 0.05$. Damping renders the diagonal dominance in obvious way. Therefore, one can note a reduction in number of iterations and subsequently in time.

Considering more grid points for a fixed frequency, the Krylov method takes less number of iterations with both preconditioners CSLP and ADEF1. This is evident in

Frequency	Solve Time		Iterations	
	CSLP-F	ADEF1-F	CSLP-F	ADEF1-F
1	1.27	5.76	13	7
10	9.85	10.36	96	13
20	68.6	65.34	175	19
40	984.7	369.7	332	33

Table 6.7: CSLP and ADEF1 performance comparison for Marmousi problem without damping i.e. $\alpha = 0.05$.

Frequency	Solve Time		Iterations	
	CSLP-F	ADEF1-F	CSLP-F	ADEF1-F
$f = 1$	1.23	5.08	13	7
$f = 10$	40.01	21.83	106	8
$f = 20$	280.08	131.30	177	12
$f = 40$	20232.6	3997.7	340	21

Table 6.8: CSLP and ADEF-1 performance comparison for Marmousi problem without damping i.e. $\alpha = 0$ for Marmousi problem discretized with a mesh size equivalent of 20 grid points per wavelength.

Table 6.8, where the number of iterations as well as solve time respective of these number of iterations are compared. Mesh size is settled such that more than 20 grid points are assured for one wavelength. As the Marmousi problem involves heterogeneity, thus attentive choice of multigrid ingredients for CSLP preconditioner solve can produce better results. An example of which is given in [40], where two smoothing steps are suggested to perform in order to get relatively more accurate CSLP preconditioner solve. Also semi-coarsening scheme can be performed for more accurate preconditioner solve [15]. However we continue with the multigrid full cycle with standard coarsening and 1 pre- and post-smoothing steps. This can be seen as a worst case scenario.

6.5 Three-Dimensional Constant Wave Number Problem

Not all the algorithms developed for and tested on two dimensional problems show similar performance behavior for three dimensional problem. In order to assure the functioning of our proposed deflation operator ADEF1, we present a test on three dimensional constant as well as non-constant wavenumber problem as follow.

Wave number k	Solve Time		Iterations	
	SLP-F	ADEF1-F	SLP-F	ADEF1-F
5	0.007	0.055	7	9
10	0.06	0.46	9	10
20	1.07	3.2	21	11
40	21.79	31.99	58	16
60	113.19	165.2	90	23
80	511.80	501.9	159	29
120	2832.7	2056.7	254	39

Table 6.9: PETSc Solve time and iteration comparison between Bi-CGSTAB preconditioned with CSLP-F and FGMRES(20) preconditioned with ADEF1-F

A three dimensional Helmholtz in a unit cube domain with constant wavenumber is considered. We intend to analyse that at what factor, the number of iterations reduces when we use ADEF1 preconditioner in comparison with CSLP preconditioner. Further we will see if there is any gain in solve time with preconditioner ADEF1.

Table 6.9 and Table 6.10 present the FGMRES iteration count using 10 and 20 grid points per wavelength respectively. These tables list the number of outer Krylov iterations and CPU-time for various values of the wavenumber. CSLP as solely preconditioner involves the shift (1,0.5), where as as a part of ADEF1 preconditioner, the shift (1, 1) is preferred. The number of Krylov iterations at second, third and subsequent levels are settled as (8, 2, 1, ...). Both preconditioners are implemented in PETSc. The solve time reported excludes the time required for the computation of the Galerkin products in the set-up phase of the two algorithms. This set-up time is fairly comparable for both algorithms. From these tables it is clear that the use of ADEF1 results in a reduction in the number of iterations that increases with the wave number. Table 6.9 in particular shows that without the ADEF1 preconditioner, the number of iterations increases more than linearly with the wavenumber. With the ADEF1 deflation preconditioner this increase is slower than linear. The CPU-time in this table shows a cross-over point between $k = 60$ and $k = 80$. The use of a ADEF1 deflation preconditioner also results in speed-up for sufficiently large wave numbers. For the largest wave number reported, the use of the ADEF1 deflation preconditioner a thirty percent reduction in solve time. Table 6.10 shows that with 20 grid points per wavelength the number of ADEF-1 iterations remains almost constant in the range of wave numbers considered. The ADEF-1 solvers outperforms the CSLP solver in CPU-time starting at wave number $k = 40$. For the largest wave number, a speed-up of forty percent is reported.

CSLP preconditioned Krylov proves to be computationally costly for large wavenum-

Wave number k	Solve Time		Iterations	
	SLP-F	ADEF1-F	SLP-F	ADEF1-F
5	0.04	0.32	7	8
10	0.48	2.32	9	9
20	8.14	17.28	20	9
40	228.29	155.52	70	10
60	1079.99	607.45	97	11

Table 6.10: PETSc Solve time and iterations for three dimensional problem with wavelength resolved over **20 grid-points**. Bi-CGSTAB has been preconditioned by CSLP-F, where flexible GMRES is preconditioned by ADEF1.

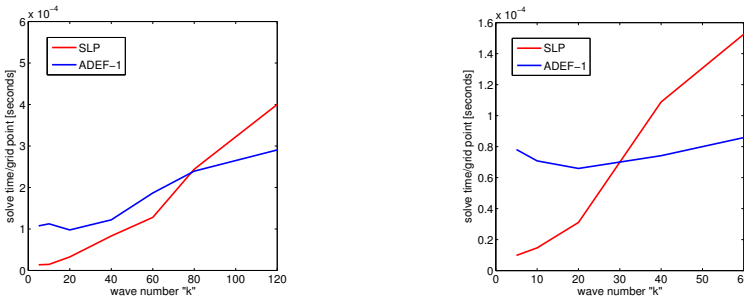


Figure 6.3: CPU time per grid point in constant wavenumber problem using 10 (left) and 20 (right) grid points per wavenumber .

ber problems, than that of Krylov preconditioned by ADEF1. Figure 6.3 shows the crossover. The CPU-time per grid point for both solvers using 10 and 20 grid points per wavelength is plotted versus the wavenumber in the left and right part of Figure 6.3, respectively. These figures show how the CPU time using the solver variant with deflation increases at a slower rate and is therefore more attractive to use at a given wavenumber. This wavenumber is equal to $k = 80$ and $k = 30$ in case that 10 and 20 grid points per wavelength are used, respectively.

6.6 Three-Dimensional Non-Constant Wave Number Problem

In this section, we perform experiments on three dimensional layered problem as defined in Section 2.4 and domain of interest is layered as seen in Figure 2.4. Numerical results for this problem using 10 and 20 grid point per wave length are given in Ta-

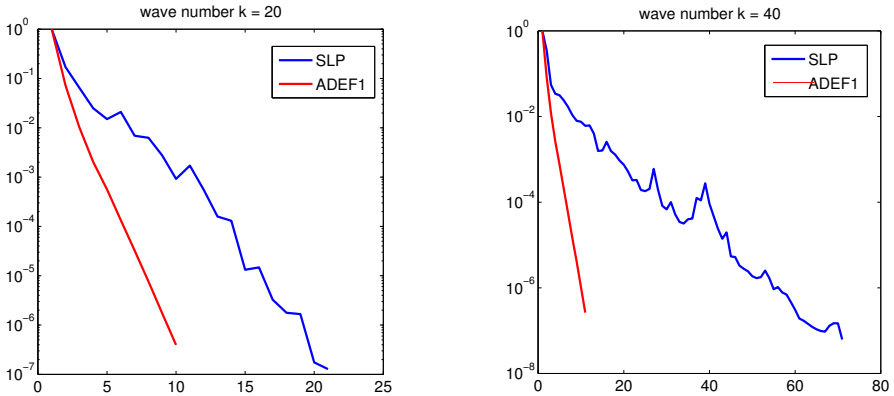


Figure 6.4: Comparison of convergence history of CSLP and ADEF1 preconditioners for three dimensional problem with wave number $k = 20$ and $k = 40$ on left and right figures respectively.

Wave number k	Solve Time		Iterations	
	SLP-F	ADEF1-F	SLP-F	ADEF1-F
5	0.09	0.24	9	11
10	1.07	1.94	15	12
20	16.70	18.89	32	16
30	73.82	78.04	43	21
40	1304.2	214.7	331	24
60	xx	989.5	xx	34

Table 6.11: PETSc Solve time comparison between BiCGSTAB preconditioned with SLP-F and FGMRES(20) preconditioned with ADEF1-F. Discretization satisfies the relation $kh \leq 0.625$ i.e. 10 gp/wl

ble 6.11 and Table 6.12, respectively. The layered heterogeneity is reflected in an increase in the required number of outer Krylov iterations of both solvers. The problem for $k = 60$ in particular could not be solved using the CSLP solver. The use of ADEF-1 is again seen to result in a reduction in number of iterations and to lower the increase in number of iterations in case that 10 grid points per wave length are used. The use of ADEF-1 results in a six-fold reduction in CPU-time for $k = 40$ and in a converged solution for $k = 60$. In case that 20 grid points per wave length are used, ADEF-1 renders the iteration count almost constant in the range of wave numbers considered. For the largest wave number $k = 30$, the use of ADEF-1 results in a more than ten-fold reduction

Wave number k	Solve Time		Iterations	
	SLP-F	ADEF1-F	SLP-F	ADEF1-F
5	0.6	1.4	9	9
10	7.5	10.04	14	9
20	324.1	79.2	72	9
30	3810.9	361.7	285	11

Table 6.12: PETSc Solve time comparison between BiCGSTAB preconditioned with CSLP-F and FGMRES(20) preconditioned with ADEF1-F. Grid resolution is such that there are 20 gp/wl.

6.7 Finite Element Discretization

In this section, we discuss the finite element method (FEM) discretization for the Helmholtz equation. For this purpose, we take the example of a two-dimensional Helmholtz equation defined on unit square domain. For complete description for the problems, the reader is referred to Chapter 2. Comsol [27] has been used to discretize the problem with standard FEM. The given geometry is discretized into a uniform mesh with triangular elements. A mesh with triangular elements assures a better imitation of finite difference discretization. However the mass matrix is not diagonal.

6.7.1 Algebraic Deflation Vectors

The necessity of FEM discretization is mostly derived by the irregular domains. The Helmholtz equation also finds applications in many fields with such irregular domains. In order to apply our deflation preconditioner, we need to have deflation vectors. Recall that our deflation vectors are simply grid transfer vectors, typically used in multigrid. For the irregular domains and irregular grid, one has to resort to algebraically constructed grid transfer operators conveniently, in the same way where algebraic multigrid is used for problems on irregular domains. For this reason, we also construct the algebraic grid transfer operators. We obtain deflation vectors by an application of the algebraic multigrid to the Poisson equation. The algebraic grid transfer operators extracted during the application of Algebraic multigrid (AMG) to the two dimension Helmholtz equation, while keeping the wave number $k = 0$. In other words we make use of the stiffness matrix from the Helmholtz operator to construct the deflation vectors. It is important to mention that keeping $k = 0$, the stiffness matrix is real valued.

6.7.2 Performance of Algebraically coarsened FEM discretization

The performance of different solvers based on the CSLP for FEM discretization and in combination with deflation preconditioner ADEF1 is discussed. ADEF1 precon-

ditioner is constructed using algebraic deflation vectors. The Helmholtz as well as CSLP are FEM discretized. We present the results for the two dimensional constant wave number Helmholtz defined on a unit square domain, where Sommerfeld boundary conditions are used. In FEM discretization, the degrees of freedom are such that at least 10 grid points are taken for one wavelength. The algorithms for both preconditioners are implemented in PETSc [7]. It is important to mention that algebraic coarsening is slower than geometric coarsening. Particularly at the first coarser level. For example, for the two dimensional problem, the coarse matrix at first coarse level is coarsened at factor of 2, whereas the same is typically coarsened at factor of 4 in standard geometric coarsening.

Flexible GMRES [95] has been chosen as outer Krylov as well as inner Krylov method. The iterations are stopped once the residual meets the tolerance 10^{-6} . The numbers representing iteration count along with solve time in seconds are presented in comprehensive Table 6.13. The solve time can be read in brackets, where as the number out of bracket shows global iterations. The ADEF1 preconditioner is implemented in a multilevel fashion, where the first and second coarser problems are allowed 8 and 2 Krylov iterations respectively. The rest of the coarser problems are given 1 iteration each. Numbers are given for different wavenumber ranging from $k = 10$ to $k = 200$. Different variation in preconditioner type is described below.

CSLP-D	:	CSLP $M(1, 1)$ inverted by LU
CSLP-F	:	CSLP $M(1, 1)$ approximated by full multigrid
Two-Lev	:	Two Level variant of ADEF1 preconditioner.
ADEF1-V	:	ADEF1, with $M(1, 1)$ approximated by multigrid VCycle.
ADEF1-F	:	ADEF1, with $M(1, 1)$ approximated by multigrid full cycle.
ADEF1-D	:	ADEF1, with $M(1, 1)$ inverted by LU.

Furthermore, the FEM discretization in combination with algebraically constructed ADEF1 preconditioner is compared with FDM discretization with geometrically constructed ADEF1 preconditioner. The results from two Tables 6.13 and 6.14 are discussed as follows:

- FEM based CSLP does not perform good. This is because of the deterioration in the CSLP preconditioner. We have noticed non-negative off-diagonal entries in the CSLP matrix. Even inverting CSLP directly does not show any speed up.
- The FEM discretized and algebraically constructed ADEF1-F preconditioner takes less iterations when compared with counterpart FDM and geometrically constructed ADEF1-F. However, the FEM solver costs more computational time. It is apparent, as the algebraic coarsening is slower, particularly at first coarse

level and this results in coarse meshes with more degrees of freedom than geometric variants. This produces more deflation vectors.

- Approximation of CSLP is sensitive while used in combination with the ADEF1 preconditioner. The difference can be seen, when CSLP is approximated by a multigrid V-Cycle or F-Cycle. More accurate approximation with multigrid F-cycle reduces the global number of iterations significantly.
- With FEM, the two-level ADEF1 preconditioner performs better than in the FDM case. This is again because of more deflation vectors.
- With FEM discretization and algebraic ADEF1 preconditioner, the ADEF1 preconditioner rules out CSLP in terms of number of iterations, however it is costly in terms of computational time.

Solver	FEM/ FDM	k=10	20	40	80	120	160	200
CSLP-D	FEM	15	30	57	108	157	204	252
CSLP-F	FEM	22	43	72	128	178	232	278
ADEF1-F	FEM	7	8	10	16	19	24	27
	FDM	10	11	15	24	32	41	51
Two-Lev	FEM	6	8	10	15	20	26	32
	FDM	7	10	14	23	37	61	87
ADEF1-V	FEM	22	40	66	118	166	214	258
	FDM	16	27	58	116	177	235	292
ADEF1-D	FEM	6	8	10	15	19	24	27
	FDM	7	10	14	21	29	36	43

Table 6.13: Comprehensive iteration count; FEM compared with FDM discretization with different solver types and difference wave numbers.

Solver	FEM/ FDM	k=10	20	40	80	120	160	200
CSLP-D	FEM	0.02	0.07	0.57	5.8	22.6	59.6	130.5
CSLP-F	FEM	0.05	0.16	0.85	6.33	21.81	55.7	115.9
ADEF1-F	FEM	0.25	0.85	2.4	15.2	38.3	81.47	144.5
	FDM	0.6	1.6	4.5	15.7	28.2	70.1	103.9
Two-Lev	FEM	0.02	0.05	0.32	2.46	8.4	21.4	43.8
	FDM	0.00	0.03	0.27	2.17	8.8	27.9	67.8
ADEF1-V	FEM	0.27	1.27	5.4	32.8	110.8	240.6	447.0
	FDM	0.25	0.8	3.6	18.4	50.3	125.2	233.1
ADEF1-D	FEM	0.07	0.5	2.9	23.68	80.40	191.84	387.30
	FDM	0.05	0.2	1.26	9.04	31.6	76.3	149.8

Table 6.14: Comprehensive solve time; FEM compared with FDM discretization with different solver types and difference wave numbers.

Conclusions and Future Work

7.1 Conclusions

This thesis is attributed to preconditioners for the indefinite Helmholtz equation, which has defied the iterations solution techniques for a long time. Work presented here is aimed at obtaining the solvers which are robust and computational cost affordable. It has been hard to develop optimal solvers independent of the parameters in the equation in general and in particular for Helmholtz with large wavenumber k (or frequency f) as parameter wavenumber k (or frequency f) induces the indefiniteness. Multigrid and Krylov methods have been appealing for large sparse linear system arising particularly from elliptic-type equations. The performance of later one depends on a good preconditioner in most cases. The vital theme discussed here is also deflation type preconditioners for the Helmholtz equation to precondition a Krylov method. Multigrid is also used, to approximate the complex shifted Laplace preconditioner (CSLP). Implementation complications were elaborated and spectral insights were discussed via rigorous Fourier analysis (RFA). There are a number of conclusions from this work which are summarily presented below.

In Chapter 3 we presented the classical preconditioners comprehensively followed by a class of (complex) shifted Laplace preconditioners CSLP. Shifts in CSLP are important to determine the efficiency of CSLP. Shifts has been settled as $(1, 0.5)$ to $(1, 1)$ in order to get an acceptable approximation of CSLP. Suggested by results, CSLP has been considered as base method to compare performance with our proposed deflation preconditioner which is called ADEF1. CSLP is combined with the ADEF1 preconditioner.

Generally, the Krylov solver suffers from stagnation due to few bad eigenvalues.

The deflation preconditioner has gained popularity due to its simple approach to tackle "a few bad eigenvalues". However there is no established theory about deflation vectors for non-symmetric (non Hermitian) matrices. An important issue of a deflation preconditioner is how to choose suitable deflation vectors. In Chapter 4, deflation is explained in general and in the context of the Helmholtz equation. Choices of deflation vectors for the Helmholtz problem have been discussed. The CSLP preconditioned Helmholtz preconditioner encounters small eigenvalues in modulus. The multigrid inter-grid transfer operators have been chosen as deflation vectors, as the "bad" eigenvalues correspond to "small" frequency nodes. The choice is justified theoretically as they are typically used in multigrid to resolve the small frequency nodes by restricting them on coarser grids, solving the coarse grid problem and then interpolating the corrections.

Variations of the deflation preconditioner are considered. They are mainly distinguished by the choice of coarse grid operator, which is a Galerkin operator in most variants. The deflation preconditioner based on the Helmholtz operator is called ADEF1 preconditioner, nevertheless it is implemented in combination with the CSLP preconditioner. The TLKM deflation preconditioner is the one which is constructed using the CSLP preconditioned Helmholtz operator. Large problems produces large (enough) coarse grid problems, which are still impractical to solve exactly. An iteration scheme for the inverse of the coarse grid operator is considered. The Galerkin coarse grid system inherits the characteristics of the Helmholtz operator, hence the coarse grid operator is preconditioned by CSLP and a deflation preconditioner at a coarser level. Recursively solving deflated and preconditioned operators at subsequent coarser levels gives rise to a *multilevel* deflation preconditioner in both case; ADEF1 and MLKM. The coarse grid operator in MLKM is difficult to implement in multilevel fashion. An approximation to the coarse grid operator has been suggested in order to make the multilevel implementation possible and to make it computationally feasible. This approximation gives rise to the ideal and practical variant of MLKM. The deflated eigenvalues are supposed to be projected to zero, however the approximate coarse system solve disturbs of deflated eigenvalues around zero. This dispersion of eigenvalues around zero caused by approximate coarse system solve effects Krylov convergence in a bad way. As a remedy, the shifting of deflated spectrum was performed by including a shift term in both deflation preconditioners.

All the claims and facts are well elaborated and justified by RFA, which is presented in Chapter 5. One dimensional and two dimensional problems are used to analyze. Dirichlet boundary conditions are employed, as they lack the damping term thus are considered the worst with respect to other boundary conditions. RFA possibly explores all the variations of the deflation preconditioner and lists the findings and gives a comparison between them. The bottom line of the RFA of TLKM, both ideal

and practical variants, is that the ideal variant produces an attractive spectrum which is well clustered around 1 in the complex plane, however this is near-impossible to implement, especially in multilevel fashion. Even in the two-level case, it is not computationally attractive. The TLKM practical variant has been an alternative to ideal variant. However, this approximation of the coarse grid operator disturbs the spectrum in a severe way, when compared to the TLKM ideal variant. The spectrum is fairly comparable with that of the ADEF1 preconditioner, which is more cost efficient in implementation. This can be summarized as follows: ADEF1 is the best method compared to all variants presented in this chapter. The coarse grid operator, being the Galerkin operator of the (preconditioned) Helmholtz operator, carries on the near-null space modes. Inverting the coarse grid operator in all deflation preconditioners shoot out some of eigenvalues to large values in modulus.

The Helmholtz equation represents wave phenomena in many physical applications. In Chapter 6, we perform numerical experiments for the model problem defined in Chapter 2. Results are presented in order of varying difficulties of the problem. Also the constant as well as spatially dependent wavenumber problems are used in the experiments. Numerical results shows efficiency of the deflation preconditioner. Convergence is nearly independent of the wavenumber (or frequency), particularly for the case when smaller grid sizes are considered for a fixed wavenumber. The ADEF1 preconditioner continues to show robustness on problems with heterogeneities. A simple and unambiguous performance behavior has been seen while moving from two dimensional problems to three dimensional problems.

A FEM discretization has been discussed in the later part of the chapter. Also based on the FEM discretized Helmholtz equation, possibilities of algebraically constructed deflation vectors has been discussed. Instant results in this regard show an pleasant course to follow in this way.

7.2 Further Implications

The proposal of deflation type preconditioners and its rigorous Fourier analysis implies a number of things. One of the important implication of the deflation preconditioner is its distinct feature of the deflating near-kernel modes. Linear systems suffering the same problem can be worth to apply on. Besides its own influence on spectrum, the deflation preconditioner also allows the shifted Laplace preconditioner to make it more diagonally dominant by increasing the imaginary part of the shift without paying any penalty in the number of Krylov iterations.

7.3 Future Work/Outlook

- Different shifts in different levels of CSLP. Its diagonally dominance at difference levels.
- Re-using deflation vectors once extracted from Krylov method, as usually it is required to solve with many frequencies.
- The coarse grid operator in every variant of deflation preconditioner seems problematic. An intensive care to remove the near-null space in coarse grid operator might be an lucrative option, as it would be relatively easy to deal at coarser levels.
- An important area which offers opportunity for development is parallelization of many sequential numerical algorithms. Parallelization of multigrid customized for any application at hand, is both important as well as non-trivial. With the advent of desktop super-computing facilities, such as offered by Field Programmable Gate Arrays (FPGAs) and Graphical Processing Units (GPUs), parallel computing is more widely accessible than it was before and forms a very interesting and potentially rewarding field of research.

ADEF1 Implementation

A.1 Introduction

The ADEF1 solver software is divided into two parts. Multilevel implementation requires matrices at all levels from finest to coarsest. The data files are constructed in Matlab and subsequently are written into `.dat` files. An important part of the ADEF1 solver software is the part which implements the multilevel preconditioner. This is implemented in PETSc, which calls the `.dat` files on run time.

Structure Main directory `Adef1_Software`, which contains two sub-subdirectories; `ConstructDatFiles` and `PetscSolver`.

The codes in directory `ConstructDatFiles` are used to construct data files, and the directory `PetscSolver` contains subroutines to solve the resulting problem.

A.2 Constructing DATA files

Add path of the directory `ConstructDatFiles` to the Matlab session, also add path of your PETSC `bin/matlab` directory in the Matlab session or adapt the path in program `MainMarmousi.m`.

Run the program `MainMarmousi.m`. It will ask for options in an input dialogue box. The program constructs coefficient matrices according to the provided options. Those options are as following;

Recommended: First make a test run with default options in order to check if it runs smoothly. Subsequently customize with other options.

Frequency f	Give values $f = 1, 10, 20$ or 40
Mesh size	In terms of grid points per wavelength. Choose 10 or 20 .
Real shift	Real shift in complex shifted Laplace preconditioner CSLP, choose whatever you want to use as CSLP.
Imag. shift	Imaginary shift in complex shifted Laplace preconditioner CSLP, choose whatever you want to use as CSLP.
Damping	Damping parameter in the Helmholtz operator.

Output file will be a `.dat` file and will be saved in the directory `../DataFiles/` with customized name `fN1gpWLN2aN3.dat` where $f, gpWL,$ and a are constant where $N1, N2$ and $N3$ will be customized according to the options. For example `f1gpWL10a0.05.dat`, is data set with frequency $f = 1$, number of $gp/wl = 10$ and damping parameter $a = 0.05$.

A.2.1 How to adapt

Reading the `FILENAME.dat` file in PETSc is sensible in terms of order of data types (matrices and vectors) written in the `.dat` file. Adapting the writing (`.dat` files) part of the solver, necessitates adapting the reading part in PETSc. Care of the order should be taken. Same order in writing and reading should be persisted.

The directory `PetscSolver` implements the ADEF1 solver.

A.3 Solving part of the software

In the terminal, go to the target directory; Compile the program as follows

```
> make GMGcycle.o; make MLadef.o; make MainSolver
```

subsequently execute the executable program and provide with a data file with “-f” as follows:

```
> ./MainSolver -f /your/path/to/datafile.dat.
```

In order to restrict the ADEF1 multilevel solver to a two-level solver, runtime option `-coarselevel` can be used. For example to apply a two-level solver, one needs to run

```
> ./MainSolver -f /your/path/to/datafile.dat -coarselevel 1.
```

This execution accepts all the possible runtime PETSc options. All these options can be listed by executing the program with “-help”.

Appendix **B**

Investigation of the behavior of small eigenvalues of ADEF1 with perturbed eigenvectors

B.1 Introduction

During the application of the ADEF1 deflation method combined with the Complex Shifted Laplace Preconditioner (CSLP) to the discretized Helmholtz operator, it appears that the near null eigenvalue components can lead to eigenvalues which are not close to one as it should be when unperturbed eigenvectors are used. The coming forth analysis illustrates the idea that the small eigenvalues (near singular coarse grid operator) combined with a perturbation of the corresponding eigenvectors lead to bad results. These eigenvalues and eigenvectors can also be interpreted as the near null-space of the fine grid operator.

B.2 Definition of the problem

We consider a very simple test problem where the matrix A is only a 2×2 diagonal matrix, given by the following expression:

$$A = \begin{pmatrix} \epsilon & 0 \\ 0 & 1 \end{pmatrix},$$

where $\epsilon \in \mathbb{R}$ has a small value. Since small eigenvalues lead to bad convergence of Krylov methods, the aim of Deflation type methods is to remove or shift these eigenvalues from the resulting iteration matrix.

B.3 ADEF1 deflation method

In order to define ADEF1 we first give a number of expressions which are used as building blocks. Suppose that the matrix Z contains the deflation vectors in its columns. These vectors are approximation vectors of bad eigenvalue components (eigenvectors corresponding to small eigenvalues). Then we define the coarse grid matrix (Galerkin matrix) as follows: $E = Z^T AZ$. The coarse grid correction operator is now defined as

$$Q = Z^T E^{-1} Z.$$

Finally the ADEF1 deflation operator is given by:

$$P_{ADEF1} = \sigma Q + (I - AQ) \text{ for some } \sigma \in \mathbb{R}.$$

In the remainder of this work we take $\sigma = 1$. If A is a symmetric matrix and Z consists of eigenvectors, then the eigenvalues of $P_{ADEF1}A$ are equal the eigenvalues of the eigenvectors not contained in Z , whereas the other eigenvalues are transformed to the value 1.

B.4 Analysis

In this section we compute the matrix $P_{ADEF1}A$ and compute the eigenvalues of this matrix. We start by using the following A , and Z :

$$A = \begin{pmatrix} \epsilon & 0 \\ 0 & 1 \end{pmatrix} \text{ and } Z = \begin{pmatrix} 1 \\ \delta \end{pmatrix}.$$

Now we compute matrix $E = Z^T AZ = \epsilon + \delta^2$. Note that E is a scalar. Furthermore if $\epsilon = -\delta^2$ the value of E is equal to zero. This means that E is only non-singular if $\epsilon \neq -\delta^2$ and $E^{-1} = \frac{1}{\epsilon + \delta^2}$. The matrix $Q = ZE^{-1}Z^T$ can now be calculated:

$$Q = \frac{1}{\epsilon + \delta^2} \begin{pmatrix} 1 & \delta \\ \delta & \delta^2 \end{pmatrix}.$$

The standard deflation operator is given by $P = I - AQ$, which leads to

$$P = \frac{1}{\epsilon + \delta^2} \begin{pmatrix} \delta^2 & -\epsilon\delta \\ -\delta & \epsilon \end{pmatrix}.$$

Including a shift in order to obtain $P_{ADEF1} = Q + P$ we obtain:

$$P_{ADEF1} = \frac{1}{\epsilon + \delta^2} \begin{pmatrix} 1 + \delta^2 & \delta(1 - \epsilon) \\ 0 & \epsilon + \delta^2 \end{pmatrix}.$$

This combined with the matrix A yields:

$$P_{ADEF1}A = \frac{1}{\epsilon + \delta^2} \begin{pmatrix} \epsilon(1 + \delta^2) & \delta(1 - \epsilon) \\ 0 & \epsilon + \delta^2 \end{pmatrix},$$

so the eigenvalues are $\{\lambda_1, \lambda_2\} = \{\frac{\epsilon(1+\delta^2)}{\epsilon+\delta^2}, 1\}$. Note that we can make the following choices for the perturbation δ

$$\delta = 0 \quad \Rightarrow \quad \lambda_1 = 0$$

$$\delta = \sqrt{\epsilon} \quad \Rightarrow \quad \lambda_1 = \frac{1+\epsilon}{2}$$

$$\delta = 1 \quad \Rightarrow \quad \lambda_1 = \frac{2\epsilon}{1+\epsilon} \approx 2\epsilon$$

Finally if $\epsilon \downarrow -\delta^2$ then $\lambda_1 \rightarrow -\infty$ and if $\epsilon \uparrow -\delta^2$ then $\lambda_1 \rightarrow \infty$.

B.5 Numerical experiments

For the given matrix A it is clear that the eigenvalues are $\{\epsilon, 1\}$ and the corresponding eigenvectors are:

$$\begin{pmatrix} 1 \\ 0 \end{pmatrix} \text{ and } \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

We choose only one deflation vector so Z is given by:

$$Z = \begin{pmatrix} 1 \\ \delta \end{pmatrix}.$$

Note that if $\delta = 0$ the deflation vector is equal to the eigenvector corresponding to the small eigenvalue ϵ . If $\delta \neq 0$ it can be seen as a perturbation of the first eigenvector by the second eigenvector. In Figure B.1 we give a number of results of the smallest eigenvalue of $P_{ADEF1}A$. Note that for small values of δ the smallest eigenvalue is close to one, however for larger values of δ the smallest eigenvalue become again small. This shows that eigenvectors corresponding to small eigenvalues are only removed if the deflation vector is a close approximation of the corresponding eigenvector. It appears that the change occurs around the value $\delta = \sqrt{\epsilon}$.

From the analysis it appears that problems can occur when $\epsilon = -\delta^2$. For this reason we take $\epsilon = -10^{-4}(1 - 10^{-4})$ so we expect a problem for $\delta = 10^{-2}$. The result of this is given in Figure B.2. Note that indeed in the vicinity of $\delta = 10^{-2}$ one of the eigenvalues becomes very large. The reason for this is that the matrix E is nearly singular.

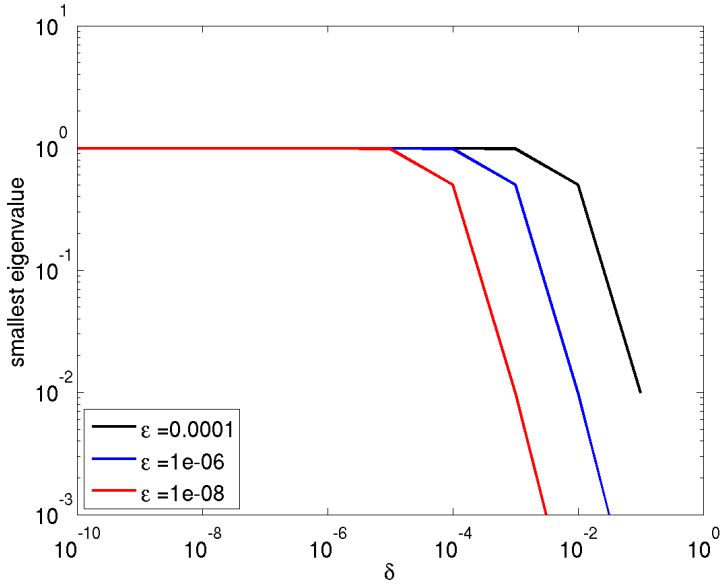


Figure B.1: Smallest eigenvalue of $P_{ADEF1}A$ as function of δ for various values of ϵ

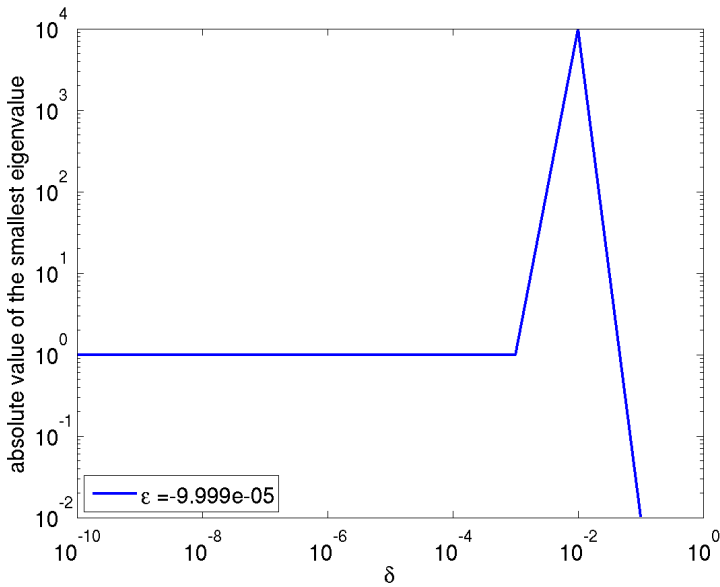


Figure B.2: The absolute value of the smallest eigenvalue of $P_{ADEF1}A$ as function of δ for various values of $\epsilon = -0.999910^{-4}$

B.6 Matlab Code

```
clear all clc

eps = 10^(-4);

for iter = 1 : 10

    delta = 10^(-iter);

    A = [eps 0 ; 0 1];

    [n m] = size(A);

    S = [1 0 ; 0 1];

    Z(:,1) = S(:,1) + delta * S(:,2);

    E = Z'*A*Z;

    Q = Z*inv(E)*Z';

    P = eye(n) - A*Q;

    PADEF = Q + P;

    eig(PADEF*A);

    plotdelta(iter) = delta;

    ploteig(iter) = min(eig(PADEF*A));

end

clf
loglog(plotdelta, ploteig)
xlabel('\delta')
ylabel('smallest eigenvalue')
title(['\epsilon = ' num2str(eps)])
```


Bibliography

- [1] T. Airaksinen, E. Heikkola, A. Pennanen, and J. Toivanen. An algebraic multi-grid based shifted-Laplacian preconditioner for the Helmholtz equation. *Journal of Computational Physics*, 226:1196 – 1210, 2007.
- [2] T. Airaksinen and S. Mönkölä. Comparison between the shifted-Laplacian preconditioning and the controllability methods for computational acoustics. *J. Comput. Appl. Math.*, 234:1796–1802, July 2010.
- [3] T. Airaksinen, A. Pennanen, and J. Toivanen. A damping preconditioner for time-harmonic wave equations in fluid and elastic material. *J. Comput. Phys.*, 228:1466–1479, March 2009.
- [4] P. Amestoy, I. Duff, and J.-Y. L’Excellent. Multifrontal parallel distributed symmetric and unsymmetric solvers. *Computer Methods in Applied Mechanics and Engineering*, 184(2-4):501 – 520, 2000.
- [5] P. R. Amestoy, A. Guermouche, J.-Y. L’Excellent, and S. Pralet. Hybrid scheduling for the parallel solution of linear systems. *Parallel Computing*, 32(2):136 – 156, 2006. *Parallel Matrix Algorithms and Applications (PMAA 04)*.
- [6] I. M. Babuska and S. A. Sauter. Is the pollution effect of the FEM avoidable for the Helmholtz equation considering high wave numbers? *SIAM Review*, 42(3):pp. 451–484, 2000.
- [7] S. Balay, J. Brown, K. Buschelman, V. Eijkhout, W. Gropp, D. Kaushik, M. Knepley, L. C. McInnes, B. Smith, and H. Zhang. *PETSc Users Manual Revision 3.4*, 2013.
- [8] A. Bamberger, R. Glowinski, and Q. Tran. A domain decomposition method for the acoustic wave equation with discontinuous coefficients and grid change. *SIAM Journal on Numerical Analysis*, 34(2):603–639, 1997.

- [9] G. Bartsch and C. Wulf. Adaptive multigrid for Helmholtz problems. *Journal of Computational Acoustics*, 11(03):341–350, 2003.
- [10] M. Baumann and M. Gijzen. Nested Krylov methods for shifted linear systems. Technical Report 14-01, DIAM, TU Delft Netherlands, 2014.
- [11] A. Bayliss, C. Goldstein, and E. Turkel. An iterative method for the Helmholtz equation. *Journal of Computational Physics*, 49:443 – 457, 1983.
- [12] A. Bayliss, C. Goldstein, and E. Turkel. On accuracy conditions for the numerical computation of waves. *Journal of Computational Physics*, 59(3):396 – 404, 1985.
- [13] J. Benamou and B. Desprs. A domain decomposition method for the Helmholtz equation and related optimal control problems. *Journal of Computational Physics*, 136(1):68 – 82, 1997.
- [14] J. Berenger. A perfectly matched layer for the absorption of electromagnetic waves. *Journal of Computational Physics*, 114: 185–200, 1994.
- [15] H. bin Zubair, C. Oosterlee, and R. Wienands. Multigrid for high-dimensional elliptic partial differential equations on non-equidistant grids. *SIAM Journal on Scientific Computing*, 29(4): 1613–1636, 2007.
- [16] A. Bogaerts, E. Neyts, R. Gijbels, and J. van der Mullen. Gas discharge plasmas and their applications. *Spectrochimica Acta Part B: Atomic Spectroscopy*, 57(4):609 – 658, 2002.
- [17] M. Bollhöfer, M. J. Grote, and O. Schenk. Algebraic multilevel preconditioner for the Helmholtz equation in heterogeneous media. *SIAM Journal on Scientific Computing*, 31:3781–3805, 2009.
- [18] A. Brandt. Multi-level adaptive solutions to boundary-value problems. *Mathematics of Computation*, 31: 333–390, 1977.
- [19] A. Brandt and I. Livshits. Wave-ray multigrid method for standing wave equations. *ETNA*, 6: 162–181, 1997.
- [20] A. Brandt, S. F. McCormick, and J. W. Ruge. Algebraic multigrid (AMG) for sparse matrix equations. In D. J. Evans, editor, *Sparsity and Its Applications*. Cambridge University Press, Cambridge, 1984.
- [21] A. Brandt and S. Ta’asan. Multigrid method for nearly singular and slightly indefinite problems. In W. Hackbusch and U. Trottenberg, editors, *Multigrid Methods II*, volume 1228 of *Lecture Notes in Mathematics*, pages 99–121. Springer Berlin Heidelberg, 1986.

- [22] W. Briggs, V. Henson, and S. McCormick. *A Multigrid Tutorial*. SIAM, 2000.
- [23] H. Calandra, S. Gratton, R. Lago, X. Pinel, and X. Vasseur. Two-level preconditioned Krylov subspace methods for the solution of three-dimensional heterogeneous Helmholtz problems in seismics. *Numerical Analysis and Applications*, 5:175–181, 2012.
- [24] H. Calandra, S. Gratton, X. Pinel, and X. Vasseur. An improved two-grid preconditioner for the solution of three-dimensional Helmholtz problems in heterogeneous media. Technical Report TR/PA/12/2, CERFACS, Toulouse, France, 2012.
- [25] M. Carpenter, C. Vuik, P. Lucas, M. van Gijzen, and H. Bijl. A general algorithm for reusing Krylov subspace information. I. unsteady Navier-Stokes. NASA/TM 2010216190, NASA, Langley Research Center, 2010.
- [26] J. F. Claerbout. *Imaging the Earth's interior*. Blackwell Scientific Publications, Oxford, UK, 1985.
- [27] Comsol. *Multiphysics Reference Guide for COMSOL 3.4*, 2011.
- [28] L. Conen, V. Dolean, R. Krause, and F. Nataf. A coarse space for heterogeneous helmholtz problems based on the dirichlet-to-neumann operator. *Journal of Computational and Applied Mathematics*, 271(0):83 – 99, 2014.
- [29] S. Cools, B. Reys, and W. Vanroose. A new level-dependent coarse grid correction scheme for indefinite Helmholtz problems. *Numerical Linear Algebra with Applications*, 21(4):513–533, 2014.
- [30] S. Cools and W. Vanroose. Local fourier analysis of the complex shifted laplacian preconditioner for Helmholtz problems. *Numerical Linear Algebra with Applications*, 20(4):575–597, 2013.
- [31] D. Darnell, R. B. Morgan, and W. Wilcox. Deflated {GMRES} for systems with multiple shifts and multiple right-hand sides. *Linear Algebra and its Applications*, 429(10):2415 – 2434, 2008. Special Issue in honor of Richard S. Varga.
- [32] T. Davis. *Solving Sparse Linear Systems*, chapter 8, pages 135–144. Fundamentals of Algorithms, SIAM, Philadelphia USA, 2006.
- [33] I. Duff, S. Gratton, X. Pinel, and X. Vasseur. Multigrid based preconditioners for the numerical solution of two-dimensional heterogeneous problems in geophysics. *Int. J. Comput. Math.*, 84(8):1167–1181, Aug. 2007.
- [34] I. S. Duff, A. M. Erisman, and J. K. Reid. *Direct Methods for Sparse Matrices*. Oxford University Press, Inc., New York, NY, USA, 1986.

- [35] I. S. Duff and J. Koster. The design and use of algorithms for permuting large entries to the diagonal of sparse matrices. *SIAM J. Matrix Anal. Appl.*, 20:889–901, 1999.
- [36] S. C. Eisenstat, H. C. Elman, and M. H. Schultz. Variational iterative methods for nonsymmetric systems of linear equations. *SIAM Journal on Numerical Analysis*, 20(2):345–357, 1983.
- [37] H. Elman, O. Ernst, and D. O’Leary. A multigrid method enhanced by Krylov subspace iteration for discrete Helmholtz equations. *SIAM J. Sci. Comput.*, 23(4):1291–1315, Apr. 2001.
- [38] B. Engquist and A. Majda. Absorbing boundary conditions for the numerical simulation of waves. *Mathematics of Computation*, 31(139):pp. 629–651, 1977.
- [39] B. Engquist and L. Ying. Sweeping preconditioner for the Helmholtz equation: Hierarchical matrix representation. *Communications on Pure and Applied Mathematics*, 64(5):697–735, 2011.
- [40] Y. Erlangga. *A robust and efficient iterative method for numerical solution of Helmholtz equation*. PhD thesis, DIAM, TU Delft, 2005.
- [41] Y. Erlangga and R. Nabben. Multilevel projection-based nested Krylov iteration for boundary value problems. *SIAM J. Sci. Comput.*, 30:1572–1595, 2008.
- [42] Y. Erlangga and R. Nabben. On a multilevel Krylov method for the Helmholtz equation preconditioned by shifted Laplacian. *Electronic Transactions on Numerical Analysis (ETNA)*, 31:403–424, 2008.
- [43] Y. Erlangga, C. Vuik, and C. Oosterlee. On a class of preconditioners for solving the Helmholtz equation. *Appl. Numer. Math.*, 50(3-4):409–425, 2004.
- [44] Y. A. Erlangga and R. Nabben. Deflation and balancing preconditioners for Krylov subspace methods applied to nonsymmetric matrices. *SIAM J. Matrix Anal. Appl.*, 30:684–699, 2008.
- [45] Y. A. Erlangga and R. Nabben. Algebraic multilevel Krylov methods. *SIAM Journal on Scientific Computing*, 31:3417–3437, 2009.
- [46] Y. A. Erlangga, C. W. Oosterlee, and C. Vuik. A novel multigrid based preconditioner for heterogeneous Helmholtz problems. *SIAM J. Sci. Comput.*, 27:1471–1492, 2006.
- [47] Y. A. Erlangga, C. Vuik, and C. W. Oosterlee. Comparison of multigrid and incomplete LU shifted-Laplace preconditioners for the inhomogeneous Helmholtz equation. *Applied Numerical Mathematics*, 56:648–666, 2006.

- [48] O. Ernst and M. Gander. Why it is difficult to solve Helmholtz problems with classical iterative methods. In I. G. Graham, T. Y. Hou, O. Lakkis, and R. Scheichl, editors, *Numerical Analysis of Multiscale Problems*, volume 83 of *Lecture Notes in Computational Science and Engineering*, pages 325–363. Springer Berlin Heidelberg, 2012.
- [49] O. Ernst and M. J. Gander. *Direct and Inverse Problems in Wave Propagation and Applications*, chapter Multigrid methods for Helmholtz problems: A convergent scheme in 1D using standard components, pages 135–186. Radon Series on Computational and Applied Mathematics 14. De Gruyter, October 2013.
- [50] C. Farhat, P. Avery, R. Tezaur, and J. LI. FETI-DPH: A dual-primal domain decomposition method for acoustic scattering. *Journal of Computational Acoustics*, 13(03):499–524, 2005.
- [51] C. Farhat, A. Macedo, and M. Lesoinne. A two-level domain decomposition method for the iterative solution of high frequency exterior Helmholtz problems. *Numerische Mathematik*, 85(2):283–308, 2000.
- [52] J. Frank and C. Vuik. On the construction of deflation-based preconditioners. *SIAM J. Sci. Comp.*, 23:442–462, 2001.
- [53] J. M. G-Jordan, S. Rojas, M. F-Villegas, and J. E. Castillo. A new second order finite difference conservative scheme. *Divulgaciones Matemáticas*, 13(1):107–122, 2005.
- [54] M. Gander, I. Graham, and E. Spence. How should one choose the shift for the shifted Laplacian to be a good preconditioner for the Helmholtz equation? *Pre-prints, submitted*, 2014.
- [55] M. Gander and F. Nataf. AILU for Helmholtz problems: a new preconditioner based on the analytic parabolic factorization. *Journal of Computational Acoustics*, 9(4):1499–1506, 2001.
- [56] M. Gander and F. Nataf. An incomplete LU preconditioner for problems in acoustics. *Journal of Computational Acoustics*, 13(3):1–22, 2005.
- [57] M. Gander and J. Nataf. AILU: A preconditioner based on the analytic factorization of the elliptic operator. *Numerical Linear Algebra with Applications*, 7(7-8):543–567, 2000.
- [58] M. Gander and H. Zhang. *Domain Decomposition Methods for the Helmholtz Equation: A Numerical Investigation*, volume 91 of *Lecture Notes in Computational Science and Engineering*, pages 215–222. Springer Berlin Heidelberg, 2013.

- [59] M. Gijzen, Y. Erlangga, and C. Vuik. Spectral analysis of the discrete Helmholtz operator preconditioned with a shifted Laplacian. *SIAM Journal on Scientific Computing*, 29:1942–1958, 2007.
- [60] M. Gijzen, G. Sleijpen, and J. Zemke. Flexible and multi-shift induced dimension reduction algorithms for solving large sparse linear systems. Technical Report 11-06, DIAM, TU Delft Netherlands, 2011.
- [61] C. I. Goldstein. Analysis and application of multigrid preconditioners for singularly perturbed boundary value problems. *SIAM Journal on Numerical Analysis*, 26(5):pp. 1090–1123, 1989.
- [62] E. Haber and S. MacLachlan. A fast method for the solution of the Helmholtz equation. *J. Comput. Phys.*, 230(12):4403–4418, June 2011.
- [63] W. Hackbusch. *Multigrid methods and applications*, volume 4 of *Springer series in computational mathematics*. Springer-Verlag Berlin, 1985.
- [64] G. R. Hadley. A complex Jacobi iterative method for the indefinite Helmholtz equation. *Journal of Computational Physics*, 203(1):358 – 370, 2005.
- [65] P. Hénon and Y. Saad. A parallel multilevel ilu factorization based on a hierarchical graph decomposition. *SIAM Journal of Scientific Computing*, 2006.
- [66] B. Hustedt, S. Operto, and J. Virieux. Mixed-grid and staggered-grid finite-difference methods for frequency-domain acoustic wave modelling. *Geophysical Journal International*, 157(3):1269–1296, 2004.
- [67] F. Ihlenburg. *Finite element analysis of acoustic scattering*. Applied mathematical sciences. Springer, New York, 1998.
- [68] F. Ihlenburg and I. Babuska. Finite element solution to the Helmholtz equation with high wave numbers. *Computers and Mathematics with Applications*, 30:9–37, 1995.
- [69] T. Jönsthövel, M. van Gijzen, C. Vuik, C. Kasbergen, and A. Scarpas. Preconditioned conjugate gradient method enhanced by deflation of rigid body modes applied to composite materials. *Computer Modeling in Engineering and Sciences*, 47:97–118, 2009.
- [70] R. Kechroud, A. Soulaimani, Y. Saad, and S. Gowda. Preconditioning techniques for the solution of the Helmholtz equation by the finite element method. *Mathematics and Computers in Simulation*, 65(4-5):303 – 321, 2004. Wave Phenomena in Physics and Engineering: New Models, Algorithms, and Applications.

- [71] S. Kim and S. Kim. Multigrid simulation for high-frequency solutions of the Helmholtz problem in heterogeneous media. *SIAM Journal on Scientific Computing*, 24(2):684–701, 2002.
- [72] A. Kuzmin, M. Luisier, and O. Schenk. Fast methods for computing selected elements of the greens function in massively parallel nanoelectronic device simulations. In F. Wolf, B. Mohr, and D. Mey, editors, *Euro-Par 2013 Parallel Processing*, volume 8097 of *Lecture Notes in Computer Science*, pages 533–544. Springer Berlin Heidelberg, 2013.
- [73] L. Laird and M. Giles. Preconditioned iterative solution of the 2D Helmholtz equation. Technical report, Comp. Lab. Oxford University UK, 2002. NA-02/12.
- [74] I. Livshits. Wave-ray algorithms for helmholtz equations with variable wave numbers: a one-dimensional implementation of two-dimensional ideas. *ArXiv e-prints*, Dec. 2013.
- [75] I. Livshits. A scalable multigrid method for solving indefinite Helmholtz equations with constant wave numbers. *Numerical Linear Algebra with Applications*, 21(2):177–193, 2014.
- [76] I. Livshits and A. Brandt. Accuracy properties of the wave-ray multigrid algorithm for Helmholtz equations. *SIAM J. Sci. Comput.*, 28(4):1228–1251, Apr. 2006.
- [77] M. M. M. Mardoche. Incomplete factorization-based preconditionings for solving the Helmholtz equation. *International Journal for Numerical Methods in Engineering*, 50:1077–1101, 2001.
- [78] MATLAB. version 7.10.0 (r2010a), 2010.
- [79] K. Meerbergen and J. Coyette. Connection and comparison between frequency shift time integration and a spectral transformation preconditioner. *Numerical Linear Algebra with Applications*, 16:1–17, 2009.
- [80] R. B. Morgan. A restarted GMRES method augmented with eigenvectors. *SIAM J. Matrix Anal. Appl.*, 16:1154–1171, 1995.
- [81] R. B. Morgan. GMRES with deflated restarting. *SIAM J. Sci. Comput.*, 24(1):20–37, 2002.
- [82] R. Nabben and C. Vuik. A comparison of deflation and the balancing preconditioner. *SIAM J. Sci. Comput.*, 27:1742–1759, 2006.

- [83] R. Nabben and C. Vuik. A comparison of abstract versions of deflation, balancing and additive coarse grid correction preconditioners. *Numer. Linear Algebra Appl.*, 15:355–372, 2008.
- [84] R. A. Nicolaides. Deflation of conjugate gradients with applications to boundary value problems. *SIAM J. Numer. Anal.*, 24(2):355–365, 1987.
- [85] L. N. Olson and J. B. Schroder. Smoothed aggregation for helmholtz problems. *Numerical Linear Algebra with Applications*, 17(2-3):361–386, 2010.
- [86] S. Operto, J. Virieux, P. Amestoy, J. L’Excellent, L. Giraud, and H. Ali. 3D finite-difference frequency-domain modeling of visco-acoustic wave propagation using a massively parallel direct solver: A feasibility study. *Geophysics*, 72(5):SM195–SM211, 2007.
- [87] D. Osei-Kuffuor and Y. Saad. Preconditioning Helmholtz linear systems. *Appl. Numer. Math.*, 60:420–431, April 2010.
- [88] J. Pestana and A. Wathen. On choice of preconditioner for minimal residual methods for nonsymmetric matrices. report na10-07., Technical report, Oxford University Mathematical Institute, Numerical Analysis group, 2010.
- [89] R. Plessix. A Helmholtz iterative solver for 3D seismic-imaging problems. *Geophysics*, 72:SM185–SM194, 2007.
- [90] R. Plessix. Three-dimensional frequency-domain full-waveform inversion with an iterative solver. *Geophysics*, 74(6):149–157, 2009.
- [91] R. E. Plessix and W. A. Mulder. Separation-of-variables as a preconditioner for an iterative Helmholtz solver. *Appl. Numer. Math.*, 44:385–400, 2003.
- [92] L. G. Ramos. Fourier analysis of the preconditioned helmholtz equation for scattering problems. Master’s thesis, Institute of Mathematics, Berlin University of Technology, Berlin Germany, 2012.
- [93] B. Reps, W. Vanroose, and H. B. Zubair. On the indefinite Helmholtz equation: Complex stretched absorbing boundary layers, iterative analysis, and preconditioning. *J. Comput. Phys.*, 229:8384–8405, November 2010.
- [94] C. Riyanti, A. Kononov, Y. Erlangga, C. Vuik, C. Oosterlee, R. Plessix, and W. Mulder. A parallel multigrid-based preconditioner for the 3D heterogeneous high-frequency Helmholtz equation. *Journal of Computational Physics*, 224:431–448, 2007.
- [95] Y. Saad. A flexible inner-outer preconditioned GMRES algorithm. *SIAM J. Sci. Comput.*, 14(2):461–469, Mar. 1993.

- [96] Y. Saad. ILUT: A dual threshold incomplete LU factorization. *Numerical Linear Algebra with Applications*, 1(4):387–402, 1994.
- [97] Y. Saad. *Iterative Methods for Linear system*. PWS Publishing Company, 1996.
- [98] Y. Saad. *Iterative Methods for Sparse Linear Systems*. SIAM, Philadelphia, PA, USA, 2003. Second edition.
- [99] Y. Saad and M. Schultz. GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 7(3):856–869, 1986.
- [100] G. Samaey and W. Vanroose. An analysis of equivalent operator preconditioning for equation-free Newton-Krylov methods. *SIAM J. Numer. Anal.*, 48:633–658, 2010.
- [101] A. Sheikh, D. Lahaye, and C. Vuik. On the convergence of shifted Laplace preconditioner combined with multilevel deflation. *Numerical Linear Algebra with Applications*, 20:645–662, 2013.
- [102] A. Sheikh, C. Vuik, and D. Lahaye. Fast iterative solution methods for the helmholtz equation. Technical Report 09-11, DIAM, TU Delft, 2009.
- [103] A. Sheikh, C. Vuik, and D. Lahaye. A scalable Helmholtz solver combining the shifted Laplace preconditioner with multigrid deflation. Technical Report 11-01, DIAM, TU Delft Netherlands, 2011.
- [104] V. Simoncini, V. Simoncini, E. Gallopoulos, and E. Gallopoulos. An iterative method for nonsymmetric systems with multiple right-hand sides. *SIAM J. Sci. Comput.*, 16:917–933, 1995.
- [105] I. Singer and E. Turkel. High-order finite difference methods for the Helmholtz equation. *Computer Methods in Applied Mechanics and Engineering*, 163(1):343–358, 1998.
- [106] P. Sonneveld and M. B. 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(2):1035–1062, 2008.
- [107] K. Stüben and U. Trottenberg. *Multigrid Methods: fundamental algorithms, model problem analysis and applications*. Springer Berlin, 1982.
- [108] J. Tang, S. MacLachlan, R. Nabben, and C. Vuik. A comparison of two-level preconditioners based on multigrid and deflation. *SIAM. J. Matrix Anal. and Appl.*, 31:1715–1739, 2010.

- [109] J. Tang, R. Nabben, C. Vuik, and Y. Erlangga. Comparison of two-level preconditioners derived from deflation, domain decomposition and multigrid methods. *Journal of Scientific Computing*, 39:340–370, 2009.
- [110] J. Tang and C. Vuik. On deflation and symmetric positive semi-definite matrices. *J. Comput. Appl. Math.*, 206(2):603–614, 2007.
- [111] J. M. Tang. *Two Level Preconditioned conjugate gradient methods with applications to bubbly flow problems*. PhD thesis, DIAM, TU Delft, 2008.
- [112] J. M. Tang, S. P. MacLachlan, R. Nabben, and C. Vuik. Theoretical comparison of two-level preconditioners based on multigrid and deflation. DIAM Report 08-05, Delft University of Technology, Delft, 2006.
- [113] U. Trottenberg, C. Oosterlee, and A. Schüller. *Multigrid*. Academic Press, London, 2001.
- [114] N. Umetani, S. MacLachlan, and C. Oosterlee. A multigrid-based shifted Laplacian preconditioner for a fourth-order Helmholtz discretization. *Numerical Linear Algebra with Applications*, 16:603–626, 2009.
- [115] H. van der Vorst. Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 13(2):631–644, 1992.
- [116] R. Versteeg and G. Grau. The Marmousi experience. In *Proc. EAGE workshop on Practical Aspects of Seismic Data Inversion (Copenhagen, 1990)*, Eur. Assoc. Explor. Geophysicists, Zeist, 1991.
- [117] A. Vion and C. Geuzaine. Double sweep preconditioner for optimized schwarz methods applied to the Helmholtz problem. *Journal of Computational Physics*, 266(0):171 – 190, 2014.
- [118] P. Wesseling. *An Introduction to Multigrid Methods*. Pure and Applied Mathematics. John Wiley & Sons, 1992.
- [119] X. Pinel. *A perturbed two-level preconditioner for the solution of three-dimensional heterogeneous Helmholtz problems with applications to geophysics*. PhD thesis, CERFACS, Toulouse France, 2010.
- [120] J. Zhu, X. Ping, R. Chen, Z. Fan, and D. Ding. An incomplete factorization preconditioner based on shifted Laplace operators for FEM analysis of microwave structures. *Microwave and Optical Technology Letters*, 52:1036–1042, 2010.

Curriculum Vitae



Abdul Hanan Sheikh was born on 7th of June, 1984 in Shikarpur, Sindh, Pakistan. He received his primary and secondary education at Shikarpur and subsequently at Gambat. He did his B.Sc. 2004 from Shah Abdul Latif University, Khairpur, Pakistan. He attended Quaid-i-Azam University Islamabad Pakistan for his M.Sc. degree in Mathematics 2007. He continued education and enrolled in MPhil and attended the intensive courses of MPhil program. January 2008, he joined Mathematics department of NED University of engineering and Technology, Karachi Pakistan as Lecturer. Since 2009, he has been in the Netherlands, pursuing his PhD in Scientific Computing in the Numerical Analysis group, at the Delft University of Technology. He was supervised by Prof. C. Vuik and Dr. D Lahaye. Lately, he has collaborated with Prof. R. Nabben (TU Berlin, Germany). This thesis is result of his research work.

He has presented his research output in seminars and conferences related to applied scientific computing. The prominent activities include his participation and presentation in EMG 2014, 11th Copper Mountain conference on Iterative Methods, 2010 Colorado, USA, 15th Copper Mountain conference on Multigrid Methods, 2011 Colorado USA and Sparse Days, 2011 Cerfacs, Toulouse, France. Currently, he is actively involved in researching iterative solution methods for the indefinite Helmholtz equation that arises in many applied problems.

His leisure activities include traveling and discussions on various topics including science, history, life, etc. As a hobby he likes to read books.

Propositions

belonging to the dissertation

Development of the Helmholtz Solver based on a Shifted Laplace Preconditioner and a Multigrid Deflation technique

by Abdul Hanan Sheikh

1. Ritz value analysis does not always give a good prediction of the convergence of Krylov methods.
2. Some way to extract the null-space, in an efficient manner, would give a better Helmholtz preconditioner than the ones that are currently available (H. Bin Zubair, TU Delft 2010).
3. Hybrid methods, a combination of direct and iterative methods, are fairly comparable with direct methods with respect to memory consumption.
4. Isolating various numerical solution techniques causes great loss to scientific computing research.
5. In a multi-factor analysis, understanding results are as difficult as presenting results.
6. In numerical solution methods, finding the sources of error is more difficult than proposing/developing a new solution method.
7. Parameter tuning in any multilevel method (preconditioner) is as essential as the method itself.
8. Despite all the scientific and technical prowess of mankind, the modern man is as far from humanity- and as prone and vulnerable to the call of the wild, as the cave man.
9. Many religions have many common decrees which are almost inevitable in a civilized society.
10. There should be subsidy on food for students, like travel and accomodation.

These propositions are considered defendable and as such have been approved by the supervisor Prof.dr.ir C. Vuik.

Stellingen

behorende bij het proefschrift

Development of the Helmholtz Solver

based on

a Shifted Laplace Preconditioner and a Multigrid Deflation technique

door Abdul Hanan Sheikh

1. Analyse van Ritzwaardes geeft niet altijd een goede voorspelling van de convergentie van Krylovmethodes.
2. Als de nulruimte op een efficiënte manier bepaald kan worden, dan zou dit leiden tot een betere Helmholtzpreconditioner dan op dit moment beschikbaar is (H. Bin Zubair, TU Delft 2010).
3. Hybride methodes, combinaties van directe en iteratieve methodes, zijn wat betreft geheugengebruik vergelijkbaar met directe methodes.
4. Het isoleren van verschillende numerieke oplossingsmethodes beperkt sterk het onderzoek in Scientific Computing.
5. Het begrijpen van de resultaten van een multicomponentsanalyse is net zo ingewikkeld als het presenteren ervan.
6. Bij numerieke oplossingsmethodes is het vinden van de oorzaak van een fout moeilijker dan het voorstellen/ontwikkelen van een nieuwe oplossingsmethode.
7. Het afstellen van een parameter in een willekeurige multilevelmethode (preconditioner) is even essentieel als de methode zelf.
8. Ondanks alle wetenschappelijke en technische kunde van de moderne man, staat hij net zo ver van de mensheid- en is even gevoelig en kwetsbaar voor de roep van de wildernis, als een holbewoner.
9. Veel religies hebben vele gemeenschappelijke decreten die nagenoeg onvermijdelijk zijn in een geciviliseerde maatschappij.
10. Voedsel zou voor studenten gesubsidieerd moeten zijn, net als vervoer en onderdak.

Deze stellingen worden verdedigbaar geacht en zijn als zodanig goedgekeurd door de promotor Prof.dr.ir. C.Vuik.