



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

공학박사학위논문

An improved shifted Laplace preconditioner  
for solving the Helmholtz equation

헬름홀츠 방정식을 풀기 위한 개선된 shifted Laplace  
preconditioner

2021 년 02 월

서울대학교 대학원  
협동과정 계산과학전공  
박 윤 서



An improved shifted Laplace preconditioner for  
solving the Helmholtz equation

지도교수 신 창 수

이 논문을 공학박사 학위논문으로 제출함

2020 년 12 월

서울대학교 대학원

협동과정 계산과학전공

박 윤 서

박윤서의 공학박사 학위논문을 인준함

2020 년 12 월

위 원 장                      신 동 우

\_\_\_\_\_ (인) 

부위원장                      신 창 수

\_\_\_\_\_ (인) 

위    원                      민 동 주

\_\_\_\_\_ (인) 

위    원                      고 승 원

\_\_\_\_\_ (인) 

위    원                      이 재 준

\_\_\_\_\_ (인) 



# Abstract

헬름홀츠 방정식은 주파수 성분의 파동장을 지배하는 편미분 방정식(PDE)이다. 이는 타원형 PDE이기 때문에, 소스 정보를 포함한 주어진 우변에 대해 선형 시스템, 즉 임피던스 매트릭스를 계산함으로써 해를 얻을 수 있다. 헬름홀츠 방정식의 상호상관성(reciprocity) 때문에, 이산화된 임피던스 행렬은 복소 대칭 행렬로 자연스럽게 표현될 수 있으며, 이를 통해 Cholesky 분해를 적용하여 메모리 요구량과 산술 연산의 수를 절반으로 줄일 수 있다. 하지만 PML 경계조건을 사용하면서 행렬의 대칭 특성을 보존하는 방법은 널리 사용되지 않았다. 따라서 본 연구에서는 PML 경계조건 층을 포함한 전체 공간 노드에 대한 대칭 행렬로서 헬름홀츠 방정식을 이산화하는 방법을 제안한다. 또한 파장당 격자 수가 4보다 큰 범위 내에서 0.5% 미만의 분산 오차를 만족하는 최적화된 이산화 계수를 제안한다. 선형 시스템은 직접법을 사용하여 2차원 혹은 작은 3차원 문제를 풀 수 있지만, 복잡도를 고려하면 일반적인 3차원 문제를 풀기에는 시간이 소요량, 행렬의 분해시 저장공간 요구량 그리고 임피던스 행렬을 계산하는데 극히 높은 비용이 필요하게 된다. 이 경우, 반복법, 특히, conjugate gradient(CG), GMRES 등과 같은 크릴로브 부분공간법이 효과적으로 사용될 수 있다. 본 연구에서는, 여러 크릴로브 부분공간법들을 대칭 행렬에 적용해 실험적으로 어떤 방법이 임피던스 행렬을 계산하는데 적합한지 확인하였다. 이를 통해, 다양한 kh 조건에 대해 conjugate residual(CR)이 가장 빠르게 해를 제공하는 것을 관찰하였다. 수렴 속도의 개선하기 위해, preconditioner 행렬의 복소 시프트( $\epsilon$ )를 조절하여 최적의 고유값 군집화를 보여주는 Shifted Laplace preconditioner가 사용되었다. 일반적으로 이러한

preconditioner를 적용하기 위해 불완전 분해가 주로 사용된다. 본 연구에서는, 분해된 행렬의  $p$ 개의 가장 큰 값만 남기는 ICT( $p$ ), 불완전 Cholesky 분해를 사용하는 것을 제안한다. 주어진  $kh$  조건에 대해 최적의  $\epsilon$ 과  $p$ 를 제안하고, 이를 인공합성 속도모델을 사용한 수치 실험을 통해 검증한다. 여기서 직접법과 비교하여 파동장의 해를 구하는데 적은 시간과 무시할 수 있는 오차를 보여주었다. 특히,  $169 \times 169 \times 50$  크기의 3차 문제의 경우, 고도로 최적화된 프론탈 솔버인 UMFPACK의 연산 시간의 약 5%인 것으로 나타났다.

**Keywords:** Helmholtz equation, Krylov subspace, Preconditioner, shifted Laplace

**Student Number:** 2014-21329

# Contents

<b>Abstract</b>	<b>i</b>
<b>Chapter 1 Introduction</b>	<b>1</b>
1.1 Background and Importance . . . . .	2
1.1.1 Helmholtz equation . . . . .	2
1.1.2 Krylov subspace methods . . . . .	3
1.1.3 Preconditioners . . . . .	4
1.2 Objective . . . . .	14
1.3 Organization of the Thesis . . . . .	14
<b>Chapter 2 The Helmholtz equation</b>	<b>15</b>
2.1 Boundary conditions . . . . .	16
2.2 Discretization . . . . .	20
<b>Chapter 3 Krylov subspace method and preconditioner</b>	<b>30</b>
3.1 Krylov subspace methods . . . . .	32
3.1.1 Convergence . . . . .	42



3.2	Preconditioner . . . . .	46
3.2.1	Shifted Laplace preconditioner . . . . .	48
3.2.2	Incomplete factorization . . . . .	61
3.2.3	Fill-in comparison between ICT(p) and IC(k) . . . . .	67
3.2.4	Spectrum of preconditioned system . . . . .	69
3.3	Comparison between IC( $k$ ) and ICT(p) with Shifted Laplace preconditioner . . . . .	72
<b>Chapter 4 Numerical Examples</b>		<b>83</b>
4.1	Parameter selection of preconditioner . . . . .	83
4.1.1	Numerical domain size dependency . . . . .	83
4.1.2	kh dependency . . . . .	94
4.1.3	Computational time . . . . .	98
4.2	Heterogeneous model . . . . .	106
4.2.1	2D : Marmousi . . . . .	106
4.2.2	2D : Pluto . . . . .	112
4.2.3	3D : SEG/EAGE 3D . . . . .	118
<b>Chapter 5 Conclusion</b>		<b>124</b>
<b>Bibliography</b>		<b>127</b>
<b>초록</b>		<b>138</b>

# List of Figures

Figure 2.1	정규화된 위상 속도 . . . . .	29
Figure 3.1	kh = 0.314159(G=20) 일때 잔차 감소 경향에 따른 GM-RES(m), BiCGSTAB, SQMR, CR의 행렬-벡터 곱 . . . . .	43
Figure 3.2	kh = 0.408407(G=15) 일때 잔차 감소 경향에 따른 GM-RES(m), BiCGSTAB, SQMR, CR의 행렬-벡터 곱 . . . . .	44
Figure 3.3	kh = 0.628319(G=10) 일때 잔차 감소 경향에 따른 GM-RES(m), BiCGSTAB, SQMR, CR의 행렬-벡터 곱 . . . . .	44
Figure 3.4	kh = 1.25664(G=5) 일때 잔차 감소 경향에 따른 GM-RES(m), BiCGSTAB, SQMR, CR의 행렬-벡터 곱 . . . . .	45
Figure 3.5	원 시스템 A의 고유값 분포 . . . . .	59
Figure 3.6	$\epsilon = 0.1$ 일 때 $M_{SL}^{-1}$ 을 직접법으로 계산하였을 때의 고유값 분포 . . . . .	59
Figure 3.7	$\epsilon = 10$ 일 때 $M_{SL}^{-1}$ 을 직접법으로 계산하였을 때의 고유값 분포 . . . . .	60

Figure 3.8	$\epsilon = 30$ 일 때 $M_{SL}^{-1}$ 을 직접법으로 계산하였을 때의 고유값 분포	60
Figure 3.9	원 시스템 A의 고유값 분포	70
Figure 3.10	$\epsilon = 0.1$ 일 때 $M_{SL}^{-1}$ 을 ICT(p)로 계산하였을 때의 고유값 분포	70
Figure 3.11	$\epsilon = 0.1$ 일 때 $M_{SL}^{-1}$ 을 IC(k)으로 계산하였을 때의 고유값 분포	71
Figure 4.1	2차원 $100 \times 100$ 일 때, p와 $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간	86
Figure 4.2	2차원 $200 \times 200$ 일 때, p와 $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간	86
Figure 4.3	2차원 $400 \times 400$ 일 때, p와 $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간	87
Figure 4.4	2차원 $600 \times 600$ 일 때, p와 $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간	87
Figure 4.5	2차원 $800 \times 800$ 일 때, p와 $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간	88
Figure 4.6	2차원 $1000 \times 1000$ 일 때, p와 $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간	88
Figure 4.7	3차원 $100 \times 100 \times 100$ 일 때, p와 $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간	89
Figure 4.8	3차원 $120 \times 120 \times 120$ 일 때, p와 $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간	89

Figure 4.9	3차원 $140 \times 140 \times 140$ 일 때, $p$ 와 $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간 . . . . .	90
Figure 4.10	3차원 $160 \times 160 \times 160$ 일 때, $p$ 와 $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간 . . . . .	90
Figure 4.11	3차원 $180 \times 180 \times 180$ 일 때, $p$ 와 $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간 . . . . .	91
Figure 4.12	3차원 $200 \times 200 \times 200$ 일 때, $p$ 와 $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간 . . . . .	91
Figure 4.13	2차원일 때, 수치영역 크기에 따른 $p$ 와 $\epsilon$ 과 평균값 . . . . .	92
Figure 4.14	3차원일 때, 수치영역 크기에 따른 $p$ 와 $\epsilon$ 과 평균값 . . . . .	92
Figure 4.15	$100 \times 100$ 일 때, 각 $kh$ 에서 $p$ 에 따른 수렴까지 걸리는 반복 횟수와 총 시간 . . . . .	95
Figure 4.16	$200 \times 200$ 일 때, 각 $kh$ 에서 $p$ 에 따른 수렴까지 걸리는 반복 횟수와 총 시간 . . . . .	95
Figure 4.17	$400 \times 400$ 일 때, 각 $kh$ 에서 $p$ 에 따른 수렴까지 걸리는 반복 횟수와 총 시간 . . . . .	96
Figure 4.18	$600 \times 600$ 일 때, 각 $kh$ 에서 $p$ 에 따른 수렴까지 걸리는 반복 횟수와 총 시간 . . . . .	96
Figure 4.19	$800 \times 800$ 일 때, 각 $kh$ 에서 $p$ 에 따른 수렴까지 걸리는 반복 횟수와 총 시간 . . . . .	97
Figure 4.20	$1000 \times 1000$ 일 때, 각 $kh$ 에서 $p$ 에 따른 수렴까지 걸리는 반복 횟수와 총 시간 . . . . .	97

Figure 4.21	2차원 $100 \times 100$ 일 때, $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽) . . . . .	98
Figure 4.22	2차원 $200 \times 200$ 일 때, $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽) . . . . .	99
Figure 4.23	2차원 $400 \times 400$ 일 때, $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽) . . . . .	99
Figure 4.24	2차원 $600 \times 600$ 일 때, $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽) . . . . .	100
Figure 4.25	2차원 $800 \times 800$ 일 때, $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽) . . . . .	100
Figure 4.26	2차원 $1000 \times 1000$ 일 때, $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽) . . . . .	101
Figure 4.27	3차원 $100 \times 100 \times 100$ 일 때, $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽) . . . . .	101
Figure 4.28	3차원 $120 \times 120 \times 120$ 일 때, $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽) . . . . .	102

Figure 4.29	3차원 $140 \times 140 \times 140$ 일 때, $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽) . . . . .	102
Figure 4.30	3차원 $160 \times 160 \times 160$ 일 때, $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽) . . . . .	103
Figure 4.31	3차원 $180 \times 180 \times 180$ 일 때, $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽) . . . . .	103
Figure 4.32	3차원 $200 \times 200 \times 200$ 일 때, $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽) . . . . .	104
Figure 4.33	3차원 실험 결과와 직접법과 이상적인 SL preconditioner 적용된 반복법의 시간 복잡도 비교 . . . . .	105
Figure 4.34	3차원 실험 결과와 직접법과 이상적인 SL preconditioner 적용된 반복법의 저장공간 복잡도 비교 . . . . .	105
Figure 4.35	The Marmousi 속도 모델 . . . . .	106
Figure 4.36	Marmousi 모델의 $kh=0.418879$ 일때 송신원 깊이에서의 직접법의 해와 반복법의 해 비교 . . . . .	108
Figure 4.37	Marmousi 모델의 $kh=0.837758$ 일때 송신원 깊이에서의 직접법의 해와 반복법의 해 비교 . . . . .	108
Figure 4.38	Marmousi 모델의 $kh=1.25664$ 일때 송신원 깊이에서의 직 접법의 해와 반복법의 해 비교 . . . . .	109
Figure 4.39	직접법으로 구한 Marmousi 모델의 seismogram . . . . .	110

Figure 4.40	반복법으로 구한 Marmousi 모델의 seismogram . . . . .	110
Figure 4.41	직접법과 반복법으로 구한 Marmousi 모델의 seismogram 의 차이 . . . . .	111
Figure 4.42	Pluto 속도 모델 . . . . .	112
Figure 4.43	Pluto 모델의 $kh=0.418879$ 일때 송신원 깊이에서의 직접 법의 해와 반복법의 해 비교 . . . . .	114
Figure 4.44	Pluto 모델의 $kh=0.837758$ 일때 송신원 깊이에서의 직접 법의 해와 반복법의 해 비교 . . . . .	114
Figure 4.45	Pluto 모델의 $kh=1.25664$ 일때 송신원 깊이에서의 직접법 의 해와 반복법의 해 비교 . . . . .	115
Figure 4.46	직접법으로 구한 Pluto 모델의 seismogram . . . . .	116
Figure 4.47	반복법으로 구한 Pluto 모델의 seismogram . . . . .	116
Figure 4.48	직접법과 반복법으로 구한 Pluto 모델의 seismogram의 차이	117
Figure 4.49	SEG/EAGE 3D 속도 모델의 단면 . . . . .	118
Figure 4.50	SEG/EAGE 3D 모델의 $kh=0.837758$ 일때 송신원 깊이에 서의 직접법의 해와 반복법의 해 비교 . . . . .	120
Figure 4.51	SEG/EAGE 3D 모델의 $kh=1.67552$ 일때 송신원 깊이에 서의 직접법의 해와 반복법의 해 비교 . . . . .	120
Figure 4.52	SEG/EAGE 3D 모델의 $kh=2.51327$ 일때 송신원 깊이에 서의 직접법의 해와 반복법의 해 비교 . . . . .	121
Figure 4.53	직접법으로 구한 SEG/EAGE 3D 모델의 seismogram . . .	122
Figure 4.54	반복법으로 구한 SEG/EAGE 3D 모델의 seismogram . . .	122
Figure 4.55	직접법과 반복법으로 구한 SEG/EAGE 3D 모델의 seis- mogram의 차이 . . . . .	123

# List of Tables

Table 3.1	kh(G)에 따른 크릴로브 부분공간법들의 수렴까지의 반복횟수	45
Table 3.2	행렬 A와 $M_{SL}^{-1}A$ 의 시프트의 값에 따른 조건수 . . . . .	58
Table 3.3	2차원 9점 콤팩트 스텐실 이산화일 때 ICT(p)와 IC(k)의 채움 갯수(N은 시스템의 크기) . . . . .	67
Table 3.4	3차원 27점 콤팩트 스텐실 이산화일 때 ICT(p)와 IC(k)의 채움 갯수(N은 시스템의 크기) . . . . .	68
Table 3.5	행렬 A와 $AM_{SL}^{-1}$ 의 조건수 . . . . .	69
Table 3.6	IC(k)와 ICT(p)의 비교실험 조건 . . . . .	72
Table 3.7	IC(k), kh=0.314159(G=20)일때 수렴까지 소요되는 반복 횟수 . . . . .	73
Table 3.8	IC(k), kh=0.408407(G=15)일때 수렴까지 소요되는 반복 횟수 . . . . .	74
Table 3.9	IC(k), kh=0.628319(G=10)일때 수렴까지 소요되는 반복 횟수 . . . . .	75
Table 3.10	IC(k), kh=1.25664(G=5)일때 수렴까지 소요되는 반복 횟수	76



Table 3.11	ICT(p), kh=0.314159(G=20)일때 수렴까지 소요되는 반복 횟수 . . . . .	77
Table 3.12	ICT(p), kh=0.408407(G=15)일때 수렴까지 소요되는 반복 횟수 . . . . .	78
Table 3.13	ICT(p), kh=0.628319(G=10)일때 수렴까지 소요되는 반복 횟수 . . . . .	79
Table 3.14	ICT(p), kh=1.25664(G=5)일때 수렴까지 소요되는 반복 횟수 . . . . .	80
Table 3.15	$4km^2$ 물리 영역에서 ICT(p)와 IC(k)의 kh 따른 최소 시간 결과 . . . . .	81
Table 4.1	kh = 1.25664, 2차원 문제에서 수치영역 별 최적의 p와 $\epsilon$ , 이때의 반복 횟수와 소요 시간 . . . . .	93
Table 4.2	kh = 1.25664, 3차원 문제에서 수치영역 별 최적의 p와 $\epsilon$ , 이때의 반복 횟수와 소요 시간 . . . . .	93
Table 4.3	Marmousi 모델 각 kh별 반복 횟수와 소요 시간 . . . . .	107
Table 4.4	Pluto 모델 각 kh별 반복 횟수와 소요 시간 . . . . .	113
Table 4.5	SEG/EAGE 3D 모델 각 kh별 반복 횟수와 소요 시간 . . . . .	119

# Chapter 1

## Introduction

파동 전파의 효율적인 계산 모델링은 많은 공학 문제에서 중요하게 다뤄진다. 시간 고조파 산란 현상(time harmonic wave scattering)은 음향, 전자기, 원자 분광학, 레이더 및 음파 탐지 기술, 지진학과 같은 다양한 과학 분야에서 응용되며, 최근 의료 영상에서도 주목을 받고 있다. 특히 에너지 산업의 석유 및 가스 탐사 분야에서는 passive 및 active source로 인해 발생된 파동이 관측된 자료를 이용하여 지하의 속도 구조를 재구성하는 문제를 주로 다루는데, 이를 역 문제(inverse problem)라고 하며, 파동 전파 시뮬레이션은 역 문제에서 중요한 부분을 차지한다. 본 분야에서 요구하는 기술의 성숙도와 문제의 스케일이 빠르게 높아짐에 따라 대규모 파동전파 시뮬레이션을 빠르고 정확하게 수행할 수 있는 알고리즘들이 제안되어 왔으며, 본 연구도 그 일환으로 수행되었다.

## 1.1 Background and Importance

### 1.1.1 Helmholtz equation

시간 영역에서 정의된 파동방정식은 쌍곡선 형식의 선형 편미분방정식(partial differential equation; PDE)이며, 초기조건으로부터 시작되는 특성 선분(characteristic line)을 따라 해가 시간-공간-의존적으로 정의된다. 이에 반해 헬름홀츠 방정식은 파동방정식을 푸리에 영역으로 변환된 형태로 시간 의존적인 성격을 잃고 타원 형식 PDE의 특성을 갖게 되어 경계조건 문제로 전환된다. 경계에서의 함수 값 및 수직방향 기울기벡터 값을 부여하는 Dirichlet, Neumann 경계조건이 일반적으로 적용되는데, 이에 대하여 제차(homogeneous) PDE로 표현된 헬름홀츠 방정식은 라플라스 연산자에 대한 고유값(eigenvalue) 문제로 환원되며, 이 때 고유값은 파수 (wavenumber)이다. 이에 반해 비제차(inhomogeneous) PDE의 경우 파동장이 정의되는 컨트롤 볼륨에 작용하는 물리적 현상들, 예컨대 외력, 질량 유입, 등이 우변에 음원(source)으로 반영되어 있으며, 이 때 문제가 잘 정의(well-posed)되기 위한 조건으로 음원이 무한대 공간으로 방사가 되는 것을 의미하는 Sommerfeld 경계 조건[29]이 적용된다. 이를 실용적으로 적용하기 위해 paraxial wave equation 기반 경계조건[21, 29, 61], 기계적 감쇄항을 고려한 sponge boundary condition[15], 혹은 모든 주파수 및 반사각에 대해 완벽하게 반사파를 감쇄하여 무한 영역을 모사하는 perfectly matched layers(PML)[7]가 제안되었다.

헬름홀츠 방정식의 수치해를 구하기 위해서는 편미분 방정식(PDE)은 이산화 과정을 거쳐 대 규모의 희소 선형 시스템으로 구성 되어야한다. 이 선형 시스템의 해를 계산하는 것은 다양한 과학 및 공학 문제에서 중요한 요소이며 시간 복잡도가 높은 과정이다. 이산화 과정에는 일반적으로 유한차분법 (Finite difference method) 혹은 유한요소법 (Finite Element method)을 이용되며, 미분을 위하여

절점 주변의 파동장 값을 이용하기 때문에 매우 희소한 행렬로 구성된다. 결과적으로 구성된 계수 행렬은 임피던스 행렬로 불리며[56], 등방성 매질에 대한 헬름홀츠 연산자는 상호상관성(reciprocity)을 만족하므로, 임피던스 행렬이 복소-대칭행렬로 구성되며, 경계조건 및 계산 절점에 부여되는 매질 속성을 계산하는 방식에 따라 종종 복소-비에르미트로 표현되기도 한다[26]. 구성된 선형 시스템의 해를 구하기 위해 우선 고려되는 방법은 직접법(Direct method)으로, 이는 용이하게 역행렬을 구성할 수 있도록 임피던스 행렬을 하, 상 삼각행렬로 분해하는 과정을 거친다. 본 방법은 수치해의 신뢰성이 주요 관심사인 산업 응용 소프트웨어에서 많이 활용되는 반면, 상당한 계산 자원을 필요로 하는 경향이 있다[54]. 따라서 직접법은 3차원 문제의 경우 구성되는 선형 시스템에서 연산 횟수 및 저장공간 필요량 측면에서 적용성이 떨어진다. 이러한 문제의 경우, 반복법(Iterative method)이 유일한 선택지이며 일반적으로 크릴로브 부분공간법이 주로 이용되는데, 이는 반복 계산 횟수가 시스템의 크기로 제한되는 성질로 인해 빠른 수렴을 기대할 수 있기 때문이다. 하지만 round-off error의 누적 및 전파로 인하여 훨씬 많은 반복 계산을 요구한다.

### 1.1.2 Krylov subspace methods

비록 헬름홀츠 방정식이 타원 타입의 PDE지만, 이산화를 통해 얻어진 임피던스 행렬의 성질은 같은 타입의 PDE인 포와송 방정식에 대한 이산화 행렬에 비해 조건수가 낮다. 특히 파수( $k$ )와 공간 격자 간격( $h$ )의 곱( $kh$ )으로 표현되는 무차원 수가 커질수록 임피던스 행렬의 조건수가 높아지는 경향이 있다. 이러한 성질로 인해 일반적으로 고주파 영역에 대한 파동장을 반복법을 통해 구하는 것은 어려운 문제로 여겨진다[35].

그럼에도 불구하고 반복법은 직접법에 비해 저장공간의 필요량이 적고 연산 횟수

가 적다는 점에서 큰 강점을 지니며, 행렬을 대각성분과 비 대각성분으로 분해하여 얻어지는 관계식을 반복 연산으로 변환한 자코비 반복법, 가우스-자이텔 그리고 SOR법 (Successive Overrelaxation method) 등이 헬름홀츠 방정식의 풀이에 이용될 수 있다[46, 72]. 그러나 본 방법들의 수렴 특성은 행렬의 조건수에 민감하게 좌우되기 때문에 일반적으로 크릴로브 부분공간법이 응용되어 왔으며, 본 방식은 이론적 반복 횟수의 상한이 존재한다. 즉, 크기가  $n$ 인 시스템에 대해  $n$ 번 안에 수렴하는 장점을 지닌다. 일반적인 비-에르미트 행렬을 풀기 위해 GMRES(General Minimal Residual) 법이 사용될 수 있으며, 만약 헬름홀츠의 상호상관성이 보존되는 임피던스 행렬이 구성된다면, SQMR (Symmetric Quasi-Minimal Residual), Bi-CG, Bi-CGStab 혹은 MINRES (Minimal Residual) 법 등이 사용될 수 있다[63, 37, 69, 19]. 그러나 크릴로브 부분공간법을 이용한 수치적인 풀이 시, 시스템의 크기가 클수록 연산 횟수가 증가하며 이에 따른 round-off error의 누적 및 전파로 인해 종종  $n$ 회 이상의 반복이 필요하게 되며, 특히 행렬의 조건수가 높은 경우 수렴의 지연 현상이 두드러지는 문제가 발생한다. 경우에 따라서는 문제의 풀이 도중 해가 발산하는 break out이 발생할 가능성도 존재한다. 따라서 조건수를 낮추기 위하여 preconditioner의 사용은 필수적이며, 특히 임피던스 행렬에 적합한 preconditioner와 크릴로브 부분공간 반복법의 조합으로 여러 연구가 수행되었다[20, 60, 13, 28, 4, 32, 70].

### 1.1.3 Preconditioners

Preconditioner는 과학적인 계산에서 도전적인 문제를 위한 효율적인 방법 개발에 가장 중요한 요소이다. 따라서 효과적인 preconditioner의 개발에 많은 노력이 투입되었고, 최근 preconditioner는 직접법이나 크릴로브 부분공간법보다 더 활발한 연구 분야였다. 또한 최적의 범용 preconditioner는 존재할 가능성이 낮기 때문에,

앞으로도 많은 연구가 진행될 것으로 예상된다.

잘 알려진 바와 같이, preconditioner를 사용하는 것은 선형 시스템을 반복법에서 풀기에 보다 유리한 특성을 가진 다른 시스템으로 변환하는 것을 말한다. preconditioner는 이러한 변환에 영향을 미치는 행렬이다. 일반적으로, preconditioner는 계수 행렬의 스펙트럼 특성을 개선시키는 역할을 한다. 대칭 positive definite (SPD) 문제의 경우, conjugate gradient (CG)의 수렴 속도는 A의 고유값 분포에 따라 달라진다. 따라서 변환된(preconditioner가 적용된) 행렬이 더 작은 스펙트럼 조건수 및/또는 고유값을 1 주위에 군집할 수 있도록 도움이 되는 preconditioner를 사용한다. 비대칭 문제의 경우 상황은 더 복잡하며 고유값은 GMRES와 같은 비대칭 행렬 반복의 수렴을 설명하지 않을 수 있다[43]. 하지만 이런 경우에도 0에서 멀리 떨어지게 군집화된 스펙트럼은 특히 preconditioner가 적용된 행렬이 정규 행렬에 가까울 때 빠른 수렴을 보여주는 경우가 많다.

일반적으로 좋은 preconditioner는 다음 요구 사항을 충족해야 한다.

- preconditioner가 적용된 시스템은 쉽게 해를 구할 수 있어야 한다.
- preconditioner는 구성 및 적용 비용이 낮아야 한다.

첫 번째 속성은 preconditioner가 적용된 시스템이 반복법을 수행 중 빠르게 수렴되어야 한다는 것을 의미하며, 두 번째 속성은 각 반복마다 너무 비용이 크지 않음을 보장해야 한다는 것이다. 이 두 가지 요건은 서로 경쟁 관계에 있기 때문에, 두 가지 필요 사이에서 균형을 잡아야 한다. 좋은 preconditioner의 경우, preconditioner가 적용 됐을 때의 반복의 계산 시간은 적용되지 않았을 때 반복의 계산 시간보다 훨씬 짧아야 한다.

preconditioner를 구성할 때 허용 가능한 비용은 일반적으로 preconditioner를 재사용할 수 있는지 여부에 따라 달라진다. 동일한 계수 행렬 또는 천천히 변화하는 행렬과 서로 다른 우변을 가진 일련의 선형 시스템을 해결해야 하는 일반적인 상

항에서, 계산 비용은 반복적인 연산에서 추가적인 계산이 요구될 수 있기 때문에 강력한 preconditioner를 사용하는데 많은 시간이 필요 할 수 있다.

### Incomplete factorization

행렬  $A$ 의 preconditioner( $M = M_1 M_2$ )가 불완전 LU 분해에 기초하여 구성된다면  $A$ 에 대한 더 나은 근사치를 얻을 수 있다[20, 24, 27]. 불완전LU 인자는 가우스 소거법(Gauss-elimination)를 수행한 다음 일부 요소를 탈락시킴으로써 얻을 수 있다. 따라서 이 경우  $M_1$ 과  $M_2$ 는 이러한 과정에서 생성된 하삼각행렬 및 상삼각형 행렬이다.

$LU = M_1 M_2$ 의 근사치는 LU 인자에 허용되는 채움(fill-in) 요소의 수에 따라 달라진다. 가장 간단한 방법은 소위 ILU(0)로, 여기서  $A$ 와 동일한 비영(non-zero) 구조가 ILU에 유지된다.

채움 수준을 높임으로써 더 정확한 근사치를 얻을 수 있다. 여기서 두 가지 방법이 있다. 첫 번째는 구조 지향적이며, LU 요인에 더 많은 대각선을 추가하여 수행된다. 이를  $ILU(n_{lev})$ 로 나타내며,  $n_{lev} > 0$ 은 채움 수준을 나타내는 작은 정수이다. 이 방법은  $L$ 과  $U$ 를 위한 구조화된 선형 시스템을 만든다. 두 번째 방법은 채움 허용오차와 관련이 있다. 따라서 이 방법은 원소의 값 지향적이다. LU 분해 중에 요소의 값이 규정된 공차 아래로 떨어질 경우 이 요소는 0으로 설정된다. 이 불완전 LU 분해를 탈락 허용오차( $\tau$ )를 가진 ILUT( $\tau$ )로 표시한다.

헬름홀츠 방정식에서 구성된 행렬  $A$ 는 indefinite이기 때문에 간단한 LU 분해뿐만 아니라 불완전 LU 분해가 안정적이지 않을 수 있다. 이는 결국  $A$ 에 대한 근사치가 좋지 않은 LU 인자로 이어질 수 있다.

불완전 LU 분해는 기초적인 편미분 방정식을 기반으로 분석적으로 구성할 수도 있다. 이 접근방식은 analytic ILU(AILU)[39, 60]이다. 헬름홀츠 방정식에 대

한 AILU의 적용할때는 우선 헬름홀츠 연산자  $\mathcal{H}(p)$ 는 두 개의 연산자로 분할된다[40, 41].

$$\mathcal{H}(p) = -(\partial_{x_1} + \Lambda_1)(\partial_{x_1} + \Lambda_2) \quad (1.1)$$

$\Lambda_1$ 과  $\Lambda_2$ 의 양, 비국소 연산자를 사용한다. 이러한 비국소 연산자는 LU 분해 맥락에서 L 및 U 행렬의 채움(fill-in)으로 볼 수 있다. AILU를 얻기 위해, 비국소 연산자는 푸리에 공간에서 파수  $k$ 와 일부 상수  $c_1, c_2 \in \mathbb{C}$ ,  $Re(c_2) > 0$ 에 따라 국소 연산자에 의해  $\hat{k}$  푸리에 주파수인 조건과 함께  $\Lambda_{1,2}^{approx} = \mathcal{F}_{x_2}^{-1}(c_1 + c_2 \hat{k}^2)$ 으로 근사된다.

$$\mathcal{H}^{approx}(p) + \mathcal{H}(p) = \mathcal{H}^{approx}(p) + f \quad (1.2)$$

혹은 고정점 반복법 방식으로 다음과 같이 사용한다.

$$\mathcal{H}^{approx}(p^{j+1}) = (\mathcal{H}^{approx} - \mathcal{H})(p^j) + f \quad (1.3)$$

상수  $c_1$ 과  $c_2$ 는 고정점 반복의 수렴 계수가 가능한 한 작도록 결정된다.  $\rho = \rho(c_1, c_2, k, \hat{k}, k_{x_1})$ 와  $c_1, c_2 \in \mathbb{C}$ 가 있으므로 주어진  $k$ 에 대해 4개의 실 변수를 최적화해야 한다. 고정점 반복법 방식 때문에 AILU는 반복법으로 사용할 수 있지만 수렴속도는 느리다고 알려져 있다[57]. preconditioner로서 AILU는 ILU(0)보다 수렴 속도 향상과 계산 효율 면에서 더 우수하다. 그러나 수렴 성향은  $k$ 와  $h$ 에 다소 강하게 의존한다[64].

## Multigrid method

멀티그리드(Multigrid) 방법[10, 13]은 헬름홀츠 문제의 preconditioner일 뿐만 아니라 해를 구하는 방법으로도 고려되었다. 멀티그리드 방법은 self-adjoint 연산자를 가진 타원 방정식에 적용할 때 효율성과 견고성이 있는 방법으로 알려져



있다[10, 13, 45]. 멀티그리드 방법은 격자 크기에 관계없이  $O(N \log N)$ 의 복잡성과 함께 빠른 수렴을 보인다. 그러나 이러한 수렴을 달성하기 위해서는 멀티그리드 방법으로 모든 구성요소를 신중하게 설계해야 한다.

타원 편미분 방정식에서 발생하는 선형 시스템을 해결하기 위한 멀티그리드 방법의 효율은 두 가지 요소에서 나온다. 먼저, 많은 고전적인 반복법은 근사치의 오차에 강력한 스무딩 효과를 제공한다. 이때 오류가 반드시 작은 것은 아니지만 매끄럽다. 이것을 스무딩이라고 하는데, 이것은 멀티그리드 방법의 첫 번째 요소이다. 두 번째 원칙은 이른바 거친 격자 보정(coarse grid correction)이다. 미세 격자의 매끄러운 양은 적절한 절차에 의해 조잡한 격자에서 충분히 근사할 수 있다.

멀티그리드 방법은 격자 순서  $\Omega_h, \Omega_H$ , 여기서  $\Omega_h$ 는 미세 격자이고  $\Omega_H$ 는 격자 크기를 두 배로 증가시켜 얻은 해당 거친 격자, 즉 two-grid 방법을 고려함으로써 설명될 수 있다. twi-grid 사이클은 사전 평활, 거친 격자 보정 및 사후 평활로 구성된 사이클이다. 사전 평활 및 사후 평활에는 평활 속성을 가진 고전적인 반복법이 사용된다. 멀티그리드 방법에서 두개 이상의 격자 시퀀스  $\Omega_h, \Omega_{2h}, \Omega_{4h}, \dots, \Omega_{h_0}$ 를 고려하게 된다.  $\Omega_{h_0}$ 이 가장 거친 그리드이며, 두개의 격자를 사용하는 사례에서와 같은 원칙을 반복적으로 적용한다. 이를 활용하는 것은 일반적인 문제에 대해 매우 효율적인 반복법으로 이어진다.

그러나 멀티그리드 방법은 indefinite 헬름홀츠 방정식에서 효과적인 방법이 아니다. 이를 확인하기 위한 이산 헬름홀츠 연산자의 고유값(균질 Dirichlet 경계 조건 포함)은 다음과 같다[11, 28].

$$\begin{aligned} \lambda_h^{l_1, l_2} &= \mu_h^{l_1, l_2} - k^2 \\ &\equiv \frac{2}{h^2} (2 - \cos \pi l_1 h - \cos \pi l_2 h) - k^2, \quad l_1, l_2 = 1, \dots, \sqrt{N} - 1. \end{aligned} \tag{1.4}$$

$k^2$ 과 이산 라플라스 연산자  $\mu_h^{l_1, l_2}$ 의 고유값들이 같지 않다면 헬름홀츠 연산자의 고유값은 0이 아닌 값을 갖게 된다. 각각의 고유값이 같아지면 특이(singular) 행렬이 되고 영 공간(null space)은 다음과 같은 고유 함수에 의해 표현된다.

$$\tilde{p}_h^{l_1 l_2} = \sin(\pi l_1 x_1) \sin(\pi l_2 x_2) \quad (1.5)$$

이때  $l_1, l_2$ 는  $\lambda_h^{l_1, l_2} = 0$ 일 때의 값이다.

$\mu_h^{1,1}$ 은 라플라스 연산자의 최소 고유값일때,  $k^2 > \mu_h^{1,1}$ 인 조건에서 행렬은 양과 음의 고유값 모두를 갖게 된다. 이 경우 완화 효과가 낮은 점별 야코비(pointwise Jacobi) 반복은 수렴되지 않지만, 평활 특성이 좋기 때문에 멀티그리드 방법의 수렴은  $k^2$  증가의 경우에만 점진적으로 악화되어, 표준 멀티그리드 방법은 발산한다. 결과적으로, 사용된 가장 거친 수준이 이러한 매끄러운 고유 주파수를 충분히 나타낼 수 있을 만큼 충분히 미세하다면 멀티그리드 방법은 여전히 수렴한다. 따라서, 가장 거친 수준은 수렴을 제한한다.  $k^2$ 가 커지면 표준 멀티그리드 방법의 수렴을 위해 가장 거친 수준으로 더 많은 변수가 필요하게 된다.

또한 원점에 가까운 고유값은 거친 그리드에서 이산화 후 부호가 바뀔 수 있다. 부호 변하는 경우 거친 격자 해는 더 미세한 격자 문제에 수렴을 가속시키지 않고 수렴 저하(또는 발산)를 일으킨다.

이러한 문제를 일으키는 고유값과 관련된 거친 격자 보정에 대한 해결책[28]들이 개발되었다. 이 방법은 멀티그리드 방법으로 preconditioner가 적용된 GMRES와 중간 단계의 오류를 줄이기 위한 GMRES가 포함된다. 안정성 높은 수렴 결과는 보여주지만 중간 단계에서 상당히 많은 GMRES 반복이 필요하다.

고유값에 매우 가까운  $k^2$  값에 대해서도 표준 멀티그리드 방법이 작동하지 않기 때문에, 이 경우 부분공간 보정(subspace correction) 기법을 사용해야 한다[12].

헬름홀츠 방정식을 위한 진보한 멀티그리드 기반 방법은 파-선(wave-ray) 멀티그

리드 방법[11]이며, 일정한 또는 부드럽게 변화하는 파형수를 가진 헬름홀츠 문제를 위해 개발되었다. 이 방법은 1차 시스템 최소 제곱 버전의 헬름홀츠 방정식[52]에 채택되었으며, 평면 파형에서 파생된 거친 그리드 기반 함수 및 대수적 멀티그리드 방법[71]에서 사용되었다.

### Domain decomposition method

또 다른 중요한 preconditioner의 종류로 도메인 분해(domain decomposition) 기법[4]이 있다. 이러한 방법은 물리적 도메인을 직접법으로 국소 문제의 해를 구할 수 있는 가능한 더 작은 하위 도메인으로 분할하여 원래 문제를 해를 구한다. 도메인 분해 방법에서 계산 도메인은 중복될 수 있는  $m$  하위 도메인  $\Omega_i, i = 1, \dots, m$ 로 분해되거나 분할된다. 도메인 분해 방법의 기본 아이디어는  $\Omega_j$ 에서 국소(하위 도메인) 문제를 해결한 다음 두 인접 도메인 사이의 인터페이스에서 해를 교환하여 해를 찾는 것이다. 승법(multiplicative) 및 가법(additive)(또는 하이브리드) Schwarz 방법, Schur 보완 방법 및 FETI 방법[36, 67]와 같은 여러 종류의 도메인 분해 방법이 존재한다. FETI 방법은 이산 타원 연산자를 이산 Lagrange 승수 형태로 표현하는 것에 기초한다. 이를 통해 전체 식을 고전적인 Lagrange 함수와 인터페이스 수량 두 부분으로 분할할 수 있다. 이 방법은 계산 성능 측면뿐만 아니라 수치적 확장성 측면에서도 유리하다. 이러한 도메인 분해 방법을 통해 직접 해를 구하는데 사용하거나 크릴로브 부분공간방법의 preconditioner로 모두 사용할 수 있다.

헬름홀츠 방정식을 해결하기 위한 도메인 분해 방법에 대한 초기 연구 중 일부는 도메인 분해 방법이 차등 수준에서 중복되지 않는 것에 대한 수렴 분석을 제공하는 방법[25]에 기인한다. 잘 정의되는 조건(well-posedness)은 인터페이스의 복소로빈 조건의 형태로 일관성 조건을 통합함으로써 보장된다.

이산 수준에 대한 수렴 분석은 2차 정확도 유한차분법, 유한요소법에서 이루어졌다. 여기서 도메인 분해 방법은 전전히 수렴된다고 알려져 있다. 도메인 분해 방법은 임의의 상수  $\theta$ 로 파수  $k$ 를 대체함으로써 로빈 조건의 일반화를 통해 수렴 속도를 개선시킬 수 있다. 상수  $\theta$ 는 이 도메인 분해 방법과 관련된 반복 행렬의 스펙트럼 반경이 1보다 훨씬 작도록 선택되며, 이는 자동으로 이루어질 수 있다[25].

### Shifted Laplace preconditioner

연산자 기반의 preconditioner는 풀고자 하는 선형 시스템의 특성에 맞춰서 설계되어 사용된다. 헬름홀츠 방정식의 경우 높은 파수의 시뮬레이션을 할때 indefinite한 성질 때문에 수치적으로 매우 불안정해 지는데, 연산자 기반 preconditioner는 이를 안정화 시키는데 효과적인 역할을 한다. 헬름홀츠 방정식에 사용되는 연산자 기반 preconditioner의 라플라시안 연산자를 활용한 방법이 주로 사용된다. 그 일반적인 형태는 다음과 같이 나타낼 수 있다.

$$\mathcal{L} = -\nabla^2 - (\alpha + \beta i)k^2, \quad \alpha, \beta \in \mathbb{R} \quad (1.6)$$

이러한 preconditioner의 초기 연구로  $\alpha = 0, \beta = 0$ 으로 시프트 없이 순수 라플라시안 연산자를 preconditioner로 사용하는 방법[6]이 제안되었고, 여기서  $\alpha = -1, \beta = 0$ 의 값으로 실수 시프트를 이용한 방법[50]으로 positive definite preconditioner를 활용하여 성능을 개선하였다. 이러한 preconditioner들은 중간 크기의 파수에서는 반복법의 수렴 속도를 상당히 개선시켰지만, 큰 파수의 경우 수렴 속도를 충분히 개선시키지 못하였다. 이후 연구에서 보다 큰 파수에서도 수렴성을 개선시키기 위해  $\alpha$ 와  $\beta$ 의 값을 함께 조절하여 복소 시프트를 고려함으로써 일반화된 shifted Laplace(SL) preconditioner[32]가 제안되었다.

Dirichlet 경계조건에서 SL preconditioner의 고유값을 명시적으로 알 수 있다는 조건하에 SL preconditioner가 적용된 행렬에 대한 스펙트럼 분석[32, 34, 33]을 통해, 행렬의 고유값이 1 근처에 군집화 하는 것을 확인하였다. 그러나 [32]의 분석은 순수 복소수 시프트만을 사용한 SL preconditioner가 균질한 매질에 적용된 경우로 일반적인 경우에 적용할 수 없다는 한계가 있다. [70]에서는 대수적 관점에서 스펙트럼 분석을 통해 일반적인 조건에서 적용되는 복소 시프트를 사용하는 SL preconditioner의 다음과 같은 특성을 확인하였다.

- 유한차분법, 유한요소법, 유한체적법 등 이산화 방법에 의존하지 않고 일관적이 경향을 보여줌.
- 헬름홀츠 방정식의 비균질 물리적 매개변수(음속, 밀도, 감쇄 등)에서도 사용할 수 있음.
- 다양한 경계 조건(반사, 방사, PML(Perfect Matched Layer))에 대해서도 분석이 유효함.

또한 분석 결과를 GMRES 잔차의 규범에 대한 한계와 결합함으로써 반복 상한에 대해 나타내었다. 고유값은 원점을 포함하지 않는 원에 둘러 쌓여있다는 가정을 하였는데 이 때문에 파수가 작은 복소수를 가져야 한다. 또한 행렬의 고유 벡터가 조건이 좋아야(well-conditioned) 한다는 가정을 하였다. 이러한 가정에서 GMRES의 반복이 최소화 되기 위한 최적의 복소 시프트는  $\alpha = 0$ 이고  $\beta = |k^2|$ 일때 임을 보여주었다. 이때 최적의 시프트는 SL preconditioner를 사용한 선형 시스템이 충분히 정확하게 계산된다는 가정 하에 도출되었다. 하지만 실제로는 근사적으로 계산하게 되어 대수적인 분석이 항상 성립하는 것은 아니나 여러 연구[32, 34, 33]에서 근사적으로 계산되는 경우에도 대수적인 분석을 통해 얻어진 최적의 복소 시프트와 유사한 복소 시프트를 사용할 수 있다는 것을 보여주었다.

다른 고유값 분석[35]에서는 Dirichlet 경계조건에서 SL preconditioner에 대해

수행되었다. 여기서는 유한차분법의 고유값을 계산하였는데, 고유값을 1 주변에 군집화 하기위한 복소시프트의 값이  $\alpha = 1$ 과  $\beta < k$ 의 값이 필요하다는 결론을 보여주었다.

앞선 연구들의 SL preconditioner의 적용은 멀티 그리드 방법을 사용하거나 SL 연산자를 불완전 분해를 사용하여 근사적으로 계산하였다. 이를 다른 방법으로 활용할 때는 최적의 복소시프트가 다른 값을 갖게 된다. 도메인 분해에서 SL preconditioner를 적용할 때 [49]에서는  $\alpha = 1$ 과  $\beta \sim k^2$ 의 복소 시프트를, sweeping preconditioner에 SL preconditioner를 적용할 때 [30]에서는  $\alpha = 1$ 과  $\beta \sim k$ 의 복소 시프트를 사용하는 것이 최적의 결과를 보여주었다.

SL preconditioner를 효과적으로 근사하기 위한 연구로 멀티그리드 방법[31, 58] 또는 불완전 분해를 이용하는 방법[55]이 활용되어 왔다. 이중, 불완전 분해를 사용하는 방법이 multigrid를 이용한 방법보다 계산 효율성을 증대시킨다는 연구결과[34]도 보여주었다.

SL preconditioner에 대한 수치 실험들[14, 16, 23]에서는 전체 시스템의 크기  $N$ 에 대해 2차원 문제에서는  $O(N^{3/2})$ , 3차원 문제에서는  $O(N^{4/3})$ 의 시간 복잡도를 보이는 것을 관찰 했다. 또한 최적의 복소 시프트가 아니더라도 복잡도를 유지하는 것을 보여주었다.

SL preconditioner에 대한 연구들은 주로 근사 방법과 복소 시프트의 선택에 대해 중점적인 연구가 이루어졌고, 헬름홀츠 방정식의 이산화시 비대칭 일반행렬을 사용하였다. 하지만 헬름홀츠 방정식은 상호상관성(reciprocity)를 만족하기 때문 이산화된 임피던스 행렬은 복소 대칭 행렬로 자연스럽게 표현될 수 있다. 이를 활용하여 대칭 이산화를 하면 복잡도에는 영향을 미치지 못하지만 실용적인 관점에서는 저장공간 요구량과 산술 연산의 수를 절반으로 줄일 수 있다.

## 1.2 Objective

본 논문의 목적은 큰 규모의 희소 선형 시스템을 구성하는 헬름홀츠 방정식의 해를 효율적으로 계산하기 위한 최적의 SL preconditioner 조건을 찾는 것이다. 이를 위해 PML경계조건 층을 포함한 전체 공간 노드에 대한 대칭 이산화 방법을 제안한다. 이를 통해 수치적 방법들을 사용할 때 저장공간 요구량과 산술 연산의 수를 절반으로 줄인다. 해를 구하는데 사용되는 반복법 중 헬름홀츠 방정식에 적용시 가장 효과적인 방법을 찾기 위해 크릴로브 부분공간법들의 해를 계산하는데 필요한 연산횟수의 비교를 통해 효과적인 반복법을 찾는다. 그리고 반복법 과정 중 수렴 속도 개선을 위해 SL preconditioner을 사용하고 이를 근사하기 위한 방법으로 선형 시스템을 대칭 이산화한 것을 활용하여 불완전 Cholesky 분해를 사용한다. 이때 최적의 성능을 얻기 위해 SL preconditioner의 시프트와 불완전 Cholesky 분해의 탈락기준을 실험을 통해 확인한다.

## 1.3 Organization of the Thesis

본 논문의 구성은, 제 2장에서는 수치해를 구해야 하는 지배방정식인 헬름홀츠 방정식과 이를 이산화를 하기 위한 방법 대해서 다룬다. 제 3장에서는 크릴로브 부분공간 법들을 비교하여 최적의 크릴로브 부분공간법에 대한 분석과 SL preconditioner의 특징과 이를 근사하기 위한 행렬기반 preconditioner인 불완전분해 대해 설명한다. 제 4장과 5장에서는 수치예제들을 통해 앞서 설명한 내용들에 기반하여 SL preconditioner에 대한 성능을 분석하고 이를 종합하여 결론을 도출한다.

## Chapter 2

# The Helmholtz equation

헬름홀츠 방정식은 다음과 같은 시간 의존 파동 방정식에서 유도할 수 있다.

$$\frac{1}{c^2} \frac{\partial^2 u(t)}{\partial t^2} - \nabla^2 u(t) = 0 \quad (2.1)$$

여기서  $u$ 는 공간과 시간에 관한 함수,  $c$ 는 매질의 속도이며  $\nabla^2$ 은 라플라시안 연산자이다. 위 식의 시간 의존 압력  $u$ 는 다음과 같이 분해할 수 있다.

$$u(t) = pe^{-i\omega t} \quad (2.2)$$

$\omega > 0$ 이면 각 주파수를 나타내고,  $i = \sqrt{-1}$ 이다. 이 식(2.2)을 (2.1)에 대입하면 다음과 같은 헬름홀츠 방정식을 얻을 수 있다.

$$-k^2 p - \nabla^2 p = 0 \quad (2.3)$$



여기서  $p$ 는 파동장,  $k = \frac{\omega}{c}$ 인 파수다.

위 헬름홀츠 방정식을 일반적인 식으로 우변에 소스 항  $f$ 를 추가하여 경계조건과 함께 주어진 영역  $\Omega \subset \mathbb{R}^3$ 과 경계  $\partial\Omega$ 에서 다음과 같이 나타낼 수 있다.

$$\begin{cases} -k^2 p - \nabla^2 p = f & \text{in } \Omega, \\ \text{boundary conditions} & \text{on } \partial\Omega, \end{cases} \quad (2.4)$$

헬름홀츠 방정식의 해는  $1/k$ 의 스케일로 진동하기 때문에, 정확하게 근사하기 위해서는 파수  $k$ 의 증가에 비례하는 자유도  $N$ 이 필요하다. 더욱이, 오염 효과는 분산오차가 큰 이산화 방법을 쓸 경우 자유도  $N$ 을  $k$ 에 비례하는 값을 갖는 것만으로는 여전히  $k$  증가와는 독립적으로 상대 오차를 제한하기 부족할 수 있다.  $k$ 에 따라 자유도  $N$ 이 커지면 그만큼 매우 큰 행렬로 이어지며, 높은 계산 비용이 발생하거나 계산할 수 없는 수준에 이를 수 있다. 따라서 분산오차를 최소화 하여 자유도  $N$ 을 적절한 수준을 유지하는 것이 필요하다.

## 2.1 Boundary conditions

헬름홀츠 방정식을 수치적으로 잘 정의되고(well-posed) 고유한 해를 보장하기 위해서는 적절한 경계조건이 필요하다. 기본적인 경계 조건으로는 Dirichlet, Neumann 경계조건등이 있다. 하지만 이 경계조건들은 무한한 영역을 모사하는데 적합하지 않다. 무한한 영역으로 파동이 전파되는 것을 고려한다면 이를 모사할 수 있는 흡수 경계조건을 사용하여야 한다. 흡수 경계조건을 사용하여 경계면에서 외부로 나가는 파동을 모방하고 나가는 파동의 비물리적 반사가 없는 조건을 만

축하게 된다. 여기서 사용되는 경계조건으로는 기본적인 흡수 경계조건과 특정한 두께의 층으로 흡수 경계조건을 구성하는 Damping/Sopnge, PML 경계조건 등이 있다.

### Dirichlet boundary condition

Dirichlet 경계조건은 경계의 위치에서 고정값 상수를 사용하는 것으로 다음과 같이 나타낼 수 있다.

$$p = 0, \quad \text{on } \partial\Omega \quad (2.5)$$

### Neumann boundary condition

Neumann 경계조건은 다음과 같이 나타낼 수 있다.

$$\frac{\partial p}{\partial \nu} = 0, \quad \text{on } \partial\Omega \quad (2.6)$$

$\nu$ 는 경계 밖으로 수직인 방향이다.

### Absorbing boundary condition

흡수 경계 조건에 대한 몇 가지 공식이 있다[3, 5, 21, 29]. 그중 1차 흡수 경계 조건은 다음과 같다[29].

$$\left(\frac{\partial}{\partial \nu} - ik\right)p = 0, \quad \text{on } \partial\Omega \quad (2.7)$$

일부 경계 조건(BGT 조건 [5])을 사용하면 비대칭 시스템으로 구성하게 된다. 이러한 경우, 반복법 관점에서 시스템의 비대칭성과 indefinite 성질로 인해 해결하기 어려운 선형 시스템으로 이어질 수 있다.

## Damping/Sponge Layer

경계에서 비물리적 반사를 더욱 줄이기 위해, 원래 계산 영역을 둘러싼 감쇠를 위한 영역을 추가할 수 있다. 그 안에서 파장은 다음과 같은 감쇠 헬름홀츠 방정식에 의해 모델링된다.

$$-(1 - \alpha i)k^2 p - \nabla^2 p = f, \quad 0 \leq \alpha \ll 1 \quad (2.8)$$

여기서  $\alpha$ 는 추가 영역에서의 감쇠 비율을 나타낸다. 지구물리 분야에서 활용될 때는  $\alpha$ 는 퀄리티 요소  $Q = 1/\alpha$ 와 관련되어 있으며  $20 < Q < 100$  사이의 값을 가지게 된다. 따라서  $\alpha$ 는 0.05까지 사용할 수 있다. 이 추가적인 영역은 감쇠/스폰지 층으로 불린다. 이 경우 원 계산 영역과의 접점에서는  $\alpha$ 의 값이 0으로 설정되고 바깥쪽으로 갈수록 점차 증가되는 값을 사용한다. 따라서, 나가는 파장은 감쇠 층에서 점차 작아지고, 비물리적 반사도 작아지게 된다.

## Perfectly Matched Layer

반사가 없는 경계 조건은 Perfectly Matched Layer(PML) 기법[9, 8]을 사용하여 효과적으로 구현할 수 있다. PML은 입사각에 관계 없이 경계에 반사가 없는 큰 흡수 능력을 가지고 있으므로 경계 반사를 억제하기 위해 필요한 영역의 크기가 감소한다. PML은 대칭 또는 비대칭 형식으로 공식화할 수 있다.

식 (2.4)는 2차원에서 다음과 같이 나타낼 수 있다.

$$-k^2 p - \frac{\partial}{\partial x} \frac{\partial p}{\partial x} - \frac{\partial}{\partial z} \frac{\partial p}{\partial z} = 0. \quad (2.9)$$

여기서 다음과 같이 복소 좌표 스트레칭(complex coordinate stretching)[18]를 사용하여 치환하게 되면,

$$\begin{aligned}\frac{\partial}{\partial x} &\rightarrow \frac{1}{s_x} \frac{\partial}{\partial x}, \\ \frac{\partial}{\partial z} &\rightarrow \frac{1}{s_z} \frac{\partial}{\partial z}.\end{aligned}\tag{2.10}$$

아래와 같은 식을 얻을 수 있다.

$$-k^2 p - \frac{1}{s_x} \frac{\partial}{\partial x} \frac{1}{s_x} \frac{\partial p}{\partial x} - \frac{1}{s_z} \frac{\partial}{\partial z} \frac{1}{s_z} \frac{\partial p}{\partial z} = 0.\tag{2.11}$$

여기서  $s_x s_z$ 를 곱하면 [68]에서 소개된 대칭화된 식을 다음과 같이 얻을 수 있다.

$$-k^2 s_x s_z p - \frac{\partial}{\partial x} \frac{s_z}{s_x} \frac{\partial p}{\partial x} - \frac{\partial}{\partial z} \frac{s_x}{s_z} \frac{\partial p}{\partial z} = 0.\tag{2.12}$$

같은 방식으로 3차원에서는 다음과 같이 나타낼 수 있다.

$$-k^2 p - \frac{1}{s_x} \frac{\partial}{\partial x} \frac{1}{s_x} \frac{\partial p}{\partial x} - \frac{1}{s_y} \frac{\partial}{\partial y} \frac{1}{s_y} \frac{\partial p}{\partial y} - \frac{1}{s_z} \frac{\partial}{\partial z} \frac{1}{s_z} \frac{\partial p}{\partial z} = 0.\tag{2.13}$$

여기서  $s_x s_y s_z$ 를 곱하면 3차원에서 대칭화된 식을 다음과 같이 얻을 수 있다.

$$-k^2 s_x s_y s_z p - \frac{\partial}{\partial x} \frac{s_y s_z}{s_x} \frac{\partial p}{\partial x} - \frac{\partial}{\partial y} \frac{s_z s_x}{s_y} \frac{\partial p}{\partial y} - \frac{\partial}{\partial z} \frac{s_x s_y}{s_z} \frac{\partial p}{\partial z} = 0.\tag{2.14}$$

## 2.2 Discretization

편미분 방정식의 수치 해법은 주어진 방정식의 이산 아날로그를 얻는 단계를 포함한다. 헬름홀츠 방정식의 이산화는 유한 요소법(FEM), 유한 차분법(FDM) 등을 포함한 많은 방법으로 이산화할 수 있다. FDM은 간단한 기하학적 구조를 쉽게 이산화 할 수 있고, 복잡한 물리적 영역의 문제는 일반적으로 FEM을 사용한다. 헬름홀츠 방정식에 대한 모든 이산화가 직면한 중요한 문제는 다음과 같다.

파수가 큰 문제는 해가 크게 진동하게 된다. 이를 적절한 분해능으로 표현하려면 미세한 격자 또는 고차 이산화 방법의 사용이 필요로 한다. 매우 높은 파수에서 크게 진동하는 해의 정확도를 보장하기 위해서는 격자의 크기를 더욱 정교하게 다듬어야 한다.  $h$ 가 이산화의 격자 크기라고 가정하면  $kh$ 를 작게 유지하면 된다. 하지만  $k$ 가 증가하면  $kh$ 가 충분히 작게 유지되더라도 정확도가 저하된다는 것이 관찰되었다. 이는 격자 크기의 증가가 파수라는 헬름홀츠 방정식의 특성에 비례해야 함을 나타낸다. 이 문제는 파동 전파에서 오염 효과[2]라 불린다. 2차 정확도  $O(h^2)$  유한 차분 및 FEM 이산화의 경우 최소 10개의 파장당 격자수를 사용해야 한다. 그러나 값이 높을수록 파장이 짧아지고 따라서 짧은 파장을 표현할 수 있는 미세한 격자가 필요하게 된다.

또한 수치 해의 파수가 수학적으로 정확한 해의 값과 달라서 수치 분산 오차가 발생하게 된다. 오염 효과와 수치 분산 오차를 통제하기 위해, 고전적인 이산화는 매우 미세한 격자 간격이 필요하며, 이는 특히 주파수가 증가할 때 매우 큰 선형 시스템으로 이어진다. 따라서 오염 효과와 수치 분산 오차를 감소시킨다면 더 큰 격자 간격을 쓸 수 있게 되어, 같은 주파수 조건에서 더 작은 선형 시스템을 구성할 수 있어 계산 비용을 줄일 수 있다. 이를 위한 방법으로 콤팩트 스텐실 9점 분산 오차 최소화[47] 방법등 여러 기법[17, 66]이 있다.

본 논문에서는 유한 차분법을 사용하여 헬름홀츠 방정식의 이산화를 수행한다. 우선 식 (2.4)의 2차원 5점 유한 차분법은 각 방향의 격자 크기  $h$ 에서 이산화된 행렬  $A$ 는 다음과 같이 나타낼 수 있다.

$$A = \frac{1}{h^2} \begin{bmatrix} & & -1 & & \\ -1 & & 4 - k^2 h^2 & & -1 \\ & & & & \\ & & & & \\ & & & & -1 \end{bmatrix} \quad (2.15)$$

여기서 5점 유한 차분법은 앞에서 설명한 오염 효과와 수치 분산 오차의 문제로 높은 주파수에서 매우 큰 선형 시스템이 필요하게 된다. 이를 보완하기 위해 사용되는 분산 오차 최적화 컴팩트 스텐실 9점 차분법은 다음과 같이 나타낼 수 있다.

$$A = \begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7 & a_8 & a_9 \end{bmatrix} \quad (2.16)$$

$a_i, i = 1, \dots, 9$ 의 값들은 사용하는 차분법에 따라 다른 값을 대입하게 된다. 본 논문에서는 [59]의 방법으로 헬름홀츠 방정식을 이산화하여  $a_i, i = 1, \dots, 9$ 의 값을 결정하게 된다.

## Conventional Discretization of Symmetric PML

복소 대칭 indefinite 형태의 선형 오퍼레이터를 가진 식(2.12)에서 이산화를 진행한다. 우선 유한차분 연산자  $\delta_x$ 와  $\delta_z$ 를 각 좌표 방향으로 1차 미분을 정의하면 다음과 같다.

$$\begin{aligned}\delta_x p_{0,0} &= (p_{\frac{1}{2},0} - p_{-\frac{1}{2},0}) / \Delta x, \\ \delta_z p_{0,0} &= (p_{0,\frac{1}{2}} - p_{0,-\frac{1}{2}}) / \Delta z.\end{aligned}\tag{2.17}$$

격자 간격은 각각  $\Delta x$ 와  $\Delta z$ 로 나타낸다. 첨자는 이산형 격자에서 기준 좌표  $(x, z)$ 에 상대적인 노드의 위치를 나타낸다. 예를 들어  $p_{0,0} = p(x, z)$  및  $p_{1,0} = p(x + \Delta x, z)$  등이 있다.  $p_{\frac{1}{2},0} = p(x + \frac{1}{2}\Delta x, z)$ 와 같은 인덱스는 비틀림(staggered) 위치에서 정의되었음을 나타낸다. 유한차분 연산자를 한 번 더 연속 적용(식 (2.17))함으로써, 우리는 또한 2차 미분을 나타낼 수 있다.

$$\begin{aligned}\delta_x^2 p_{0,0} &= (p_{1,0} - 2p_{0,0} + p_{-1,0}) / \Delta x^2 \\ \delta_z^2 p_{0,0} &= (p_{0,1} - 2p_{0,0} + p_{0,-1}) / \Delta z^2.\end{aligned}\tag{2.18}$$

유한차분 연산자를 사용하여 식(2.12)를 기본적인 5점 스텐실 이산화를 따른다면 이산화식은 다음과 같아진다.

$$-(k^2 s_x s_z)|_{0,0} p_{0,0} - \delta_x \frac{s_z}{s_x} \delta_x p_{0,0} - \delta_z \frac{s_x}{s_z} \delta_z p_{0,0} = 0.\tag{2.19}$$

유한차분 연산자를 확장하면 식(2.19)는 다음과 같이 표현할 수 있다.

$$\begin{aligned}
& - (k^2 s_x s_z)|_{0,0} p_{0,0} \\
& - \frac{1}{\Delta x^2} \left( \frac{s_z}{s_x} \Big|_{\frac{1}{2},0} (p_{1,0} - p_{0,0}) - \frac{s_z}{s_x} \Big|_{-\frac{1}{2},0} (p_{0,0} - p_{-1,0}) \right) \\
& - \frac{1}{\Delta z^2} \left( \frac{s_x}{s_z} \Big|_{0,\frac{1}{2}} (p_{0,1} - p_{0,0}) - \frac{s_x}{s_z} \Big|_{0,-\frac{1}{2}} (p_{0,0} - p_{0,-1}) \right) = 0.
\end{aligned} \tag{2.20}$$

식(2.20)를 재배열 하면 다음과 같이 표현할 수 있다.

$$\begin{aligned}
& \left[ - (k^2 s_x s_z)|_{0,0} + \frac{1}{\Delta x^2} \left( \frac{s_z}{s_x} \Big|_{\frac{1}{2},0} + \frac{s_z}{s_x} \Big|_{-\frac{1}{2},0} \right) \right. \\
& \quad \left. + \frac{1}{\Delta z^2} \left( \frac{s_x}{s_z} \Big|_{0,\frac{1}{2}} + \frac{s_x}{s_z} \Big|_{0,-\frac{1}{2}} \right) \right] p_{0,0} \\
& - \frac{1}{\Delta x^2} \frac{s_z}{s_x} \Big|_{\frac{1}{2},0} p_{1,0} - \frac{1}{\Delta x^2} \frac{s_z}{s_x} \Big|_{-\frac{1}{2},0} p_{-1,0} \\
& - \frac{1}{\Delta z^2} \frac{s_x}{s_z} \Big|_{0,\frac{1}{2}} p_{0,1} - \frac{1}{\Delta z^2} \frac{s_x}{s_z} \Big|_{0,-\frac{1}{2}} p_{0,-1} = 0.
\end{aligned} \tag{2.21}$$

식 (2.21)은 대칭 때문에  $p_{0,0}$ ,  $p_{0,1}$  및  $p_{1,0}$ 의 세 계수만 정의함으로써 실질적으로 구현될 수 있다. 5점 이산화 방법은 심각한 수치 분산으로 인해 고주파 또는 고파수 영역에서 사용할 수 없기 때문에, 분산 오차 최소화 방법이 필요하다. 그러나, 분산 오차 최소화 방법을 사용할 때 대칭 PML의 이산화는 대칭 행렬을 산출하지 못할 수 있다. 따라서 분산 최소화 방법을 사용하면서 이산화된 행렬의 대칭성을 보존할 수 있는 방법이 필요하다.

### Dispersion-Minimizing Discretization of Symmetric PML

[59]에서 PML과 분산오차 최소화를 적용하면서 대칭화된 9점 차분법을 구현하였다. 이에 대한 구체적인 구현 방법은 다음과 같다. 우선 [47]에서 제안된 방법으로 기존의 5점 라플라시안과 45도 회전 라플라시안을 결합하여 4개의 대각선 결절점



을 추가로 조합한다.

$$\nabla^2 p \approx a \nabla_{(0)}^2 p + (1 - a) \nabla_{(45)}^2 p, \quad (2.22)$$

여기서  $\nabla_{(0)}^2$  및  $\nabla_{(45)}^2$ 는 이산화된 라플라시안이다. [47]의 조합된 라플라시안에는 동일한 격자 간격  $\Delta x = \Delta z$ 이 제한되어 있으며, 다음과 같은 유한 차분 연산자를 사용하여 이를 나타낼 수 있다.

$$\begin{aligned} \delta_x^{2D} p_{0,0} &= \frac{1}{2\Delta x} \left( p_{\frac{1}{2}, \frac{1}{2}} - p_{-\frac{1}{2}, \frac{1}{2}} + p_{\frac{1}{2}, -\frac{1}{2}} - p_{-\frac{1}{2}, -\frac{1}{2}} \right), \\ \delta_z^{2D} p_{0,0} &= \frac{1}{2\Delta z} \left( p_{\frac{1}{2}, \frac{1}{2}} + p_{-\frac{1}{2}, \frac{1}{2}} - p_{\frac{1}{2}, -\frac{1}{2}} - p_{-\frac{1}{2}, -\frac{1}{2}} \right). \end{aligned} \quad (2.23)$$

(2.23)는 2차 정확도를 갖는다. 아래와 같이 비 PML 영역에서 방정식 (2.23)를 사용하여 라플라시안을 구성할 수 있다( $s_x = s_z = 1$ ).

$$\begin{aligned} &((\delta_x^{2D})^2 + (\delta_z^{2D})^2) p_{0,0} = \\ &\quad (p_{1,1} - 2p_{0,1} + p_{-1,1} + 2p_{1,0} - 4p_{0,0} + 2p_{-1,0} \\ &\quad \quad + p_{1,-1} - 2p_{0,-1} + p_{-1,-1}) / (4\Delta x^2) \\ &\quad + (p_{1,1} + 2p_{0,1} + p_{-1,1} - 2p_{1,0} - 4p_{0,0} - 2p_{-1,0} \\ &\quad \quad + p_{1,-1} + 2p_{0,-1} + p_{-1,-1}) / (4\Delta z^2), \end{aligned} \quad (2.24)$$

(2.24) 또한 2차 정확도를 갖는다.

$$(\delta_x^{2D})^2 + (\delta_z^{2D})^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial z^2} + O(\Delta x^2, \Delta z^2). \quad (2.25)$$

$h = \Delta x = \Delta z$ 이면 (2.24) 방정식이 다음으로 단순화된다.

$$\begin{aligned} ((\delta_x^{2D})^2 + (\delta_z^{2D})^2) p_{0,0} = \\ \frac{p_{1,1} + p_{1,-1} - 4p_{0,0} + p_{-1,1} + p_{-1,-1}}{(\sqrt{2}h)^2}, \end{aligned} \quad (2.26)$$

[47]에서와 같이 45도 회전 격자 상의 라플라시안과 동일하다. 따라서 방정식 (2.24)는  $\Delta x = \Delta z$ 의 특수 사례로 (2.26) 방정식을 포함한다.  $(\delta_x^{2D})^2 + (\delta_z^{2D})^2$ 와 달리, 회전 그리드  $(x', z')$ 에서 유한 차분  $\delta_{x'}^2 + \delta_{z'}^2$ 은 그리드 크기가 동일하지 않은  $\Delta x \neq \Delta z$ 에 대해 다음과 같은 라플라시안을 적용하지 않는다.

$$\begin{aligned} \delta_{x'}^2 + \delta_{z'}^2 = \\ \frac{2}{\Delta x^2 + \Delta z^2} \left( \Delta x^2 \frac{\partial^2}{\partial x^2} + \Delta z^2 \frac{\partial^2}{\partial z^2} \right) + O(\Delta x^2, \Delta z^2). \end{aligned} \quad (2.27)$$

방정식 (2.22)과 유사하게 일반화된 라플라시안 두 버전을 PML 계수와 혼합한다.

$$\begin{aligned} \frac{\partial}{\partial x} \frac{s_z}{s_x} \frac{\partial}{\partial x} + \frac{\partial}{\partial z} \frac{s_x}{s_z} \frac{\partial}{\partial z} \approx \\ a \left( \delta_x \frac{s_z}{s_x} \delta_x + \delta_z \frac{s_x}{s_z} \delta_z \right) \\ + (1-a) \left( \delta_x^{2D} \frac{s_z}{s_x} \delta_x^{2D} + \delta_z^{2D} \frac{s_x}{s_z} \delta_z^{2D} \right). \end{aligned} \quad (2.28)$$

식 (2.28)로 일반화된 라플라시안의 대칭 이산화를 얻을 수 있다. 그러나 [47]에서 도입된 평균화 방법은 대칭 PML을 사용하더라도 직접 대칭이 되지 않는다. 평균화 방법은  $p_{0,0}$  항을 다음과 같은 이웃한 노달 포인트의 조합으로 대체하는

것이다.

$$\begin{aligned}
p_{0,0} \rightarrow & cp_{0,0} + d(p_{1,0} + p_{-1,0} + p_{0,1} + p_{0,-1}) \\
& + e(p_{1,1} + p_{1,-1} + p_{-1,1} + p_{-1,-1})
\end{aligned} \tag{2.29}$$

여기서 각 계수들은  $c + 4d + 4e = 1$ 의 제한으로, 가중 계수의 합계가 1이 되는 것을 의미한다.  $k^2 s_x s_z p$ 라는 항을 평균내면서 다음과 같은 형태로 대칭적으로 이산화할 수 있도록 해야한다.

$$(k^2 s_x s_z) \Big|_{\frac{i}{2}, \frac{j}{2}} p_{i,j}, \tag{2.30}$$

여기서  $i = -1, 0, 1$  및  $j = -1, 0, 1$ 이다. 위 방법들을 종합하여 평균화 방법과 결합된 라플라시안(식 (2.28))을 합쳐서 대칭 이산화율 형성한다.

$$\begin{aligned}
& -c(k^2 s_x s_z) \Big|_{0,0} p_{0,0} \\
& -d \left( (k^2 s_x s_z) \Big|_{\frac{1}{2},0} p_{1,0} + (k^2 s_x s_z) \Big|_{-\frac{1}{2},0} p_{-1,0} \right. \\
& \quad \left. + (k^2 s_x s_z) \Big|_{0,\frac{1}{2}} p_{0,1} + (k^2 s_x s_z) \Big|_{0,-\frac{1}{2}} p_{0,-1} \right) \\
& -e \left( (k^2 s_x s_z) \Big|_{\frac{1}{2},\frac{1}{2}} p_{1,1} + (k^2 s_x s_z) \Big|_{\frac{1}{2},-\frac{1}{2}} p_{1,-1} \right. \\
& \quad \left. + (k^2 s_x s_z) \Big|_{-\frac{1}{2},\frac{1}{2}} p_{-1,1} + (k^2 s_x s_z) \Big|_{-\frac{1}{2},-\frac{1}{2}} p_{-1,-1} \right) \\
& -a \left( \delta_x \frac{s_z}{s_x} \delta_x + \delta_z \frac{s_x}{s_z} \delta_z \right) p_{0,0} \\
& -(1-a) \left( \delta_x^{2D} \frac{s_z}{s_x} \delta_x^{2D} + \delta_z^{2D} \frac{s_x}{s_z} \delta_z^{2D} \right) p_{0,0} = 0.
\end{aligned} \tag{2.31}$$

균질 매체와  $s_x = s_z = 1$  및  $\Delta x = \Delta z$ 를 가정하면 (2.31) 방정식은 [47]에서 유도한 형태로 나타나게 되므로 [47]에서 도출한 가중치  $a, c, d, e$ 를 사용할 수 있다. 최적화된 계수의 값은  $a = 0.5461$ ,  $c = 0.6248$ ,  $d = 0.09381$ ,  $e = 1 \times 10^{-5}$ 이다. 대각선 노드의 기여  $e$ 는 작기 때문에 숫자 구현의 단순화를 위해  $a = 0.5461$ ,  $c = 0.6248$ ,  $d = 0.0938$ ,  $e = 0$ 를 사용한다.

간편한 수치 구현을 위해 방정식 (2.31)를 다음과 같이 확장 및 재배열할 수 있다.

$$\begin{aligned}
& \left[ -c(k^2 s_x s_z)|_{0,0} \right. \\
& + \frac{a}{\Delta x^2} \left( \frac{s_z}{s_x} \Big|_{\frac{1}{2},0} + \frac{s_z}{s_x} \Big|_{-\frac{1}{2},0} \right) + \frac{a}{\Delta z^2} \left( \frac{s_x}{s_z} \Big|_{0,\frac{1}{2}} + \frac{s_x}{s_z} \Big|_{0,-\frac{1}{2}} \right) \\
& + \frac{1-a}{4\Delta x^2} \left( \frac{s_z}{s_x} \Big|_{\frac{1}{2},\frac{1}{2}} + \frac{s_z}{s_x} \Big|_{-\frac{1}{2},\frac{1}{2}} + \frac{s_z}{s_x} \Big|_{\frac{1}{2},-\frac{1}{2}} + \frac{s_z}{s_x} \Big|_{-\frac{1}{2},-\frac{1}{2}} \right) \\
& \left. + \frac{1-a}{4\Delta z^2} \left( \frac{s_x}{s_z} \Big|_{\frac{1}{2},\frac{1}{2}} + \frac{s_x}{s_z} \Big|_{-\frac{1}{2},\frac{1}{2}} + \frac{s_x}{s_z} \Big|_{\frac{1}{2},-\frac{1}{2}} + \frac{s_x}{s_z} \Big|_{-\frac{1}{2},-\frac{1}{2}} \right) \right] p_{0,0} \\
& + \left[ -d(k^2 s_x s_z)|_{0,\frac{1}{2}} - \frac{a}{\Delta z^2} \frac{s_x}{s_z} \Big|_{0,\frac{1}{2}} + \frac{1-a}{4\Delta x^2} \left( \frac{s_z}{s_x} \Big|_{\frac{1}{2},\frac{1}{2}} + \frac{s_z}{s_x} \Big|_{-\frac{1}{2},\frac{1}{2}} \right) \right. \\
& \quad \left. - \frac{1-a}{4\Delta z^2} \left( \frac{s_x}{s_z} \Big|_{\frac{1}{2},\frac{1}{2}} + \frac{s_x}{s_z} \Big|_{-\frac{1}{2},\frac{1}{2}} \right) \right] p_{0,1} \\
& - \left[ \frac{1-a}{4\Delta x^2} \frac{s_z}{s_x} \Big|_{\frac{1}{2},-\frac{1}{2}} + \frac{1-a}{4\Delta z^2} \frac{s_x}{s_z} \Big|_{\frac{1}{2},-\frac{1}{2}} \right] p_{1,-1} \\
& + \left[ -d(k^2 s_x s_z)|_{\frac{1}{2},0} - \frac{a}{\Delta x^2} \frac{s_z}{s_x} \Big|_{\frac{1}{2},0} - \frac{1-a}{4\Delta x^2} \left( \frac{s_z}{s_x} \Big|_{\frac{1}{2},\frac{1}{2}} + \frac{s_z}{s_x} \Big|_{\frac{1}{2},-\frac{1}{2}} \right) \right. \\
& \quad \left. + \frac{1-a}{4\Delta z^2} \left( \frac{s_x}{s_z} \Big|_{\frac{1}{2},\frac{1}{2}} + \frac{s_x}{s_z} \Big|_{\frac{1}{2},-\frac{1}{2}} \right) \right] p_{1,0} \\
& - \left[ \frac{1-a}{4\Delta x^2} \frac{s_z}{s_x} \Big|_{\frac{1}{2},\frac{1}{2}} + \frac{1-a}{4\Delta z^2} \frac{s_x}{s_z} \Big|_{\frac{1}{2},\frac{1}{2}} \right] p_{1,1} \tag{2.32} \\
& + \text{the other terms for } p_{-1,-1}, p_{-1,0}, p_{-1,1}, p_{0,-1} = 0.
\end{aligned}$$

식 (2.32)은 대칭성 때문에 전체 9개의 계수가 아닌  $p_{0,0}$ ,  $p_{0,1}$ ,  $p_1$ ,  $p_{1,1}$ 의 5개의 계수만 할당하여 코드에 직접 구현할 수 있다.

## Dispersion Analysis

대칭 PML 방정식의 수치 분산 특성은  $\Delta x = \Delta z$ 일 경우 [47]의 연구의 결과와 동일하다.  $\Delta x \neq \Delta z$ 로 격자 간격이 같지 않다면 [47]에서 제안된 가중치는 최적의 분산 특성을 갖지 않게 된다. 격자 간격의 비율에 따른 효과를 분석하기 위해 PML 계수  $s_x, s_z$ 를 단일 값, 매질은 균질 매질로 가정하였다. 식 (2.31)에 가중치  $e = 0$ 를 사용하면 다음과 같은 식을 얻을 수 있다.

$$\begin{aligned}
 & -k^2(cp_{0,0} + d(p_{1,0} + p_{-1,0} + p_{0,1} + p_{0,-1})) \\
 & -a(\delta_x^2 p_{0,0} + \delta_z^2 p_{0,0}) \\
 & - (1-a)((\delta_x^{2D})^2 p_{0,0} + (\delta_z^{2D})^2 p_{0,0}) = 0.
 \end{aligned} \tag{2.33}$$

실 파수  $k$ 와 수치 파수  $k_{\text{num}}$ 사이의 관계는 각각  $k_x = k_{\text{num}} \cos(\theta)$ 와  $k_z = k_{\text{num}} \sin(\theta)$ 를 사용하여  $p_{0,0} = \exp(i(k_x x + k_z z))$ 와  $p_{1,0} = \exp(i(k_x(x + \Delta x) + k_z z))$ 를 대입하여 계산할 수 있다. 여기서  $\theta$  평면파의 전파 각도이다. 여기서 유한 차분 연산자를 다음과 같은 형태의 유사 미분 연산자  $\delta_x = 2 \sinh(\Delta x \partial_x / 2) / \Delta x$ 로 나타내면, 식(2.33)의 유한 차분 연산자는 다음과 같은 공간 푸리에 변환된 표현으로 나타낼 수 있다.

$$\begin{aligned}
 & -k^2 (c + 2d (\cos(k_x \Delta x) + \cos(k_z \Delta z))) \\
 & + a \left( \frac{4}{\Delta x^2} \sin^2 \left( \frac{k_x \Delta x}{2} \right) + \frac{4}{\Delta z^2} \sin^2 \left( \frac{k_z \Delta z}{2} \right) \right) \\
 & + \frac{1-a}{\Delta x^2} \left( \sin \left( \frac{k_x \Delta x - k_z \Delta z}{2} \right) + \sin \left( \frac{k_x \Delta x + k_z \Delta z}{2} \right) \right)^2 \\
 & + \frac{1-a}{\Delta z^2} \left( \sin \left( \frac{-k_x \Delta x + k_z \Delta z}{2} \right) + \sin \left( \frac{k_x \Delta x + k_z \Delta z}{2} \right) \right)^2 = 0.
 \end{aligned} \tag{2.34}$$

식(2.34)에서 파수의 비율( $v_{\text{num}}/v = k/k_{\text{num}}$ )을 계산할 수 있다. 이는  $k_{\text{num}} \max(\Delta x, \Delta z)$

$= 2\pi/G$ 에 대한 함수이고,  $G$ 는 가장 큰 격자 간격 방향에서의 파장당 격자수이다.  $1/G$ 은 가장 큰 격자 간격  $\max(\Delta x, \Delta z)$ 에 의해 정의되며 분산 곡선을 이용해 표현할 수 있다.

이에 따른 분석[59]에서 그림(2.1)와 같이 정규화된 위상 속도는  $1/G < 0.25$ (파장당 격자 수 4 이하)인 모든 비율에서 0.5%이하의 분산 오차를 보여주었다.

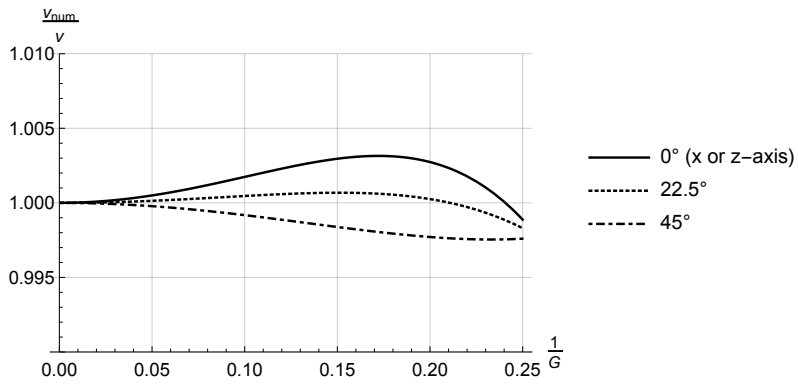


Figure 2.1: 정규화된 위상 속도

## Chapter 3

# Krylov subspace method and preconditioner

헬름홀츠 방정식은 다음과 같은 선형 시스템을 구성한다.

$$Ap = f, \quad A \in \mathbb{C}^{n \times n} \quad (3.1)$$

여기서  $A$ 는  $n \times n$  크기의 복소대칭 희소 행렬,  $p$ 와  $f$ 는  $n$ 개의 원소를 가진 복소 벡터이다.

일반적으로 (3.1)와 같은 선형 시스템은 가우스 소거법과 같은 직접법으로 해를 수할 수 있다. 하지만 3차원 문제의 경우 직접법으로 해를 구할 경우 저장공간의 필요량이 너무 커지기 때문에 비효율적이다.

이러한 문제를 가진 직접법의 대안으로 선형 시스템 (3.1)을 효율적으로 풀기 위해 반복법을 사용할 수 있다. 이러한 방법들은 크릴로브 부분공간을 기반으로 한다.

가장 잘 알려진 크릴로브 부분공간법으로는 conjugate gradient(CG) 방법이 있다. 그러나 CG의 경우 정확한 해를 얻기 위해서는 실수 positive definite 선형 시스

템이어야 하는데 헬름홀츠 방정식은 이에 해당하지 않기 때문에 CG를 적용하기 어렵다.

이 장에서는 헬름홀츠 방정식을 푸는데 활용되는 크릴로브 부분공간법들에 대해 설명한다. 헬름홀츠 방정식의 경우 복소 대칭이면 indefinite 선형 시스템을 가지기 때문에, 이러한 종류의 선형 시스템에 효과적인 방법을 고려한다. 여기에 해당하는 방법들로 아래 절에서 GMRES, Bi-CGSTAB, SQMR, CR들을 다룬다.

또한 반복법은 희소하고 크기가 큰 선형 시스템의 경우 preconditioner를 사용하지 않으면 해를 구하는데 충분히 효율을 낼 수 없다. 적절한 preconditioner를 사용한다면 선형 시스템을 변환하여 단위행렬에 가깝게 만들고 보다 빠르게 해를 구할 수 있다. 본 논문에서 크릴로브 부분공간법의 수렴을 효과적으로 가속시키는 preconditioner로 연산자 기반 preconditioner인 Shifted Laplace preconditioner와 행렬기반 preconditioner인 불완전분해 방법에 대해서 다룬다.



### 3.1 Krylov subspace methods

(3.1)의 선형 시스템의  $A$ 에 대해, 비특이 행렬  $M$ 과 초기 해  $x_0$ 가 주어지면,  $A = M - N$ 로 분리해 다음과 같은 기본 반복법을 정의할 수 있다.

$$\begin{aligned} Mx_m &= Nx_{m-1} + b \\ x_m &= M^{-1}(Nx_{m-1} + b) \\ &= x_{m-1} + M^{-1}(b - Ax_{m-1}) \end{aligned} \quad (3.2)$$

여기  $M = I$ 로 두면  $m$ 번째 해  $x_m$ 은 다음과 같이 나타낼 수 있다.

$$x_m = x_{m-1} + (b - Ax_{m-1}) = x_{m-1} + M^{-1}r_{m-1} \quad (3.3)$$

식 (3.3)에서 잔차  $r_m$ 은 다음과 같이 나타낼 수 있다.

$$r_m = b - Ax_m = (I - A)r_{m-1} = (I - A)^m r_0 \quad (3.4)$$

이에 따른  $x_m$ 은 다음과 같다.

$$\begin{aligned} x_m &= x_0 + r_0 + r_1 + \cdots + r_{m-1} \\ &= x_0 + \sum_{i=0}^{m-1} (I - A)^i r_0 \end{aligned} \quad (3.5)$$

$$\in x_0 + \text{span}\{r_0, Ar_0, \cdots, A^{m-1}r_0\} \equiv x_0 + \mathcal{K}_m(A, r_0)$$

여기서  $\mathcal{K}_m(A, r_0)$ 는 다음과 같은 크릴로브 부분공간을 나타낸다.

$$\mathcal{K}_m(A, r_0) = \text{span}\{r_0, Ar_0, \cdots, A^{m-1}r_0\}. \quad (3.6)$$

크릴로브 부분공간은  $m$ 차원이고,  $A$ 와  $r_0$ 로 구성된다.

$m$  번째 근사해  $x_m$ 는 아핀 공간  $x_0 + \mathcal{K}_m(A, r_0)$ 에서 찾는다. 이는 다음과 같이 표현할 수 있다.

$$x_m = x_0 + P_{m-1}(A)r_0 \quad (3.7)$$

여기서  $P_{m-1}$ 은  $m - 1$ 차 다항식이고, 여기에 따르는 잔차는 다음과 같다.

$$\begin{aligned} r_m &= (I - AP_{m-1}(A))r_0 \\ &= r_0 - AV_m y_m \end{aligned} \tag{3.8}$$

여기서  $V_m = [v_1, v_2, \dots, v_m]$ 으로  $\mathcal{K}_m(A, r_0)$ 의 정규 직교 기저 이고  $y_m \in \mathbb{C}^n$ 이다. 크릴로브 부분공간법은 기저  $V_m$ 과 벡터  $y_m$ 의 구성에 따라 방법이 달라지게 된다.  $V_m$ 의 기저는 아놀디(Arnoldi) 방법[1], 란초스(Lanczos) 방법[51] 등으로 구할 수 있고,  $y_m$ 은 잔차 투영 이나 잔차 노름 최소화법을 사용하여 구할 수 있다.

여기서 크릴로브 부분공간법은 기저  $V_m$ 과 벡터  $y_m$ 를 무엇을 사용하느냐에 따라 특성이 정해지기 때문에 반복법의 수렴 속도, 정확도 및 저장공간과 계산 시간에서의 효율을 위해 적절한 선택이 필요하다.

헬름홀츠 방정식의 행렬  $A$ 는 일반적으로 에르미트(Hermitian)도 아니고 positive-definite의 성질도 가지지 않는다. 따라서 복소 대칭인 비에르미트(non-Hermitian), 부정부호(indefinite) 성질을 가지는 행렬에 적용할 수 있는 크릴로브 반복법이 필요하다. 다음으로 여기에 해당하는 크릴로브 부분공간법으로 GMRES, Bi-CGSTAB, QMR, CR등에 대해 설명한다.

## GMRES

GMRES는 크릴로브 부분공간 전체에 대한 잔차 노름(norm)을 최소화하는 방법이다. 이를 위해 풀고자 하는 선형 시스템의 긴 회귀를 갖는 직교 벡터 시퀀스를 생성한다.

이 방법은 아놀디(Arnoldi) 방법을 사용하여 정규직교벡터 셋(set)을 찾게된다. 이에 대한 GMRES의 알고리즘은 다음과 같다.

---

**Algorithm 3.1** GMRES

---

$x_0$  is an initial guess,  $r_0 = b - Ax_0$ ,  $\beta = \|r_0\|_2$ ,  $v_1 = r_0/\beta$

Define the  $(m+1) \times m$  matrix  $H_m = (h_{i,j})_{1 \leq i \leq m+1, 1 \leq j \leq m}$ , Set  $H_m=0$

**for**  $j = 0, 1, \dots$ , until convergence **do**

$$w_j = Av_j,$$

**for**  $i=0, 1, \dots, j$  **do**

$$h_{i,j} = (w_j, v_i),$$

$$w_j = w_j - h_{i,j}v_i,$$

**end for**

$$h_{j+1,j} = \|w_j\|_2,$$

$$v_{j+1} = \frac{w_j}{h_{j+1,j}}$$

**end for**

Compute  $y_m = \operatorname{argmin}_y \|\beta e_1 - H_m y\|_2$ ,

Compute  $x_m = x_0 + V_m y_m$ ,

---

반복 과정 중  $j$ 번째 반복 중  $h_{j+1,j} = 0$ 이 되면 GMRES 알고리즘이 멈추게 될 수 있다. 이는 잔차벡터가 0이 된다는 뜻으로 이 단계에서 정확한 해를 구했다는 것이다. 그렇기 때문에  $h_{j+1,j}$ 의 값을 확인하는 것이 중요하다.

GMRES는 CG에 필적한 초선형(superlinear) 수렴 경향을 보여주며,  $n \times n$  선형 시스템 A를 GMRES를 사용해 수치 오차가 없는 정확한 산술계산을 한다면 n번의 반복 안에 수렴하게 된다. 하지만 실제 계산을 하게되는 헬름홀츠 방정식에서 나온 선형 시스템은 일반적으로 n의 크기가 크기 때문에 GMRES의 방법으로는 저장 공간과 계산 시간등의 비용의 문제로 실용적이지 못하다. 이는 아놀디(Arnoldi) 방법을 진행하는 중에 추가적인 벡터를 기록해야하는 저장공간이 증가하기 때문이다. 이를 보완하기위해서 재시작 및 절삭 방법등이 있다.

재시작 GMRES는 GMRES(m)으로 표현되는데 이는 알고리즘을 m번 반복 후 근사해를 구한 후, 이를 초기 벡터로 다시 GMRES를 적용하는 것이다. 하지만 GMRES(m)은 GMRES의 좋은 특성들을 잃게 된다. 여기에 해당되는 GMRES(m)의 단점으로 optimality 특성이 m번의 반복 동안에만 유지되고 초선형(superlinear) 수렴 경향도 잃게 된다는 것이다.

## Bi-CGSTAB

CGS 알고리즘은 잔차의 다항식을 제공하는 것을 기초로 하는 알고리즘으로, 종종 불규칙한 수렴패턴을 보여주는 알고리즘으로 큰 오류의 가능성을 가지는 알고리즘이다. Bi-CGSTAB 알고리즘은 CGS 알고리즘의 변형된 것으로 불규칙한 수렴패턴을 극복하기 위해 개발된 방법이다. CGS에서 잔차 벡터  $\tilde{r}_j = R_j^2(A)\tilde{r}_0$  (여기서  $R_j(A)$ 는  $A$ 의  $j$ -th 다항식)을 계산하는 대신 Bi-CSTAB은 다음과 같이 잔차 벡터를 계산한다.

$$\tilde{r}_j = Q_j(A)R_j(A)r_0 \quad (3.9)$$

$Q_j(A)$ 는 각 단계에서 반복적으로 정의되는 새로운 다항식으로 수렴성을 안정화 또는 부드럽게 만드는 역할을 한다.  $Q_j$ 는 다음과 같이 정의된다.

$$Q_{j+1}(t) = (1 - \omega_j t)Q_j \quad (3.10)$$

이는 다음 식과 동일하다.

$$Q_{j+1}(t) = (1 - \omega_0 t)(1 - \omega_1 t)\dots(1 - \omega_{st}) \quad (3.11)$$

여기서 모든  $k = 0, 1, \dots, j$ 에 대해 스칼라  $\omega_k$ 를 결정한다. 이는 스칼라  $\omega_k$ 에 관하여 (3.9)와 같이  $\tilde{r}_j$ 를 최소화함으로써 이루어진다. Bi-CSTAB의 완전한 파생은 [39, 49]에서 확인할 수 있다. 이에 대한 Bi-CGSTAB의 알고리즘은 다음과 같다.

---

**Algorithm 3.2** Bi-CGSTAB

---

$x_0$  is an initial guess,  $r_0 = b - Ax_0$ ,

set  $\tilde{r}_0 = \text{conjugate}(r_0)$ ,  $p = r_0$ ,  $rr = (r_0, \tilde{r}_0)$ ,

**for**  $j = 0, 1, \dots$ , until convergence **do**

$$Ap = Ap_j,$$

$$\alpha = \frac{rr}{(r_0, Ap)},$$

$$x_{n+1} = x_n + \alpha p_n,$$

$$r_{n+1} = r_n + \alpha Ap,$$

$$Ar = Ar_{n+1},$$

$$\omega = \frac{(\text{conjugate}(Ar), r_{n+1})}{(\text{conjugate}(Ar), Ar)},$$

$$x_{n+1} = x_{n+1} + \omega r_{n+1},$$

$$r_{n+1} = r_{n+1} - \omega Ar,$$

$$rr_{new} = (r_0, r_{n+1}),$$

$$\beta = \frac{rr_{new}}{rr} * \frac{\alpha}{\omega},$$

$$rr = rr_{new},$$

$$p_{n+1} = r_{n+1} + \beta(p_n - \omega Ap)$$

**end for**

---

이 방법은 짧은 재귀를 사용한다는 장점이 있지만 semi-optimality 특성을 기반으로 한다. 따라서 추가적인 행렬-벡터 곱이 필요하고 수렴 특성이 증명되지 않았다. Bi-CGSTAB 알고리즘의 주요 연산으로 두번의 행렬-벡터 곱과 4번의 내적이 필요하다.

일반적으로 Bi-CGSTAB의 수렴 경향은 부드러운 수렴률을 가지지만 약점이 있다. 이는 알고리즘의 반복이 진행되는 동안  $\omega_k$ 가 0에 매우 가까울 경우 수렴이

정체되거나 정지할 수 있다는 것이다. 이러한 문제가 발생하지 않고 수렴을 할 경우 Bi-CGSTAB의 잔차의 노름(norm)은 정확한 잔차인  $\|b - Ax_k\|$ 와의 비교가 필수적이다.

## QMR

QMR의 주요 아이디어는 GMRES에서 따르는 접근방식과 유사하게 삼대각(tridiagonal) 시스템을 최소자승(least square) 접근 방식으로 해결하는 것이다. 크릴로브 부분공간의 구축된 기저는 GMRES에서와 같이 직교하기 보다는 양직교이기 때문에, 계산된 해는 준-최적(quasi-minimal) 잔차의 해로 간주된다. QMR의 알고리즘은 다음과 같다.

---

### Algorithm 3.3 QMR

---

$x_0$  is an initial guess,  $r_0 = b - Ax_0$ ,  $\gamma_1 = \|r_0\|_2$ ,  $w_1 = v_1 = r_0/\gamma_1$ ,

Set  $\beta_1 = \delta_1 = 0$ ,  $w_0 = v_0 = 0$ ,

**for**  $n = 0, 1, \dots$ , until convergence **do**

**for**  $j = 1, 2, \dots, n$  **do**

$$\alpha_j = (Av_j, w_j),$$

$$\tilde{v}_{j+1} = Av_j - \alpha_j v_j - \beta_j v_{j-1}, \tilde{w}_{j+1} = A^T w_j - \alpha_j w_j - \delta_j w_{j-1},$$

$$\delta_{j+1} = \sqrt{|\tilde{v}_{j+1}, \tilde{w}_{j+1}|} \text{ If } \delta_{j+1} = 0 \text{ Stop,}$$

$$\beta_{j+1} = (\tilde{v}_{j+1}, \tilde{w}_{j+1})/\delta_{j+1},$$

$$w_{j+1} = \tilde{w}_{j+1}/\beta_{j+1}, v_{j+1} = \tilde{v}_{j+1}/\delta_{j+1},$$

**end for**

$$\gamma_{n+1} = -s_m \gamma_n,$$

$$\gamma_n = -c_m \gamma_m,$$

$$\alpha_m = c_m \alpha_m + s_m \delta_{m+1},$$

$$p_m = (v_m - \text{sum}(t_{i,m} p_i))/t_{m,m},$$

$$x_m = x_{m-1} + \gamma_m p_m,$$

**end for**

---



QMR은 비대칭 행렬에도 사용할 수 있는 일반적인 알고리즘이지만 헬름홀츠 방정식에서 나온 선형 시스템을 대칭 행렬로 이산화 한다면 이점을 활용할 수 있는 같은 특성을 공유하면서 대칭 시스템에 특화된 버전이 SQMR을 사용할 수 있다.

## CR

대칭 positive definite 행렬에 대한 대표적인 알고리즘으로 CG 알고리즘이 있다. 이와 비슷한 구조를 가지지만 A가 에르미트(Hermitian)인 특정한 경우에 대해 GMRES에서 CR이 도출될 수 있다. 이 경우 잔차 벡터들은 A-직교로 공액(conjugate)관계이어야 한다. 또한 벡터  $r_n$ 의  $n = 0, 1, \dots$  는 모두 직교한다. CR은 CG와 구조가 같지만 이러한 조건을 만족시키는 알고리즘이다. CR의 알고리즘은 다음과 같다.

---

**Algorithm 3.4 CR**

---

$x_0$  is an initial guess,  $r_0 = b - Ax_0$

set  $p_{-1} = 0, \beta_{-1} = 0,$

**for**  $n = 0, 1, \dots,$  until convergence **do**

$$p_n = r_n + \beta_{n-1}p_{n-1},$$

$$\alpha_n = \frac{(r_n, Ar_n)}{(Ap_n, Ap_n)},$$

$$x_{n+1} = x_n + \alpha_n p_n,$$

$$r_{n+1} = r_n + \alpha_n Ap_n,$$

$$\beta_n = \frac{(r_{n+1}, Ar_{n+1})}{(r_n, Ar_n)},$$

**end for**

---

CR은 일반적으로 CG와 유사한 빠른 수렴경향을 나타내기 때문에 에르미트(Hermitian)인 행렬의 해를 구할 때 주로 활용된다. 헬름홀츠 방적식에서 나오는 선형 시스템은 복소 대칭인 비에르미트(non-Hermitian)이지만 [65]를 통해 CR도 이와 같은 문제에도 최소 잔차(minimal residual) 특성을 유지하며 잔차의 노름(norm)에 대해 부드러운 수렴 경향을 유지할 수 있다는 것이 알려져 있다.

### 3.1.1 Convergence

크릴로브 부분공간법들의 수렴 경향을 보기 위해 2D 헬름홀츠 방정식을 물리적 영역의 크기가  $4km^2$ 인 문제를 풀었다. 경계조건은 흡수 경계조건을 사용하였다. GMRES의 경우 문제의 크기가 커지면 저장공간의 부족으로 사용할수 없다. 따라서 성능에 제약은 있지만 저장공간 사용에서 이점이 있는 GMRES(m)을 사용하였다. 재시작 상수  $m = 20$ 을 사용하였다. SQMR은 유사잔차를 최소화 하기 때문에 다른 방법들과 비교를 위해 계산 과정에 원 잔차를 추가적으로 계산하였다. 실험의 종료 조건은 상대 잔차가  $10^{-5}$  보다 작아지거나 1500회 이내에 수렴하지 않으면 종료하였다.

각 크릴로브 부분공간법들의 수렴까지 걸린 반복횟수는 Table 3.1에 나와있다. 같은 물리 영역을 풀때 파장당 격자수가 클수록 더 큰 수치 영역을 사용하게 되는데 이때 모든 방법들의 수렴속도가 느려지는 것을 확인하였다.

Figure 3.1 부터 3.4는 수렴할 때 까지의 행렬-벡터 곱을 나타낸 그래프 들이다. 수렴 경향을 보면 SQMR, CR이 GMRES(m)과 비슷한 수렴 경향을 보여주고 짧은 회귀를 사용하기 때문에 사용하기 좋은 방법으로 보인다. GMRES(m)은 문제의 크기가 더 커지면 수렴 경향을 개선하기 위해서 더 큰 재시작 상수 m을 사용할 필요가 있기 때문에 저장공간의 활용에서 단점이 있다. BiCGSTAB은 반복 횟수에서는 비슷한 결과를 보여주지만 행렬-벡터 곱에서 다른 방법들에 비해서 2배 가량 더 연산이 필요하였다. SQMR의 경우 좋은 수렴 경향을 보여주지만 유사잔차를 계산하기 때문에 원 잔차를 확인하기 위해서는 추가적인 연산이 필요하였다. 따라서 종합적으로 짧은 회귀를 사용하여 적은 저장공간이 필요하고 행렬-벡터

곱이 적게 필요한 CR이 가장 적은 계산 비용을 들이면서 헬름홀츠 문제의 해를 구하는데 적합할 것으로 사료된다. 이에 따라 이후의 preconditioner를 적용하는 실험은 CR을 사용하여 실험을 진행하였다.

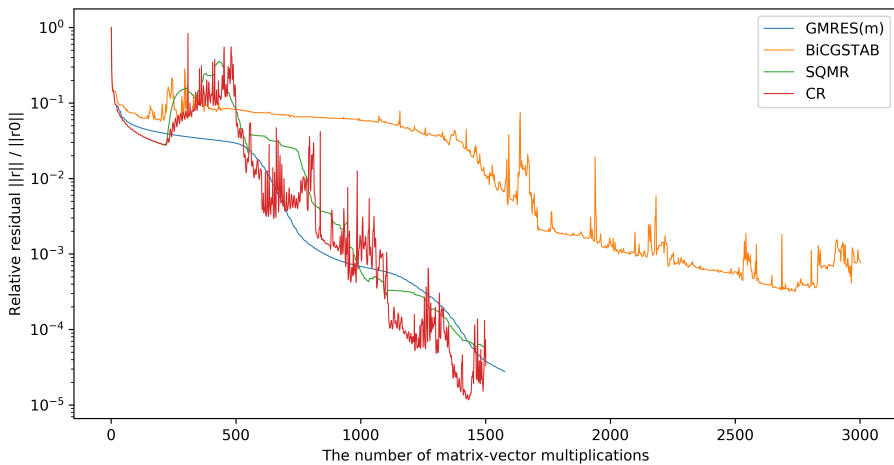


Figure 3.1:  $kh = 0.314159(G=20)$  일때 잔차 감소 경향에 따른 GMRES(m), BiCGSTAB, SQMR, CR의 행렬-벡터 곱

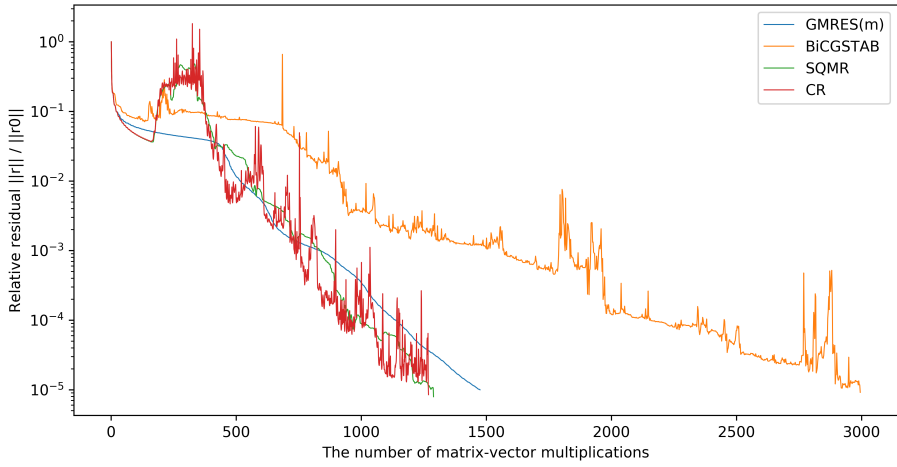


Figure 3.2:  $kh = 0.408407(G=15)$  일때 잔차 감소 경향에 따른 GMRES(m), BiCGSTAB, SQMR, CR의 행렬-벡터 곱

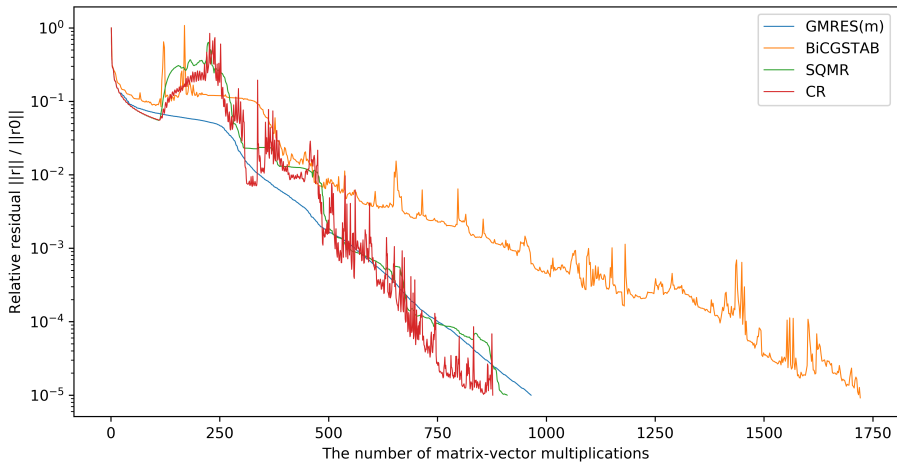


Figure 3.3:  $kh = 0.628319(G=10)$  일때 잔차 감소 경향에 따른 GMRES(m), BiCGSTAB, SQMR, CR의 행렬-벡터 곱

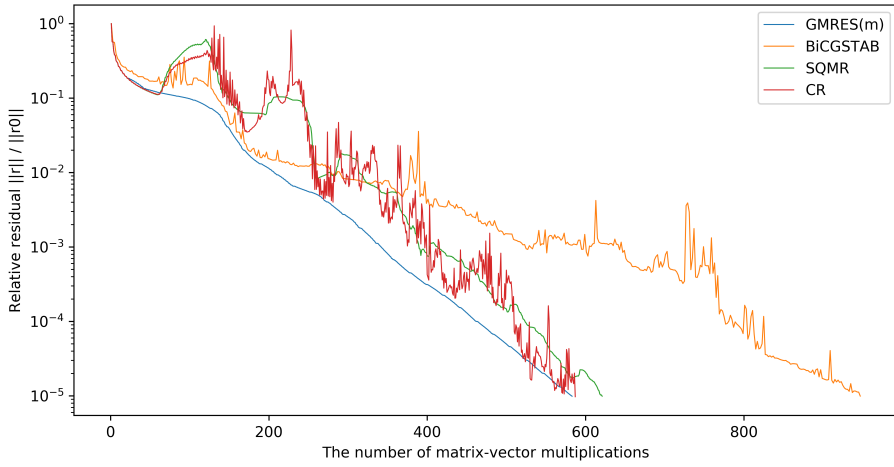


Figure 3.4:  $kh = 1.25664(G=5)$  일때 잔차 감소 경향에 따른 GMRES(m), BiCGSTAB, SQMR, CR의 행렬-벡터 곱

kh(G)	GMRES(20)	BiCGSTAB	SQMR	CR
0.314159(20)	<1500	<1500	<1500	<1500
0.408407(15)	1474	1497	1288	1268
0.628319(10)	964	860	909	876
1.25664(5)	582	473	620	586

Table 3.1: kh(G)에 따른 크릴로브 부분공간법들의 수렴까지의 반복횟수

## 3.2 Preconditioner

반복법이 직접법에 비해 약점으로 강건성(robustness)과 효율성의 부족이다. 이는 주로 크릴로브 부분공간법의 수렴성이 선형 시스템의 고유값 분포에 강하게 의존하기 때문이다. 이 약점은 preconditioner를 사용하여 극복할 수 있다. preconditioner를 사용하는 것은 원래의 선형 시스템을 같은 해를 가지는 다른 시스템으로 변형시키는 것으로, 이를 통해 더 빠르게 반복법의 해를 구할 수 있다.

$Ax = b$ 의 선형 시스템을 preconditioner  $M$ 을 활용해 좌측에 preconditioner를 적용하여 해를 구한다면 다음과 같은 시스템을 풀게 된다.

$$M^{-1}Ax = M^{-1}b \quad (3.12)$$

혹은 preconditioner를 우측에 적용해 다음과 같은 시스템을 풀 수도 있다.

$$AM^{-1}y = b, \quad x \equiv M^{-1}y \quad (3.13)$$

이 경우  $y = Mx$ 로 미지수  $y$ 에 대해 시스템의 해를 구한 후  $x$ 에 값을 구하게 된다. 마지막으로 가장 일반적인 형태로  $M = M_L M_R$ 로 분해된 형태로 나타내어 분할 preconditioner를 적용할 수 있다.

$$M_L^{-1}AM_R^{-1}y = M_L^{-1}b, \quad y = M_R x \quad (3.14)$$

여기서  $M_L$ 과  $M_R$ 은 삼각행렬이다. 원래 행렬이 대칭일 때는 시스템의 대칭을 보존하기 위해서는 분할 preconditioner의 적용이 필요하다.

이상적인 preconditioner는  $M = A$ 이다. 하지만  $A$ 의 역행렬은 일반적으로 계산하는데 비용이 크기 때문에 비현실적이다. 따라서 좋은 preconditioner는 다음과 같은 조건을 만족해야 한다.

1.  $M^{-1}$ 의 계산 비용이 낮아야 한다.
2.  $M^{-1}A$ 의 고유값이 1 주변으로 군집화해야 한다.

헬름홀츠 방정식의 이산화를 통해 구성된 선형 시스템은 강하게 분산된 스펙트럼을 가질 수 있는데 이는 indefinite 시스템이 된다. 이러한 시스템의 스펙트럼은 양과 음의 실수 고유값으로 구성된다. 이러한 시스템은 반복법으로 해를 구할 경우 느린 수렴 혹은 반복법의 작동이 멈출 수 있다. 좋은 preconditioner는 빠른 수렴 속도와 원 선형 시스템을 군집화된 스펙트럼을 갖게 변형시켜준다. 변형된 시스템의 스펙트럼은 특정 구역에 군집화된 고유값들로 구성된다. 또한 좋은 preconditioner가 적용된 시스템은 느린 수렴속도를 유발하는 0 근처의 고유값도 제거해 준다. preconditioner는 이러한 역할로 반복법을 이용하여 선형 시스템의 해를 구할 때 성능의 신뢰성은 어떤 크릴로브 부분공간법을 사용하느냐 보다 preconditioner의 질에 더 의존적이다.

preconditioner는 행렬 기반과 연산자 기반 두가지 유형으로 나눌 수 있다. 본 논문에서는 행렬기반 preconditioner인 불완전 Cholesky 분해와 연산자 기반 preconditioner인 shifted Laplace preconditioner의 조합으로 성능을 향상시켜 헬름홀츠 방정식에서 유도된 선형 시스템의 해를 구한다.



### 3.2.1 Shifted Laplace preconditioner

주요 관심사인 선형 시스템  $Ap = f$ 는 다음과 같은 헬름홀츠 방적식에서 유도된다.

$$-k^2p - \nabla^2p = f \quad (3.15)$$

여기서 헬름홀츠 연산자는 다음과 같이 다시 쓸 수 있다.

$$\mathcal{H} = -\nabla^2 - k^2 \quad (3.16)$$

행렬  $A$ 는 헬름홀츠 연산자  $\mathcal{H}$ 에 기반하여 만들어 지는데 이는 라플라시안 행렬  $L$ 과 헬름홀츠  $K$  행렬 두 부분으로 나뉘서 볼 수 있다.

$$A = -L - K \quad (3.17)$$

다음으로 여기에 복소 시프트( $\alpha + \beta i$ )를 추가하는 것으로 shifted Laplace(SL) 연산자를 정의할 수 있다.

$$\mathcal{L} = -\nabla^2 - (\alpha + \beta i)k^2 \quad (3.18)$$

여기서  $\alpha, \beta \in \mathbb{R}$ 이며  $\alpha \geq 0$ 이다. SL preconditioner  $M_{SL}$ 은  $\mathcal{L}$ 을 기반으로 다음과 같이 나타낼 수 있다.

$$M_{SL} = -L - (\alpha + \beta i)K \quad (3.19)$$

위 방법은 고유값의 군집화를 개선하고 0에서 멀리 떨어진 작은 고유값을 허수 축을 따라 추가 이동시키기 위해 도입 되었다.

### Eigenvalue of shifted Laplace preconditioner

헬름홀츠 연산자의 고유값은 다음과 같이 나타낼 수 있다.

$$\lambda_n = k_n^2 - k^2, \quad k_n = n\pi, \quad n \in \mathbb{N} \quad (3.20)$$

$k_n$ 은 시스템의 고유 주파수이다.

여기서 SL 연산자를 preconditioner로 적용한 시스템의 고유값 문제로 다음과 같은 식을 얻을 수 있다.

$$(-\nabla^2 - k^2)\phi_v = \lambda(-\nabla^2 - (\alpha + \beta i)k^2)\phi_v \quad (3.21)$$

이때의 고유값은 다음과 같다.

$$\lambda = \frac{k_n^2 - k^2}{k_n^2 - (\alpha + \beta i)k^2}, \quad |\lambda|^2 = \frac{(k_n^2 - k^2)^2}{(k_n^2 - \alpha k^2)^2 + (\beta k)^2} \quad (3.22)$$

여기서 고유값의 최대값과 최소값을 다음과 같이 찾을 수 있다.

$$|\lambda_{max}|^2 = \max\left(\frac{1}{\alpha^2 + \beta^2}, 1\right), \quad |\lambda_{min}|^2 = \frac{4}{(1 - \alpha)^2 + \beta^2} \left(\frac{\delta}{k}\right) \quad (3.23)$$

여기서  $\delta$ 는 최소 고유값이 0이되지 않게하는 아주 작은 수이다.

이제 여기서 다음과 같은 조건수를 얻을 수 있다.

$$\kappa^2 = \begin{cases} \frac{1}{4}\left(1 + \frac{1-2\alpha}{\alpha^2 + \beta^2}\right)(k/\delta)^2, & \alpha^2 + \beta^2 \leq 1 \\ \frac{1}{4}\left(1 + (1 - \alpha)^2 + \beta^2\right)(k/\delta)^2, & \alpha^2 + \beta^2 \geq 1 \end{cases} \quad (3.24)$$

$\alpha^2 + \beta^2$ 는 음수가이 아니기 때문에 주어진  $\alpha$ 에 대해 (22)의 첫 번째 식에서  $\alpha^2 + \beta^2 = 1$ 을 취하는 것이 가장 작은  $\kappa^2$ 을 제공한다. 마찬가지로  $\alpha \geq 0$ 의 경우,  $\alpha^2 + \beta^2 = 1$ 일 때  $\kappa^2$ 의 두 번째 식에 대해  $\kappa^2$ 은 최소값을 가진다. (반경 1의 원보다 더 낮은  $\kappa^2$ 를 주는 다른 원이 없음을 확인할 수 있다. 예를 들어 조건  $\alpha^2 + \beta^2 = 1 + \delta, \delta \geq 0$ 에서 이를 알 수 있다.) 조건  $\alpha^2 + \beta^2 = 1$ 에서,  $\alpha = 0$ 을 취할 경우,  $\beta = 1$ 일 때  $\kappa$ 는 최소값이다. 이 조합은 1D 모델 문제 경우 SL preconditioner에 대해 가능한 가장 낮은 조건수를 제공한다[32].

## Spectral analysis of preconditioned system

SL preconditioner의 성질을 파악하기 위해 대수적인 관점에서 분석한다. 이는 이산화 방법과 무관하고 불균질한 물리적 변수들에도 적용 가능하다[70].

경계조건과 함께 이산화된 헬름홀츠 방정식은 다음과 같은 형태로 나타낼 수 있다.

$$(L + iC - z_1K)p = f \quad (3.25)$$

여기서  $L$ 은 라플라시안,  $C$ 는 감쇄와 관련된 경계조건 항,  $K$ 는  $k$ 에 관한 헬름홀츠 항,  $f$ 는 소스 항이다. 복소수  $z_1 = \alpha_1 + i\beta_1$ 이다.

$L$ 과  $C$  실수 대칭이며 positive semidefinite이다. 그리고 행렬  $M$ 은 실수 대칭이며 positive definite이다. 여기에 행렬  $L + iC - z_1M$ 는 복소 대칭이며 indefinite이다.

이와 같은 형태로 SL preconditioner는 다음과 같이 나타낼 수 있다.

$$M_{SL} = L + iC - z_2K \quad (3.26)$$

복소수  $z_2 = \alpha_2 + i\beta_2$ 와 같이 나타낼 수 있다. 복소수  $z_2$ 는 복소 시프트 일반화된 값으로 반복법의 수렴성을 향상시키는 값을 선택한다. 이를 반복법에서 사용하기 위해 선형 시스템에 적용한 형태는 다음과 같이 나타낼 수 있다.

$$(L + iC - z_2K)^{-1}(L + iC - z_1K)p = (L + iC - z_2K)^{-1}f \quad (3.27)$$

위와 같은 조건에서 우선  $C = 0$ 일 때 헬름홀츠 방정식을 위한 SL preconditioner를 고려하여  $z_1, z_2$ 에 따라 시스템의 고유값  $\sigma$ 가 어떻게 위치하는 지 분석한다.

preconditioner가 적용된 행렬의 고유값  $\sigma$ 는 다음의 일반화된 고유값 문제의 해이다.

$$(L - z_1 K)x = \sigma(L - z_2 K)x \quad (3.28)$$

여기서 행렬  $(L - z_1 K)$ 와  $(L - z_2 K)$ 는  $Lx = \lambda Kx$ 의 고유 벡터를 같은 고유 벡터로 공유한다.

$L$ 은 대칭 positive semidefinite이고  $K$ 은 대칭 positive definite이기 때문에 고유값  $\lambda$ 는 실수이며 음의 값이 아니다. 위 식에서  $\lambda Kx$ 로  $Lx$ 를 대체하면 다음 식을 얻을 수 있고,

$$(\lambda - z_1)Kx = \sigma(\lambda - z_2)Kx \quad (3.29)$$

여기서 다음 식이 성립하는 것을 알 수 있다.

$$\lambda - z_1 = \sigma(\lambda - z_2) \quad (3.30)$$

따라서  $z_2 \neq \lambda$ 라면

$$\sigma = \frac{\lambda - z_1}{(\lambda - z_2)} \quad (3.31)$$

고유값  $\lambda$ 는 preconditioner가 적용된 시스템의 고유값  $\sigma$ 들이 위치한 복소평면상에 있는 식(3.31)의 매개변수화로 간주할 수 있다.

이 식의 값을 결정하기 위해  $\sigma = \sigma^r + i\sigma^i$ 로 식 (3.30) 대체하면 다음과 같은 식을 얻을 수 있다.

$$\lambda - \alpha_1 - i\beta_1 = \sigma^r(\lambda - \alpha_2) - i\sigma^r\beta_2 + i\sigma^i(\lambda - \alpha_2) + \sigma^i\beta_2 \quad (3.32)$$

이 식을 실수부와 허수부로 나누면 다음과 같다.

$$\begin{aligned} \lambda - \alpha_1 &= \sigma^r(\lambda - \alpha_2) + \sigma^i\beta_2 \\ -\beta_1 &= -\sigma^r\beta_2 + \sigma^i(\lambda - \alpha_2) \end{aligned} \quad (3.33)$$

여기서  $\beta_1 = \alpha^r \beta_2$ 라면, 두번째 식은  $\alpha^i = 0$ 이 된다. 이 경우가 아니라면  $\lambda$ 에 대해 다음과 같은 식을 얻는다.

$$\lambda = \alpha_2 + \frac{\sigma^r \beta_2 - \beta_1}{\sigma^i} \quad (3.34)$$

실수부에서  $\lambda$ 를 대입하면 다음과 같은 결과가 나온다.

$$\beta_2(\sigma^r)^2 - (\beta_1 + \beta_2)\sigma^r + \beta_2(\sigma^i)^2 + (\alpha_1 - \alpha_2)\sigma^i = -\beta_1 \quad (3.35)$$

이 식은 모든  $\alpha_1, \beta_1, \alpha_2, \beta_2$ 에 대해서 성립한다. 여기서  $\beta_2 = 0$ 인 경우와  $\beta_2 \neq 0$ 인 경우의 고유값의 스펙트럼을 정의할 수 있다.

$\beta_2 = 0$ 인 경우 고유값  $\sigma = \sigma^r + i\sigma^i$ 는 다음과 같은 식을 만족하며 복소평면 상의 직선에 분포하게 된다.

$$-\beta_1\sigma^r + (\alpha_1 - \alpha_2)\sigma^i + \beta_1 = 0 \quad (3.36)$$

$\beta_2 \neq 0$ 인 경우  $\sigma = \sigma^r + i\sigma^i$ 는 복소평면에 위에 다음과 같은 식을 만족하는 원 위에 분포하게 된다.

$$\left(\sigma^r - \frac{\beta_2 + \beta_1}{2\beta_2}\right)^2 + \left(\sigma^i - \frac{\alpha_2 - \alpha_1}{2\beta_2}\right)^2 = \frac{(\beta_2 - \beta_1)^2 + (\alpha_2 - \alpha_1)^2}{(2\beta_2)^2} \quad (3.37)$$

원의 중심  $c = \left(\frac{\beta_2 + \beta_1}{2\beta_2}, \frac{\alpha_2 - \alpha_1}{2\beta_2}\right)$ 이고 반경  $R = \sqrt{\frac{(\beta_2 - \beta_1)^2 + (\alpha_2 - \alpha_1)^2}{(2\beta_2)^2}}$ 이다.

반복법의 수렴에 대해 이해하기 위해서는 원점이 원에 둘러싸여 있는지 확인해야 한다.

$\beta_1\beta_2 > 0$ 인 경우 원점은 원에 둘러싸여 있지않는데 이는 다음과 같은 식을 통해

$$\frac{(\beta_2 + \beta_1)^2 + (\alpha_2 - \alpha_1)^2}{(2\beta_2)^2} > \frac{(\beta_2 - \beta_1)^2}{(2\beta_2)^2} + \frac{(\alpha_2 - \alpha_1)^2}{(2\beta_2)^2} \quad (3.38)$$

원점과 원의 중심의 거리가 반경보다 크다는 것으로 확인할 수 있다.

원의 중심과 반경은  $z_1, z_2$ 로 다음 같이 복소 값으로 나타낼 수 있다.

$$\begin{aligned} c &= \frac{z_1 - \bar{z}_2}{z_2 - \bar{z}_2} \\ R &= \left| \frac{z_2 - z_1}{z_2 - \bar{z}_2} \right| \end{aligned} \quad (3.39)$$

이제 여기서 보다 일반적인  $C \neq 0$ 인 감쇄된 헬름홀츠 연산자  $L + iC - z_1K$ 에 대해 SL preconditioner를 고려하여  $z_1, z_2$ 에 따라 시스템의 고유값  $\sigma$ 가 어떻게 위치하는 지 분석한다. preconditioner가 적용된 행렬의 고유값  $\sigma_S$ 는 다음과 같이 주어진다.

$$(L + iC - z_1K)x = \sigma_S(L + iC - z_2K)x \quad (3.40)$$

$\lambda_S$ 를 일반화된 문제의 고유값이라 하면,

$$(L + iC)x = \lambda_S Kx \quad (3.41)$$

이전  $C = 0$ 일 때의 분석 처럼 위 두 식은 같은 고유 벡터  $x$ 를 공유하고 preconditioner가 적용된 시스템의 고유값  $\sigma_S$ 은  $\lambda_S$ 에 관한 식으로 다음과 같이 쓸 수 있다.

$$(\lambda_S - z_2)\sigma_S = \lambda_S - z_1 \quad (3.42)$$

여기서  $C = 0$ 일 때와 다른 점은  $\lambda_S$ 가 복소수라는 것이다.  $\lambda_S$ 가 복소수일 경우, 고유값  $\sigma_S$ 는 일반적으로 복소평면상에 직선이나 원 안에 위치하지 않게된다. 하지만 복소 시프트를 사용하여 이 위치를 조정하는 것이 가능하다. 여기서  $\beta_2 = 0$ ,  $\beta_2 < 0$  그리고  $\beta_2 > 0$ 인 경우에 대해 다룬다

$\beta_2 = 0$ 일 경우  $\sigma_S = \sigma_S^r + i\sigma_S^i$ 는 반평면 상에 존재하게 된다. 이는 다음을 통해 알

수 있다.  $\beta_2 = 0$ 를 대입하면 식 (3.42)는 다음과 같아진다.

$$(\lambda_S - \alpha_2)\sigma_S = \lambda_S - z_1 \quad (3.43)$$

이 식을 실수부와 허수부로 나누면 다음과 같다.

$$\begin{aligned} \lambda_S^r \sigma_S^r - \lambda_S^i \sigma_S^i - \alpha_2 \sigma_S^r &= \lambda_S^r - \alpha_1 \\ \lambda_S^r \sigma_S^i - \lambda_S^i \sigma_S^r - \alpha_2 \sigma_S^i &= \lambda_S^i - \beta_1 \end{aligned} \quad (3.44)$$

허수부 식에서  $\sigma_S^i = 0$  또는  $\lambda_S^r = \alpha_2 - \frac{\beta_1}{\sigma_S^i} + \lambda_S^i \frac{1 - \sigma_S^r}{\sigma_S^i}$ 인 것을 확인 할 수 있다. 이를 실수부 식에 대입하면 다음을 얻을 수 있다.

$$-\beta_1 \sigma_S^r + (\alpha_1 - \alpha_2) \sigma_S^i + \beta_1 = \lambda_S^i ((\sigma_S^r - 1)^2 + \sigma_S^i{}^2) \quad (3.45)$$

여기서  $\lambda_S^i \geq 0$ 이기 때문에 우변항은 0보다 크거나 같게 된다. 따라서 다음 조건을 만족하게 된다.

$$-\beta_1 \sigma_S^r + (\alpha_1 - \alpha_2) \sigma_S^i + \beta_1 \geq 0 \quad (3.46)$$

$\beta_2 < 0$ 인 경우 고유값  $\sigma_S$ 는 중심  $c = \frac{z_1 - \bar{z}_2}{z_2 - \bar{z}_2}$ 와 반경  $R = \left| \frac{z_2 - z_1}{z_2 - \bar{z}_2} \right|$ 인 원 위 혹은 안쪽에 위치하게 된다. 이는 다음을 통해 알 수 있다.

$$\begin{aligned} |\sigma_S - c| &= \left| \frac{\lambda_S - z_1}{\lambda_S - z_2} - \frac{z_1 - \bar{z}_2}{z_2 - \bar{z}_2} \right| \\ &= \left| \frac{(\lambda_S - z_1)(z_2 - \bar{z}_2) - (\lambda_S - z_2)(z_1 - \bar{z}_2)}{(\lambda_S - z_2)(z_2 - \bar{z}_2)} \right| \\ &= \left| \frac{\lambda_S(z_2 - z_1) + (z_1 - z_2)\bar{z}_2}{(\lambda_S - z_2)(z_2 - \bar{z}_2)} \right| \\ &= \left| \frac{\lambda_S - \bar{z}_2}{\lambda_S - z_2} \frac{z_2 - z_1}{z_2 - \bar{z}_2} \right| \\ &= \left| \frac{\lambda_S - \bar{z}_2}{\lambda_S - z_2} \right| R \end{aligned} \quad (3.47)$$

여기서  $\beta_2 < 0$ 인 경우  $|\sigma_S - c| < R$ 을 확인 할 수 있다.

$|\frac{\lambda_S - \bar{z}_2}{\lambda_S - z_2}| < 1$ 인 것은  $\lambda_S = \lambda_S^r + i\lambda_S^i$ 로 나타내면 다음 식을 얻을 수 있다.

$$\left| \frac{\lambda_S - \bar{z}_2}{\lambda_S - z_2} \right|^2 = \frac{(\lambda_S^r - \alpha_2)^2 + (\lambda_S^i + \beta_2)^2}{(\lambda_S^r - \alpha_2)^2 + (\lambda_S^i - \beta_2)^2} \quad (3.48)$$

$\beta_2 < 0$ 이기 때문이 앞서 설명한 조건에 만족하게 하는 다음 식을 얻을 수 있다.

$$\frac{(\lambda_S^r - \alpha_2)^2 + (\lambda_S^i + \beta_2)^2}{(\lambda_S^r - \alpha_2)^2 + (\lambda_S^i - \beta_2)^2} \leq 1 \quad (3.49)$$

$\beta_2 > 0$ 인 경우 고유값  $\sigma_S$ 는 중심  $c = \frac{z_1 - \bar{z}_2}{z_2 - \bar{z}_2}$ 와 반경  $R = \left| \frac{z_2 - z_1}{z_2 - \bar{z}_2} \right|$ 인 원 위 혹은 밖에 위치하게 된다. 이는 앞서  $\beta_2 < 0$ 의 설명을 통해 알 수 있다.

위 설명을 통해 preconditioner가 적용된 행렬의 고유값이 위치한 복소 평면의 구역을 알 수 있다. 이러한 영역은 매개변수  $z_1$ 과  $z_2$ 에 의해 결정된다. 고유값이 위치한 복소 평면의 영역은 매질의 속도나 밀도와 같은 다른 물리적 매개변수나 행렬의 크기나 격자 크기  $h$ 와 같은 계산 매개변수에 의존하지 않는다는 것을 알 수 있다.

### Upper bound on residual norm

여기서는 GMRES 잔차 노름(norm)에 대한 상한과 스펙트럼 분석에 대한 설명을 한다. 이 상한은 스펙트럼이 원으로 둘러싸인다고 가정하며, 따라서 이 경계는 앞서 설명한 고유값의 분포와 결합될 수 있다. 앞서 preconditioner가 적용된 행렬의 고유값을 반지름  $R$ 과 중심  $c$ 가 있는 원으로 둘러싸여있는 것을 확인하였다. 이때  $k$ 번 반복 후 GMRES 잔차 노름(norm)  $\|r^k\|$ 은 다음을 만족한다.

$$\frac{\|r^k\|}{\|r^0\|} \leq c_2(X) \left( \frac{R}{|c|} \right)^k \quad (3.50)$$



이 식에서  $X$ 는 고유 벡터의 행렬이고  $c_2(X)$ 는 2-노름(norm)의 조건수이다. 이 조건수가 클 경우, 상한은 수렴에 관한 정보를 제공하지 않는다. 이 경우 고유값의 위치와 preconditioner가 적용된 크릴로브 부분공간법의 수렴 거동 사이에는 관계가 없기 때문이다. 다행스럽게도, 헬름홀츠 문제에서는 고유 벡터 행렬의 조건수가 상대적으로 작을 것으로 예상할 수 있다.  $C = 0$ 인 경우 preconditioner가 적용된 행렬의 고유 벡터는 시프트 변수와 무관하다.  $c_2(X)$ 가 실제로 작을 것으로 예상되는데 이것은 다음을 통해 알 수 있다.

$$\begin{aligned} X^T K X = I &\leftrightarrow c_2(X^T K X) = 1 \leftrightarrow c_2(K^{1/2} X) = 1 \\ c_2(X) &= c_2(K^{-1/2} K^{1/2} X) \leq c_2(K^{-1/2}) c_2(K^{1/2} X) \end{aligned} \quad (3.51)$$

여기서 다음 조건을 얻을 수 있다.

$$c_2(X) \leq \sqrt{c_2(K)} \quad (3.52)$$

$C \neq 0$ 일 경우 preconditioner가 적용된 시스템의 고유 벡터는 직교 성질은 아니지만, 많은 문제에서  $C = 0$ 일 때와 비교적 작은 변화로 고려할 수 있으며, 이 경우 여전히  $c_2(X)$ 가 작다고 기대할 수 있다.

### Selection of shift

앞에서 다룬 분석[70]에서 Dirichlet 경계조건일 경우 SL preconditioner( $\nabla^2 + (\alpha + \beta i)k^2$ )의 복소 시프트는,  $\alpha \leq 0$ 라는 제한 하에, 조건수는  $\alpha = 0$ ,  $\beta = \pm 1$ 일 때 최소가 된다고 알려져 있다. 하지만 흡수 경계조건일 경우 고유값의 스펙트럼 분석[33]을 통해  $\alpha = 1$ ,  $0 < \beta < 1$ 일 경우 고유값의 군집화가 더 향상될 수 있는 것을 보여주었다. 이를 통해 이후 연구[38, 42, 22]에서  $\alpha = 1$ 을 사용하고  $\beta$ 의 값을 기준으로 최적의 복소 시프트에 대한 연구가 진행되었다.  $\beta$ 기준의 복소 시프트를

$\epsilon$ 으로 다시 표현하면 다음과 같이 표현할 수 있다.

$$(\alpha + \beta i)k^2 \rightarrow k^2 + \epsilon i \quad (3.53)$$

이때  $0 < \beta < 1$ 을 만족하는  $\epsilon$ 의 값을 사용하기 위해  $\epsilon \leq k^2$ 으로 가정한다.

최적의 복소 시프트에 관한 연구[38, 42, 22]에서 SL preconditioner가 좋은 성능을 갖기 위한  $\epsilon$ 의 조건은 다음과 같다.

헬름홀츠 방적식의 이산화에 사용되는 격자  $h$ , 파수  $k$ 와 복소 시프트 $\epsilon$ 에 독립적인 임의의 상수  $O > 0$ 가 있다고 하면  $hk^2 \geq O$ 란 조건에서 다음을 만족하는  $O_1, O_2 > 0$ 가 존재한다.

$$hk\sqrt{|k^2 - \epsilon|} \leq O_1 \quad (3.54)$$

$$\|I - M_{SL}^{-1}A\|_2 \leq O_2 \frac{\epsilon}{k} \quad (3.55)$$

따라서  $\epsilon/k$ 가 충분히 작다면  $M_{SL}$ 은 A에 대해 좋은 preconditioner가 된다. 위와 같은 가정아래 GMRES로 preconditioner가 적용된 시스템의 해를 구하게 되면 k에 독립적인 반복수로 수렴하게 된다[38].

$\epsilon/k$ 에 따른 경향을 확인하기 위한  $\epsilon = k/4$ 에서  $\epsilon = k^2$ 까지의 실험[38]에서는  $\epsilon = k/4$ 일때 항상 가장 좋은 결과를 보여주었고  $\epsilon = k$ 보다 작아지기 시작하면 이보다 큰 값을 가질때 보다 큰 성능 향상이 있는 것을 보여주었다.

$$\epsilon \propto k, \quad 0 < \epsilon < k \quad (3.56)$$

본 논문에서는 위와 같은 조건의 최적의 복소 시프트를 구한다. 또한  $\epsilon$ 는 k에 비례하는 값을 사용하며 k를 생략하고 비례 상수만 나타낸다.(만약  $\epsilon = 0.5k$ 라면  $\epsilon = 0.5$ 로 표기)

## Spectrum of preconditioned system

표 3.2는 원 시스템 A와 preconditioner가 적용된 시스템  $M_{SL}^{-1}A$ 의 조건수를 나타낸 것이다. 시프트의 값에 따라서 조건수가 변하지만 항상 원 시스템 A보다 조건수가 개선된 것을 알 수 있다.

System	Condition number
A	633.60
$AM_{SL}^{-1}, \epsilon = 0.1$	1.04
$AM_{SL}^{-1}, \epsilon = 10$	6.86
$AM_{SL}^{-1}, \epsilon = 30$	27.45

Table 3.2: 행렬 A와  $M_{SL}^{-1}A$ 의 시프트의 값에 따른 조건수

그림 3.5는 원 시스템의 고유값 분포를 나타낸다. 원 시스템의 고유값들은 복소 평면 상에서 넓은 범위에 존재하고 있는 것을 확인할 수 있다. 그림 3.6 ~ 3.8는  $M_{SL}$ 의 시프트의 값에 따라 시스템  $M_{SL}^{-1}A$ 의 고유값 분포를 나타낸다. 값의 변화에 따라 군집화 경향이 바뀌는 것을 확인할 수 있다. 예시에서는 시프트  $\epsilon = 0.1$ 을 쓴 그림 3.6에서 대부분의 고유값이 1에 모여있어 시스템이 단위행렬에 가깝다는 것을 알 수 있다. 이 외의 시프트  $\epsilon$  값을 사용한 경우에도 고유값들이 1 근처에서 군집화 되어 있다는 것을 확인할 수 있다. 하지만 시프트 값에 따라 군집화 되는 정도가 달라지기 때문에 적절한 시프트의 선택이 필요하다. 적절한 시프트를 사용하였어도  $M_{SL}$ 을 직접법으로 구하는 것은 큰 비용이 들기 때문에 이를 근사할 방법이 필요하다. 근사 방법으로는 행렬기반 preconditioner를 사용할 수 있다.

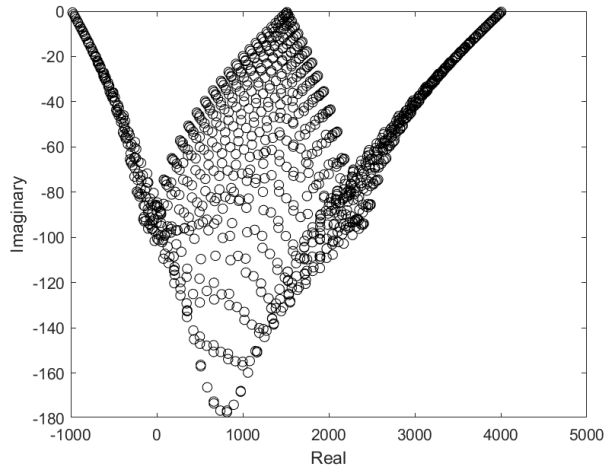


Figure 3.5: 원 시스템 A의 고유값 분포

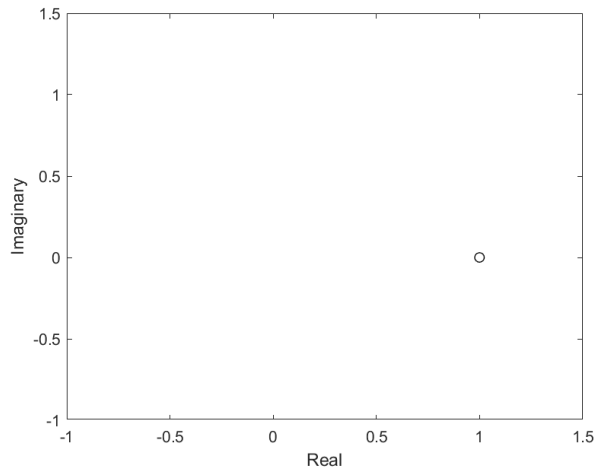


Figure 3.6:  $\epsilon = 0.1$  일 때  $M_{SL}^{-1}$ 을 직접법으로 계산하였을 때의 고유값 분포

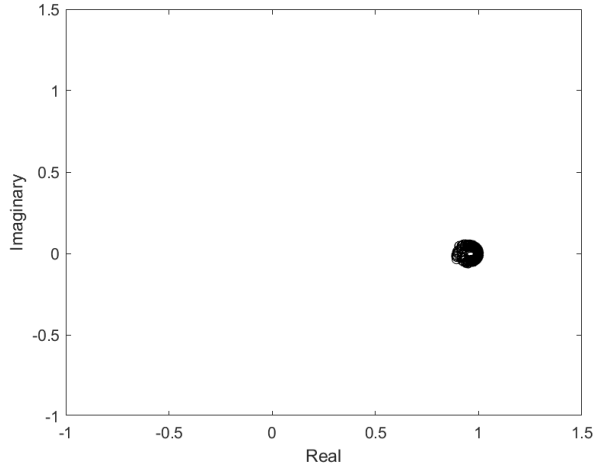


Figure 3.7:  $\epsilon = 10$  일 때  $M_{SL}^{-1}$ 을 직접법으로 계산하였을 때의 고유값 분포

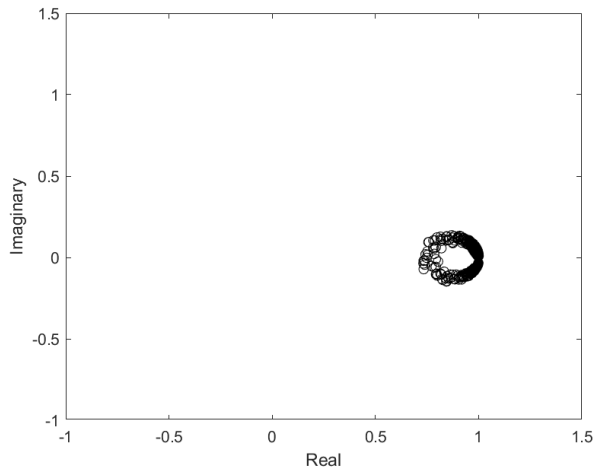


Figure 3.8:  $\epsilon = 30$  일 때  $M_{SL}^{-1}$ 을 직접법으로 계산하였을 때의 고유값 분포

### 3.2.2 Incomplete factorization

행렬을 가우스 소거법을 사용하여 분해하면 그 과정에서 행렬의 채움(fill-in)이 발생하게 된다. 분해 과정 중 특정 기준에 따라 일부 채움을 버리게 되면 불완전 분해된 행렬을 얻을 수 있다. 불완전 분해에 사용되는 행렬은 풀고자 하는 원 행렬을 사용하거나 앞 장에서 설명한 SL preconditioner같은 연산자에서 생성되는 행렬을 사용할 수 있다. 사용되는 행렬이 비대칭일 경우 불완전 LU 분해(Incomplete LU factorization, ILU), 대칭일 경우 불완전 Cholesky 분해(Incomplete Cholesky factorization, IC)를 수행하여 preconditioner를 얻게 된다.

가장 간단한 방법으로 ILU(0) 혹은 대칭행렬일 경우 IC(0)이 있다. 이 방법[57]은 원 행렬  $A$ 의 0이 아닌 원소의 구조와 같은 위치에 채움을 수행하여 얻을 수 있다. 쉽게 구현이 가능하고 비용이 적게드는 방법이지만 어려운 문제들의 경우 충분한 성능을 발휘하지 못한다. 따라서 더 정확한 근사를 위해 원 행렬의 연결성에 근거해 채워짐 수준(level of fill-in)에 따른 방법[44]을 사용할 수 있다. 채워짐 수준에 따른 방법은 행렬의 구조에 의존하는 방법으로  $k$ 에 따라  $A^{k+1}$ 의 비영인자 위치를 사용하는 방법으로 ILU( $k$ ) 혹은 대칭행렬일 경우 IC( $k$ )로 나타낸다. 여기서  $k$ 는 0 이상의 양의 정수를 사용하게 되고, 0을 사용할 경우 앞서말한 ILU(0), IC(0)와 같은 방법을 나타내게 된다. 많은 경우  $k = 1$ 만 사용하더라도  $k = 0$ 인 경우보다 상당한 성능의 향상을 보여준다.  $k = 1$ 보다 큰 값을 사용해야 할 정도로 매우 어려운 문제의 경우  $k$ 가 커짐에 따라 연산시간과 저장공간의 요구량이 크게 증가하기 때문에 사용하는데 제약이 따른다. 또한 채워짐 수준에 따른 방법은 대각지배행렬의 경우  $k$ 가 커지더라도 많은 채움의 절대값들이 작기 때문에 preconditioner 성능의 향상 대비 연산비용의 증가가 더 크게 된다. 이러한 문제를

개선시키기 위해 채움을 원소의 크기에 따라 사용하는 방법을 사용할 수 있다.

원소의 크기에 근거해 탈락 한도(drop-tolerance)를 사용하는 방법은 탈락 한도  $\tau$ 에 따라 채움을 수행하는 것이다.  $\tau$ 보다 크기가 큰 경우에만 채움을 하는 것이다. 크기를 기준으로 하면 preconditioner의 성능에 큰 영향을 미치는 것들 위주로 채움을 구성할 수 있다. 이 방법의 단점으로는 좋은 성능을 발휘하는 탈락 한도  $\tau$ 를 선택하기 어렵다 것이다. 또한 저장공간의 요구량을 예측할 수 없다는 문제점도 있다. 이 문제를 해결하기 위해 각 행(또는 열)의 비영의 수를 제한하여 저장공간의 요구량을 예측가능한 방법으로 이중 임계값(dual threshold) 방법[62]이 있다. ILUT( $\tau, p$ )로 표현되는 이 방법은 소거법의 과정 중 각 행(또는 열)의 채움을 기준  $\tau$ 에 부합하는 것 중 크기가 큰 순서대로  $p$ 개의 비영인자만 남겨두는 것으로 수행된다. 하지만 이 방법에서도 여전히 탈락 한도  $\tau$ 의 선택이 어렵다고 분해를 통해 생성된 L과 U가 원 시스템이 대칭 시스템이었어도 비대칭으로 생성될 수 있다는 문제가 있어 대칭 시스템에 사용하는 것에는 한계가 있다.

탈락 한도  $\tau$ 를 사용하는 방법들은 문제의 특성에 따라 사용되는  $\tau$ 값이 선택하기 어렵고 대칭 시스템에 적용하기 용이하지 않다. 대칭 시스템에 적합한 불완전 Cholesky 분해에서 이를 보완하는 방법[48]이 제안되었다. 이 방법은 원 행렬의 각 행(또는 열)에 비영의 수와 같은 양의 원소만 크기 순으로 남기고 나머지 원소들은 버리는 것이다. 이 방법은 IC(0)와 같은 저장공간을 요구하면서 개선된 수렴성을 보여준다. 문제에 따라 바뀌는 탈락 기준을 정할 필요 없고 예측 가능한 저장공간을 사용할 수 있는 방법이다. 하지만 어려운 문제에서는 수렴성이 느려진다는 단점이 있다. 이를 개선하기 위해 [53]의 방법이 제안되었다. 이 방법은 특정 정수  $p \geq 0$ 를 사용하여 크기순으로 갯수  $p$ 만큼의 비영 원소를 추가하는 것이다.  $p = 0$

일 때 [48]의 방법과 같아진다. 이 방법은 더 많은 저장공간을 사용하여 성능을 극적으로 향상시킨다. 이 방법(본 논문에서는 이 방법을 ICT(p)로 표현하겠다.)은  $\tau = 0$ 일때의 ILUT( $\tau, p$ )와 정확히 같은 방법이지만, 자유도가 더 높은 탈락한도  $\tau$ 를 사용하지 않기 때문에 사용하기 더 간단하다. 또한 저장공간의 사용량도 예측 가능하다.

본 논문에서 사용하는 불완전 Cholesky 분해의 구현은 다음에 나타낸 알고리즘인 Cholesky 분해의 jki 버전에 기초한다. 이 분해는 하삼각행렬 L의 j번째 열과 함께 A의 j번째 열을 덮어쓴다. 대각선 요소는 분해가 진행됨에 따라 업데이트된다.

---

**Algorithm 3.5** Cholesky factorization

---

```

for n = 1, ..., n do
     $a(i, j) = \sqrt{a(i, j)}$ 
    col`len = size( $i > j : a(i, j) \neq 0$ )
    for k = 1, ..., j - 1 &  $a(j, k) \neq 0$  do
        for i = j + 1, ..., n &  $a(i, k) \neq 0$  do
             $a(i, j) = a(i, j) - a(i, k) * a(j, k)$ 
        end for
    end for
    for i = j+1, ..., n &  $a(i, j) \neq 0$  do
         $a(i, j) = a(i, j) / a(j, j)$ 
         $a(i, i) = a(i, i) - a(i, j)^2$ 
    end for
end for

```

---



ICT(p)는 Cholesky 분해에서 j번째 반복이 시작될 때 하삼각 행렬 L의  $l_1, \dots, l_{j-1}$  열을 계산하고  $a(i, k) (i > k)$ 에서  $l_k$ 의 i번째 요소를 저장한다. 다음 계산에서 대각선 요소  $a_{ii}^{(j)} (i > j)$ 는

$$A_j = A - \sum_{k=1}^{j-1} l_k l_k^T \quad (3.57)$$

$a(i, i) (i \leq j)$ 에 저장된다.  $l_j$ 의 계산은 다음 과정에서 이루어진다.

$$A_j = l_j l_j^T - \sum_{k=j+1}^n l_k l_k^T \quad (3.58)$$

따라서  $l_{ik} = 0 (i < k)$ 이고, 이러한  $l_j$ 의 정의는 다음을 의미한다.

$$l_{jj}^2 = a_{jj}^{(j)}, \quad l_{jj} l_{ij} = a_{ij} - \sum_{k=1}^{j-1} l_{ik} l_{jk}, \quad i > j \quad (3.59)$$

ICT(p)는 이러한 표현식에서  $l_{jj}$ 와  $l_{ij}$ 를 계산하고  $l_j$ 에서 가장 큰  $n_j + p$ 개의 요소를 유지하며, 여기서  $n_j$ 는 원 행렬 A의 대각 성분을 제외한 하위 삼각형 부분에서 0이 아닌 요소의 개수이다.  $A_{j+1}$ 의 대각선 요소는 다음 연산으로 계산된다.

$$a_{ii}^{(j+1)} = a_{ii}^{(j)} - l_{ij}^2, \quad i > j \quad (3.60)$$

ICT(p)는 비대각 요소가 현재 열에서 가장 큰 요소 중 하나일 경우에만 해당 요소를 유지한다. 대각선 요소는 항상 유지된다. 특히, 이것은 불완전한 요인의 희소 패턴이 원래 행렬 A의 희소성 패턴을 포함하지 않을 수 있음을 암시한다. ICT(p) 구현의 장점은 탈락 허용오차를 기반으로 한 분해와는 달리 동적 메모리 관리가 필요 없다는 것이다.

---

**Algorithm 3.6** Incomplete Cholesky factorization, ICT(p)

---

**for**  $n = 1, \dots, n$  **do**

$$a(i, j) = \sqrt{a(i, j)}$$

$\text{col\_len} = \text{size}(i > j : a(i, j) \neq 0)$

**for**  $k = 1, \dots, j - 1$  &  $a(j, k) \neq 0$  **do**

**for**  $i = j + 1, \dots, n$  &  $a(i, k) \neq 0$  **do**

$$a(i, j) = a(i, j) - a(i, k) * a(j, k)$$

**end for**

**end for**

**for**  $i = j+1, \dots, n$  &  $a(i, j) \neq 0$  **do**

$$a(i, j) = a(i, j) / a(j, j)$$

$$a(i, i) = a(i, i) - a(i, j)^2$$

**end for**

    Retain the largest  $\text{col\_len} + p$  elements in  $a(j+1:n, j)$ .

**end for**

---

---

**Algorithm 3.7** Incomplete Cholesky factorization, IC(k)

---

```
for n = 1, ..., n do  
   $a(i, j) = \sqrt{a(i, j)}$   
  col`len = size(i > j : a(i, j) ≠ 0)  
  for k = 1, ..., j - 1 & a(j, k) ≠ 0 do  
    for i = j + 1, ..., n & a(i, k) ≠ 0 do  
       $a(i, j) = a(i, j) - a(i, k) * a(j, k)$   
    end for  
  end for  
  
  for i = j+1, ..., n & a(i, j) ≠ 0 do  
     $a(i, j) = a(i, j) / a(j, j)$   
     $a(i, i) = a(i, i) - a(i, j)^2$   
  end for
```

Retain sparsity pattern of the matrix  $A^{k+1}$ .

```
end for
```

---

이 알고리즘은 성능의 비교를 위한 IC(k)에 대한 알고리즘이다.

### 3.2.3 Fill-in comparison between ICT(p) and IC(k)

ICT(p)		IC(k)	
p	Fill-in (9+p)N	k	Fill-in (3 + 2k) <sup>2</sup> N
0	9N	0	9N
5	14N	1	25N
10	19N	2	49N
15	24N	3	81N
20	29N	4	121N

Table 3.3: 2차원 9점 컴팩트 스텐실 이산화일 때 ICT(p)와 IC(k)의 채움 갯수(N은 시스템의 크기)

표 3.3와 3.4는 각각 2차원과 3차원 컴팩트 스텐실로 이산화를 했을 때 ICT(p)와 IC(k) 각각으로 불완전 분해를 할 때 필요한 채움의 갯수를 나타낸 것이다. ICT(p)의 경우 2차원과 3차원 모두 p의 값을 1의 단위로 조절하며 채움의 양을 조절할 수 있다. 하지만 IC(k)의 경우 2차원에서는  $k^2$ , 3차원에서는  $k^3$ 에 비례하여 채움의 양이 늘어나게 된다. 작은 시스템을 풀 경우 적은 채움을 사용하고도 반복법의 수렴성을 개선할 수 있지만 문제의 크기가 크다면 더 많은 채움이 필요한 경우가 생긴다. IC(k)의 경우 추가적인 채움을 위해서 필요한 저장공간이 급격하게 늘어나는 것을 알 수 있다. ICT(p)의 경우 채움의 양을 더 유동적으로 조정할 수 있기 때문에 저장공간을 더 효율적으로 활용할 수 있다. 또한 같은 정도의 채움을 사용할 때도 [53]에서 ICT(p)가 더 나은 성능을 보여주는 것을 알 수 있다.

ICT(p)		IC(k)	
p	Fill-in (27+p)N	k	Fill-in (3 + 2k) <sup>3</sup> N
0	27N	0	27N
5	32N	1	125N
10	37N	2	343N
15	42N	3	729N
20	47N	4	1331N

Table 3.4: 3차원 27점 컴팩트 스텐실 이산화일 때 ICT(p)와 IC(k)의 채움 갯수 (N은 시스템의 크기)

### 3.2.4 Spectrum of preconditioned system

이전의 설명에서 SL preconditioner  $M_{SL}$ 를 직접법으로 행렬 A에 적용하여 시스템의 스펙트럼이 개선되는 것을 확인하였다. 행렬  $M_{SL}$ 은 A와 비슷한 시스템 크기를 가지기 때문에  $M_{SL}^{-1}A$ 를 직접법으로 계산하는데 제약이 따른다. 그렇기 때문에 이를 적절히 근사하는 방법이 필요하다. 근사하는 방법으로 불완전 분해를 사용할 수 있다. 적절한 시프트  $\epsilon$ 을 사용할 경우  $M_{SL}^{-1}$ 가 고유값의 스펙트럼을 개선 해주지만 이를 근사하는 과정에서 개선되는 정도의 차이가 생기게 된다. ICT(p)와 IC(k) 모두 성능을 끌어올리는 기준을 높이면  $M_{SL}^{-1}$ 의 성능을 최대치로 끌어낼 수 있으나 이는 비용의 문제로 더 효율적인 방법이 무엇인지 확인이 필요하다. 이에 대해서는 앞서 설명한 내용들을 통해 ICT(p)가 더 적은 저장공간 비용이 들면서 나은 성능을 나타 낼 수 있을 것으로 확인하였다. 여기서는 작은 시스템에서 각 불완전분해가 고유값 분포를 어떻게 나타내는지 확인해 본다. ICT(p)와 IC(k)으로  $M_{SL}$ 을 근사적으로 계산한 시스템의 고유값 분포를 그림 3.10와 그림 3.11에 나타내었다. 직접법을 적용 했을 때 보다는 고유값들이 넓은 범위에 분포하는 것을 확인 할 수 있다. 하지만 여전히 원 시스템보다 균집화되고 조건수를 낮추는 결과를 얻을 수 있다. 작은 크기의 시스템에서는 ICT(p)와 IC(k)가 비슷한 경향을 보여주는 것을 확인할 수 있다.

System	Condition number
A	633.60
$AM_{SL}^{-1}$ with ICT(p)	456.44
$AM_{SL}^{-1}$ with IC(k)	577.38

Table 3.5: 행렬 A와  $AM_{SL}^{-1}$ 의 조건수

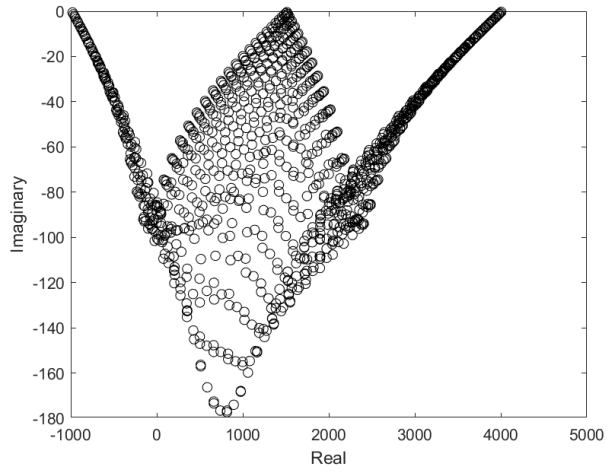


Figure 3.9: 원 시스템 A의 고유값 분포

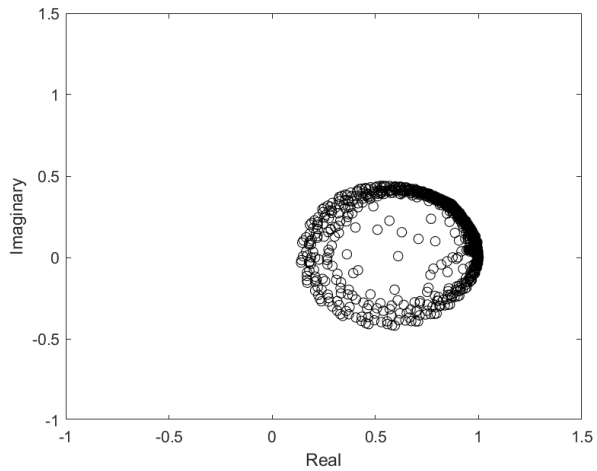


Figure 3.10:  $\epsilon = 0.1$  일 때  $M_{SL}^{-1}$ 을 ICT(p)로 계산하였을 때의 고유값 분포

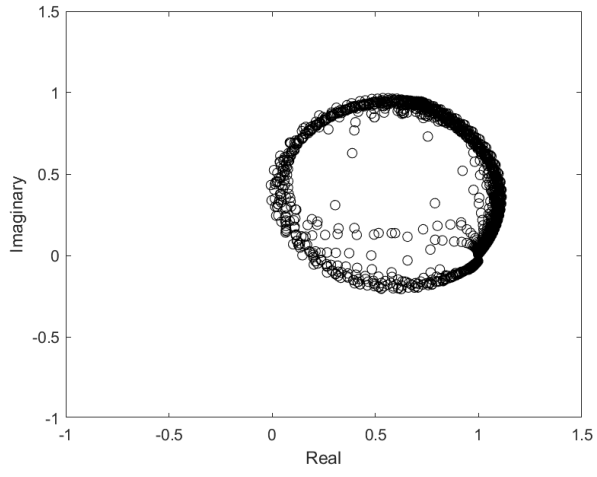


Figure 3.11:  $\epsilon = 0.1$  일 때  $M_{SL}^{-1}$ 을  $IC(k)$ 으로 계산하였을 때의 고유값 분포



### 3.3 Comparison between IC( $k$ ) and ICT( $p$ ) with Shifted Laplace preconditioner

IC( $k$ )와 ICT( $p$ )를 비교하기 위해 각 방법들이 최적의 수렴 결과를 낼 수 있는 각각의 계수  $k$ ,  $p$ ,  $\epsilon$ 를 찾는 실험을 진행하였다.

2D 헬름홀츠 방정식을 물리적 영역의 크기가 4km x 4km 인 정사각 균질 매질 문제를 풀었다. 수치적이 조건은 다음 표와 같다. 여기서  $kh = \frac{2\pi f}{c}h$  ( $G = \frac{c}{fh}$ )에서  $f = 7.5\text{hz}$ ,  $c = 1.5\text{km/s}$ 로 고정하고  $h$ 를 변경하면서  $kh$ 를 조정하였다.

kh (G)	nx	nz
0.314159(20)	400	400
0.408407(15)	300	300
0.628319(10)	200	200
1.25664(5)	100	100

Table 3.6: IC( $k$ )와 ICT( $p$ )의 비교실험 조건

IC( $k$ )의 실험은 4가지  $kh$ 에서 시프트  $\epsilon$ 과 레벨 계수  $k$ 를 바꿔가며 실험을 진행하였고, ICT( $p$ )의 실험은 4가지  $kh$ 에서 시프트  $\epsilon$ 과 탈락 기준  $p$ 를 바꿔가며 실험을 진행하였다. 각각의 실험에서 반복법의 반복 횟수, 총 계산 시간 불완전 분해를 위해 필요한 저장 공간등을 기록하였다.

표 3.7 ~ 3.14은 IC( $k$ )와 ICT( $p$ )의  $kh$ 에 따른 실험에서의 반복 횟수를 기록한 결과들이다.

		IC( $k$ )			
		0	1	2	3
shift $\epsilon$	0.03	-	-	-	-
	0.05	-	-	-	26
	0.07	-	-	-	27
	0.09	-	-	-	35
	0.11	-	-	-	40
	0.13	-	-	-	51
	0.15	-	-	-	49
	0.17	-	-	-	68
	0.19	-	-	-	-
	0.21	-	-	-	-

Table 3.7: IC( $k$ ), kh=0.314159(G=20)일때 수렴까지 소요되는 반복 횟수

		IC( $k$ )			
		0	1	2	3
shift $\epsilon$	0.03	-	-	-	42
	0.05	-	-	-	31
	0.07	-	-	-	28
	0.09	-	-	174	30
	0.11	-	-	80	31
	0.13	-	-	67	36
	0.15	-	-	72	40
	0.17	-	-	82	45
	0.19	-	-	85	50
	0.21	-	-	90	84

Table 3.8: IC( $k$ ), kh=0.408407(G=15)일때 수렴까지 소요되는 반복 횟수

		IC( $k$ )			
		0	1	2	3
shift $\epsilon$	0.03	-	-	-	15
	0.05	-	-	88	12
	0.07	-	-	35	14
	0.09	-	-	27	16
	0.11	-	-	32	18
	0.13	-	-	30	21
	0.15	-	-	31	22
	0.17	-	148	33	23
	0.19	-	106	42	27
	0.21	-	-	43	31

Table 3.9: IC( $k$ ), kh=0.628319(G=10)일때 수렴까지 소요되는 반복 횟수

		IC( $k$ )			
		0	1	2	3
shift $\epsilon$	0.03	-	51	15	9
	0.05	-	39	14	8
	0.07	-	35	12	9
	0.09	-	30	13	11
	0.11	-	28	12	12
	0.13	103	29	13	12
	0.15	61	26	14	13
	0.17	49	27	16	14
	0.19	40	28	16	15
	0.21	35	28	17	17

Table 3.10: IC( $k$ ), kh=1.25664(G=5)일때 수렴까지 소요되는 반복 횟수

		Drop factor p					
		0	5	10	15	20	25
shift $\epsilon$	0.03	-	-	-	-	39	26
	0.05	-	-	-	28	29	23
	0.07	-	-	-	26	30	23
	0.09	-	-	52	32	36	29
	0.11	-	-	52	41	41	34
	0.13	-	-	64	47	49	37
	0.15	-	-	64	51	73	54
	0.17	-	-	64	-	-	65
	0.19	-	-	61	-	-	-
	0.21	-	-	-	-	-	-

Table 3.11:  $ICT(p)$ ,  $kh=0.314159(G=20)$ 일때 수렴까지 소요되는 반복 횟수

		Drop factor p					
		0	5	10	15	20	25
shift $\epsilon$	0.03	-	-	-	50	23	13
	0.05	-	-	-	30	20	15
	0.07	-	-	25	28	22	17
	0.09	-	-	26	29	24	21
	0.11	-	-	34	30	27	23
	0.13	-	-	36	30	30	27
	0.15	-	-	44	32	36	30
	0.17	-	-	45	43	39	40
	0.19	-	-	50	61	43	45
	0.21	-	-	59	82	51	49

Table 3.12:  $ICT(p)$ ,  $kh=0.408407(G=15)$ 일때 수렴까지 소요되는 반복 횟수

		Drop factor p					
		0	5	10	15	20	25
shift $\epsilon$	0.03	-	-	44	15	14	8
	0.05	-	-	32	13	13	9
	0.07	-	-	25	15	13	12
	0.09	-	30	25	17	16	14
	0.11	-	25	26	18	17	16
	0.13	-	26	30	20	20	20
	0.15	-	28	33	22	21	20
	0.17	-	32	32	25	23	22
	0.19	-	34	34	26	25	25
	0.21	-	39	37	29	32	27

Table 3.13: ICT(p), kh=0.628319(G=10)일때 수렴까지 소요되는 반복 횟수



		Drop factor p					
		0	5	10	15	20	25
shift $\epsilon$	0.03	-	48	12	8	6	6
	0.05	-	22	11	8	8	8
	0.07	-	17	10	10	8	8
	0.09	-	15	11	10	11	9
	0.11	-	15	11	11	11	10
	0.13	76	16	13	14	12	11
	0.15	58	16	13	14	13	13
	0.17	43	17	16	15	15	13
	0.19	51	18	16	17	15	15
	0.21	40	19	17	17	17	17

Table 3.14: ICT(p), kh=1.25664(G=5)일때 수렴까지 소요되는 반복 횟수

kh(G)	$n^2$	ICT(p)			IC(k)		
		Iteration	Time	Fill-in factor	Iteration	Time	Fill-in factor
0.314159(20)	$400^2$	26	1.04	3	20	3.53	28.5
0.408407(15)	$300^2$	25	0.46	3	67	1.80	8.1
0.628319(10)	$200^2$	13	0.13	3	27	0.31	8.1
1.25664(5)	$100^2$	8	0.016	3	26	0.024	2.6

Table 3.15:  $4km^2$  물리 영역에서 ICT(p)와 IC(k)의 kh 따른 최소 시간 결과

표 3.15는 실험을 통해 얻은 ICT(p)와 IC(k) 각각 최적의 결과들을 나타낸 것이다. 여기서 채움 요소(Fill-in factor)는 원 행렬의 비영인자 수로 불완전 분해로 생성된 행렬의 비영인자 수를 나눈 값이다.

각각 최적의 결과일때 항상 ICT(p)가 우수하다는 것을 알 수 있다. 저장공간의 경우 근소한 차이로  $kh = 1.25664$ 일 때 IC(k)가 더 적게 사용하는 경우가 있다. 하지만 이 때에도 ICT(p)의 계산 시간은 더 적게 걸리는 것을 알 수 있다. ICT(p)는 kh에 따라 최적의 결과가 나올 때 필요한 저장공간 요소가 일정하다. 하지만 IC(k)의 경우 문제의 영역이 커질수록 그에 비례해 필요한 저장공간도 같이 늘어났다. 불완전 분해를 하기 위해 필요한 저장공간의 크기의 차이는 최대 9.5배 까지 났다. 불완전 분해에 더 많은 채움을 사용하기 때문에 그만큼 연산량이 늘어 반복 횟수가 더 적은 경우에도 계산 시간이 더 걸리는 것도 확인할 수 있다.

# Chapter 4

## Numerical Examples

### 4.1 Parameter selection of preconditioner

수렴 속도에 영향을 주는 주요한 요소들은  $kh$ , 수치영역  $N$ , 탈락 기준  $p$ , 시프트  $\epsilon$  등이 있다.  $kh$ 와 수치영역  $N$ 은 풀기 위한 시스템을 구성하는 요소들이고, 이 요소들에 따라 최적의 수렴 속도를 내는  $p$ 와  $\epsilon$ 의 값들이 바뀌게 된다. 이를 확인하기 위해 수치 영역  $N$ 을 기준으로  $p$ 와  $\epsilon$ 의 경향성을 파악하기 위한 실험과  $kh$ 를 기준으로  $p$ 와  $\epsilon$ 의 경향성을 파악하는 실험을 진행하였다.

#### 4.1.1 Numerical domain size dependency

그림 4.1에서 4.6은 2차원에서 수치영역의 크기와  $p$ 와  $\epsilon$  변화에 따른 수렴 경향에 대해 밀도 함수의 형태로 나타낸 것이고, 그림 4.7에서 4.12은 3차원에서 수치영역의 크기와  $p$ 와  $\epsilon$  변화에 따른 수렴 경향에 대해 밀도 함수의 형태로 나타낸 것이다. 파란색에 가까울수록 수렴을 잘하는 것이고 빨간색에 가까울수록 발산을 하거나 조건 이내에 수렴하지 못한 것을 나타낸다. 이 그림들을 보면 2차원의 경우 수치영

역의 크기가 커질수록 잘 수렴하는  $p$ 와  $\epsilon$ 의 범위가 크게 줄어드는 것을 확인할 수 있다. 하지만 3차원의 경우 약간의 경향의 차이는 있지만 거의 일정한 것을 확인할 수 있다.

그림 4.13과 4.14는 각각 2차원과 3차원에서  $p$ 에 따라 최적의 결과를 낼때의  $\epsilon$ 을 나타낸 것이다. 2차원과 3차원 모두  $p$ 의 값이 커짐에 따라  $\epsilon$ 은 작은 값을 사용할 때 좋은 수렴 경향을 나타내는 것을 알 수 있다. 2차원의 경우 작은  $p$ 에서는 수렴하지 않는 경우도 포함되어  $\epsilon$ 의 값이 수치영역의 크기에 따라 큰 차이를 보여주지만 수렴을 잘 하는  $p$ 의 값들의 경우에는 좋은 결과를 내는  $\epsilon$ 의 값들이 비슷한 값을 가지는 것을 확인할 수 있다. 3차원의 경우 2차원 보다  $\epsilon$ 의 값들이 영역의 크기에 따라 변화가 조금 더 크지만 위의 밀도함수를 확인해도 알 수 있듯이 최적의  $\epsilon$ 의 값 주변에서도 수렴 경향은 크게 차이 나지 않는 것을 알 수 있다. 이를 통해 각  $p$ 에 따른  $\epsilon$ 값들의 평균값을 구하여 그림 4.13과 4.13의 오른쪽 그림에 나와있는  $p$ 값에 따라 일반적으로 사용할 수 있는  $\epsilon$ 값을 구하였다.

표 4.1와 4.2는 각각 그림 4.1부터 4.12에서 나타낸 밀도함수에서 2차원과 3차원 결과에서 가장 빠른 수렴속도를 보여준  $p$ 와  $\epsilon$ 의 값과 이 때의 반복 횟수와 시간을 나타낸 것이다. 앞서 그림 4.1부터 4.12에서 나타났듯이 2차원의 경우 수치영역의 크기가 커질수록 큰  $p$ 를 사용해야 하고 이에 맞춰  $\epsilon$ 은 작은 값을 사용하였을 때 최적의 수렴속도를 얻을 수 있었다. 밀도함수에서 확인하였듯이 3차원의 경우 수치영역의 크기가 커지더라도 일정한  $p$ 의 사용이 가능한 것을 확인하였고 이에 맞는  $\epsilon$ 은 약간의 차이는 있지만 2차원 문제보다는 일정한  $\epsilon$ 을 사용하고 최적의 수렴속도를 얻을 수 있었다. 하지만 3차원 문제의 경우에도 보다 큰 수치영역을 풀거나 불균질 모델등 수치적으로 더 풀기 어려운 문제의 경우 안정성을 위해 더

큰 p값이 필요할 것으로 보인다.

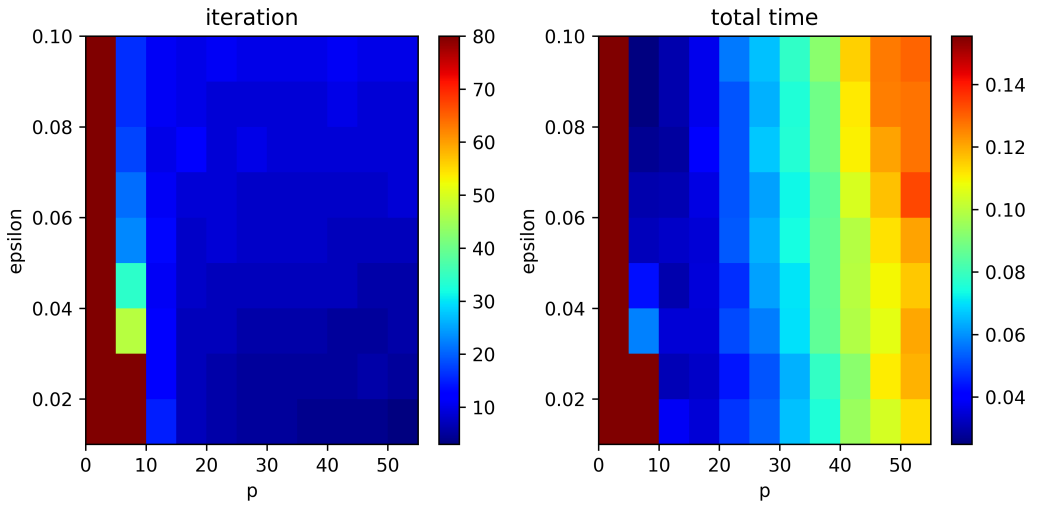


Figure 4.1: 2차원 100×100일 때,  $p$ 와  $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간

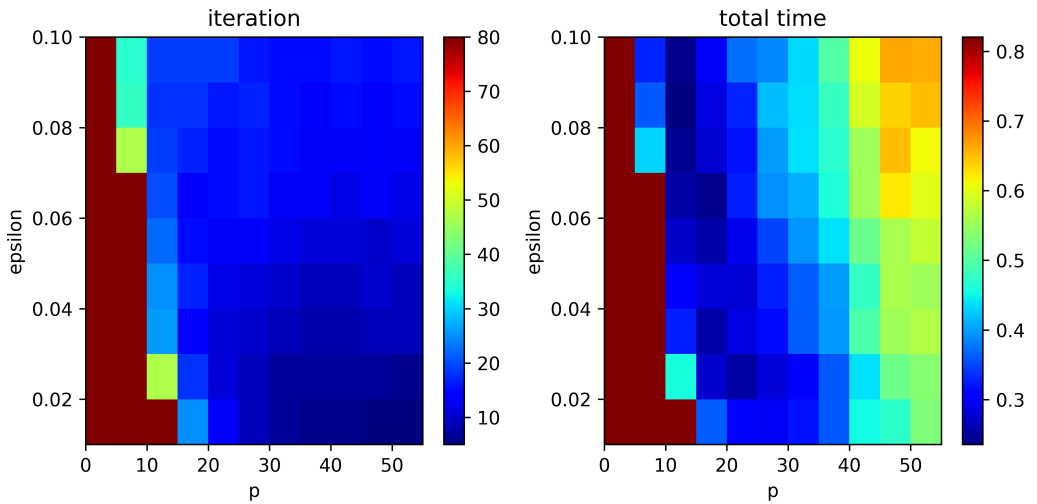


Figure 4.2: 2차원 200×200일 때,  $p$ 와  $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간

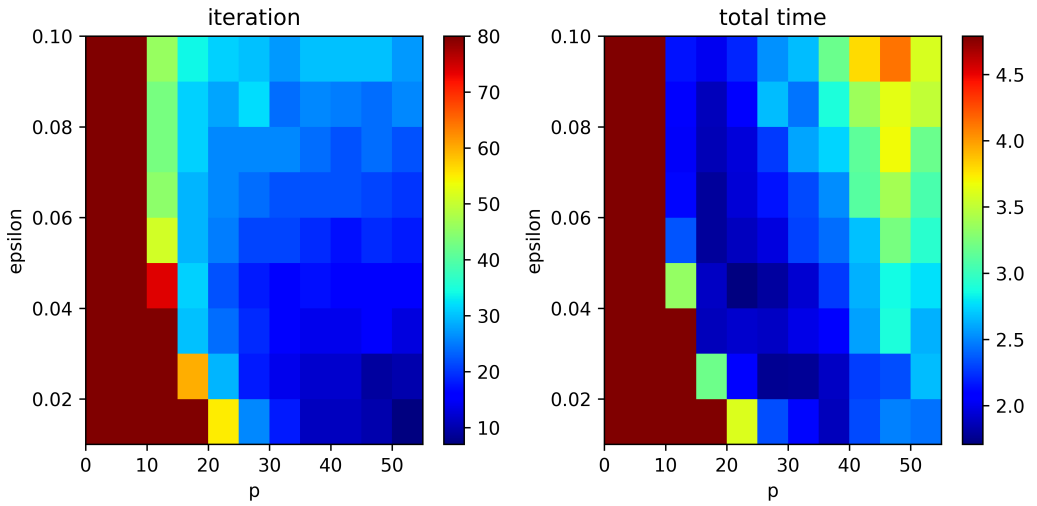


Figure 4.3: 2차원  $400 \times 400$ 일 때,  $p$ 와  $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간

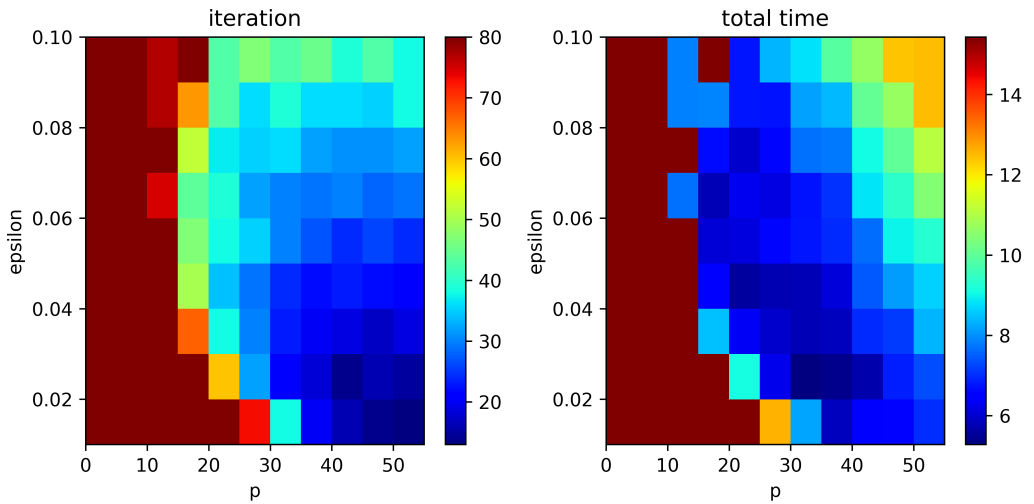


Figure 4.4: 2차원  $600 \times 600$ 일 때,  $p$ 와  $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간



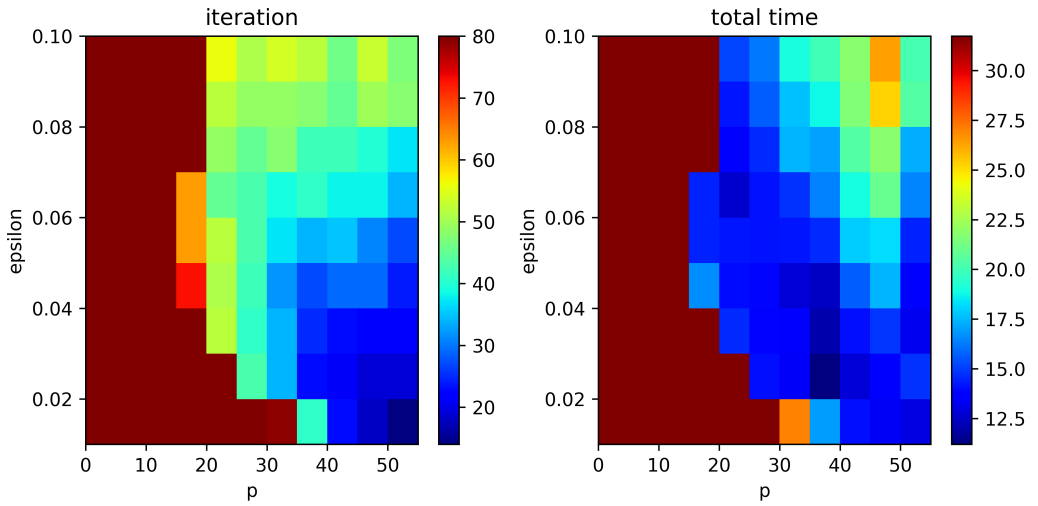


Figure 4.5: 2차원 800×800일 때,  $p$ 와  $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간

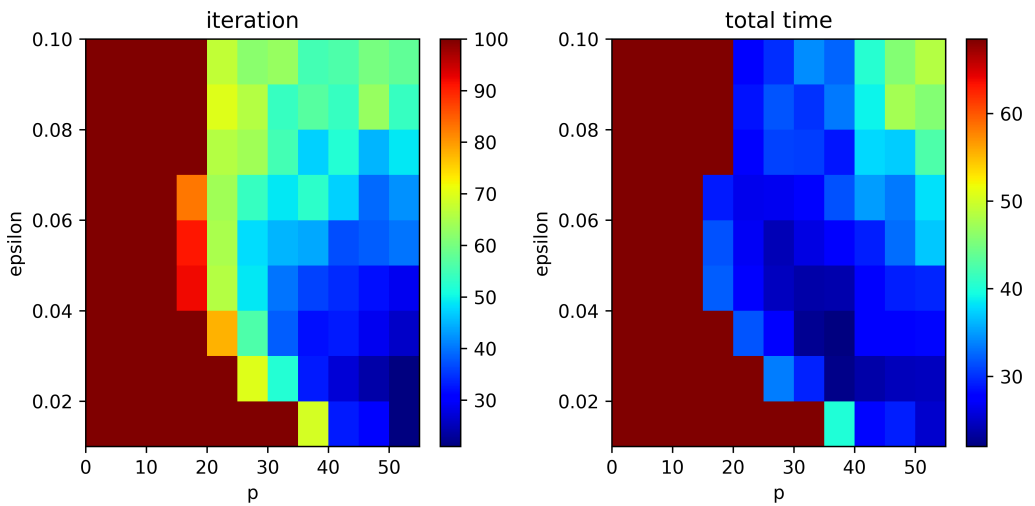


Figure 4.6: 2차원 1000×1000일 때,  $p$ 와  $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간

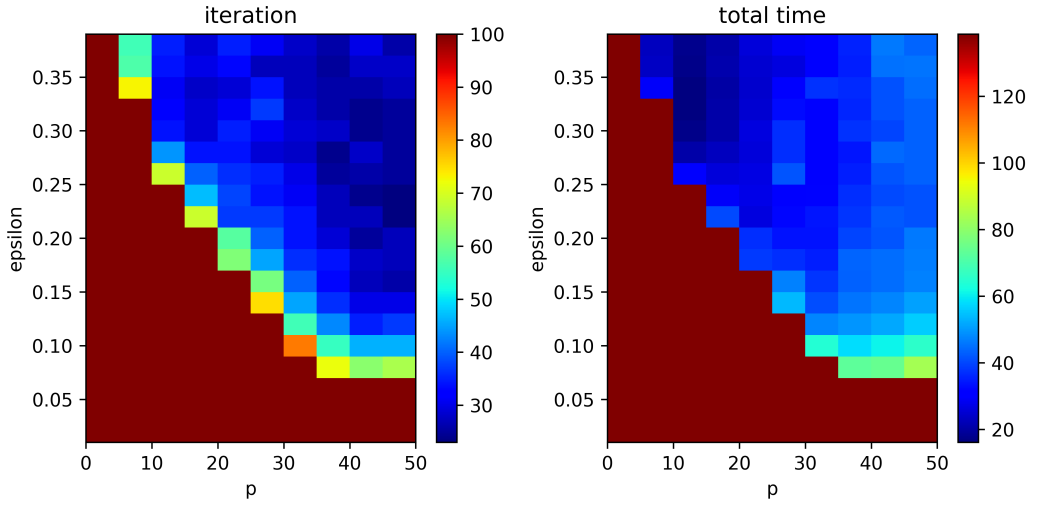


Figure 4.7: 3차원  $100 \times 100 \times 100$ 일 때,  $p$ 와  $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간

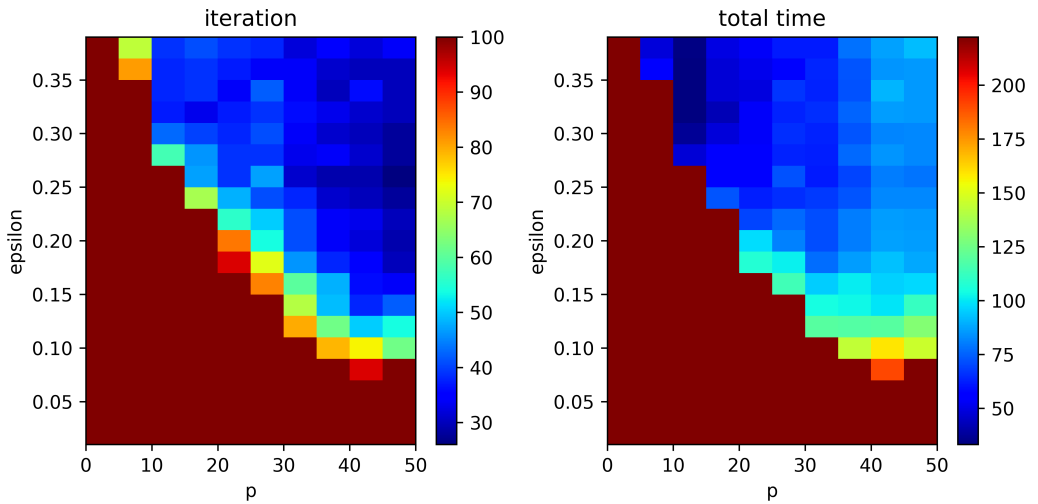


Figure 4.8: 3차원  $120 \times 120 \times 120$ 일 때,  $p$ 와  $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간

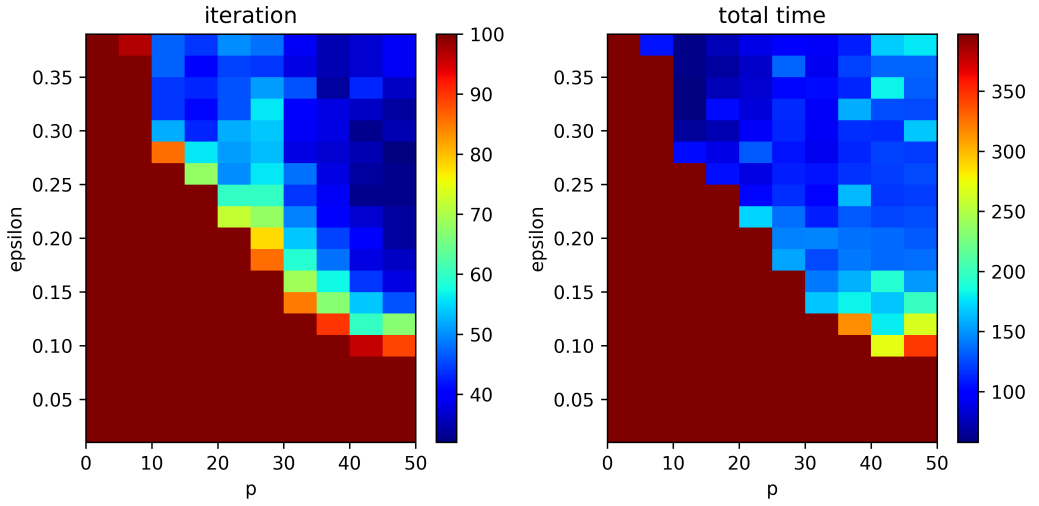


Figure 4.9: 3차원  $140 \times 140 \times 140$ 일 때,  $p$ 와  $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간

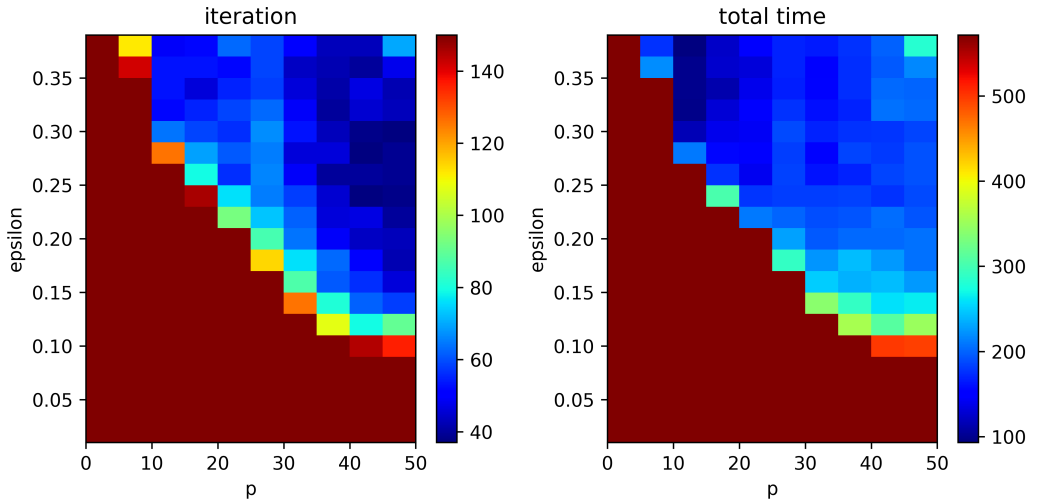


Figure 4.10: 3차원  $160 \times 160 \times 160$ 일 때,  $p$ 와  $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간

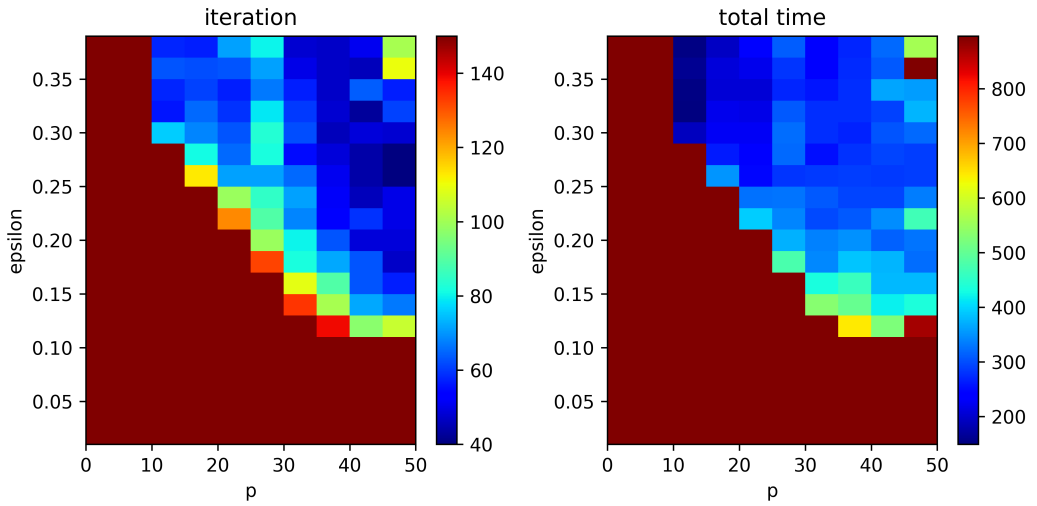


Figure 4.11: 3차원  $180 \times 180 \times 180$ 일 때,  $p$ 와  $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간

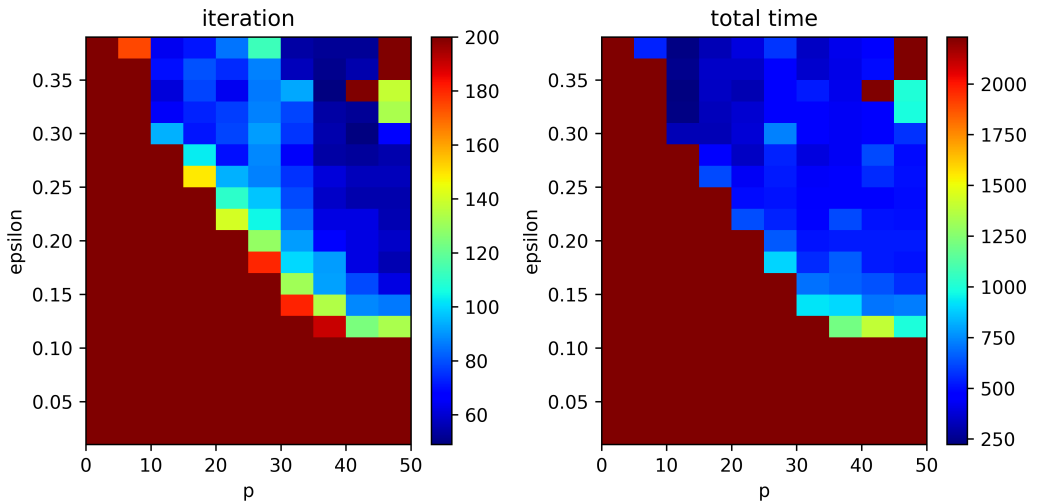


Figure 4.12: 3차원  $200 \times 200 \times 200$ 일 때,  $p$ 와  $\epsilon$ 에 따른 반복법의 수렴까지 반복 횟수와 그때 소요된 총 시간

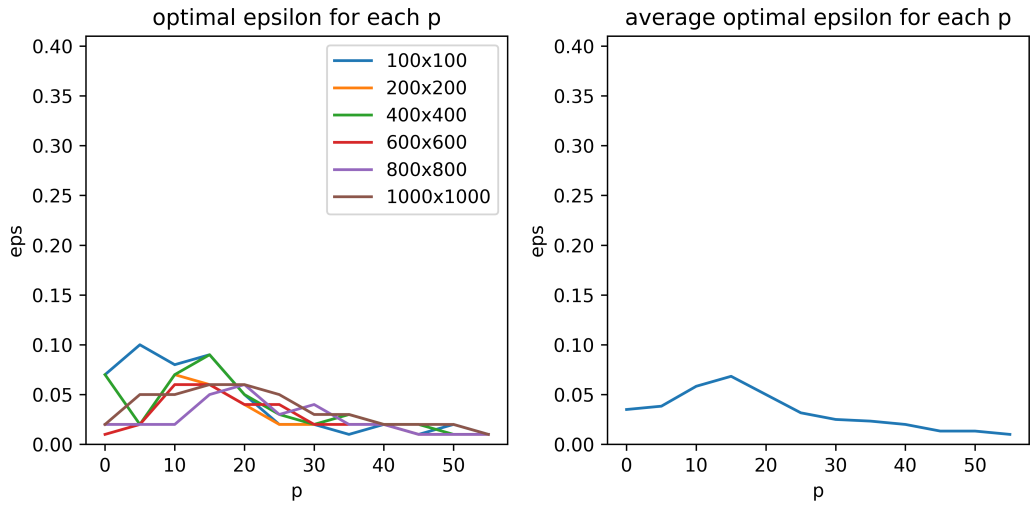


Figure 4.13: 2차원일 때, 수치영역 크기에 따른  $p$ 와  $\epsilon$ 과 평균값

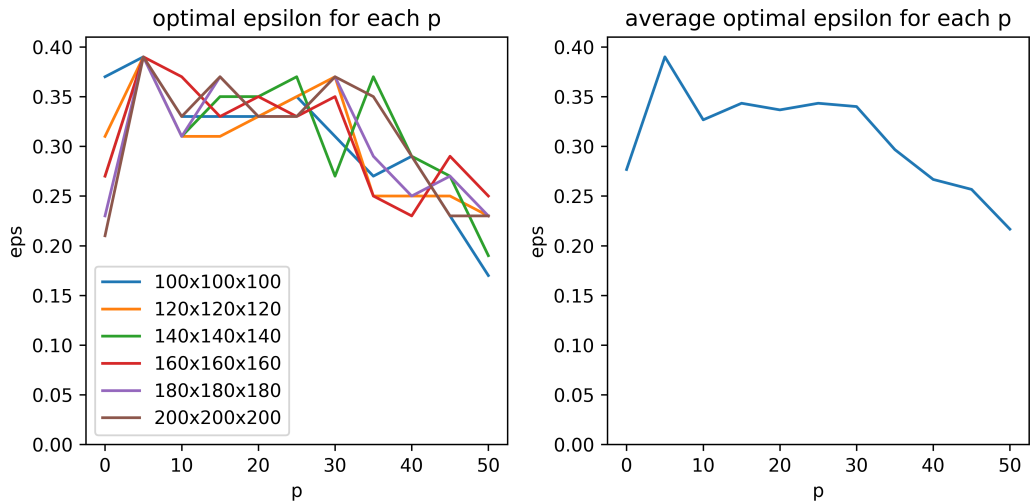


Figure 4.14: 3차원일 때, 수치영역 크기에 따른  $p$ 와  $\epsilon$ 과 평균값

ICT(p)					
domain size	N	p	$\epsilon$	iteration	time(sec)
100×100	10000	5	0.09	16	0.018
200×200	40000	10	0.08	18	0.186
400×400	160000	20	0.04	22	1.30
600×600	360000	30	0.02	21	3.581
800×800	640000	35	0.02	23	7.361
1000×1000	1000000	35	0.03	32	15.915

Table 4.1:  $kh = 1.25664$ , 2차원 문제에서 수치영역 별 최적의  $p$ 와  $\epsilon$ , 이때의 반복 횟수와 소요 시간

ICT(p)					
domain size	N	p	$\epsilon$	iteration	time(sec)
100×100×100	1000000	10	0.33	31	11.71
120×120×120	1728000	10	0.31	37	25.64
140×140×140	2744000	10	0.31	44	46.14
160×160×160	4096000	10	0.37	49	76
180×180×180	5832000	10	0.31	56	124.37
200×200×200	8000000	10	0.33	61	188.53

Table 4.2:  $kh = 1.25664$ , 3차원 문제에서 수치영역 별 최적의  $p$ 와  $\epsilon$ , 이때의 반복 횟수와 소요 시간

### 4.1.2 kh dependency

kh에 따른 경향을 확인하기 위해 수치영역, p와 각 p에서 최적의 결과가 나오는  $\epsilon$ 에 따른 실험을 진행하였다. 여기서 사용된 수치영역은 2차원  $100 \times 100 \sim 1000 \times 1000$  까지 실험을 진행하였다. 그림 4.15에서 4.20에서 그 결과를 나타내었다. 각 결과를 보면 수치영역이 같을 경우 kh에 변화에 따른 수렴 경향이 유사한 것을 확인할 수 있다. 하지만 4.19과 4.20에서는 작은 p를 사용하였을 때 kh에 따라 수렴 경향이 약간 차이가 나는 것을 볼 수 있다. 이 경우에도 일정이상의 p를 사용할 경우 kh의 변화에도 비슷한 수렴 경향을 보여주었다. 그렇기 때문에 다양한 kh에서 일정한  $\epsilon$ 을 사용하면 안정적인 수렴 경향을 얻기 위해서는 충분히 큰 p를 사용하는 것으로 시간과 저장공간에서 더 비용을 사용하는 것을 고려할 수 있다.

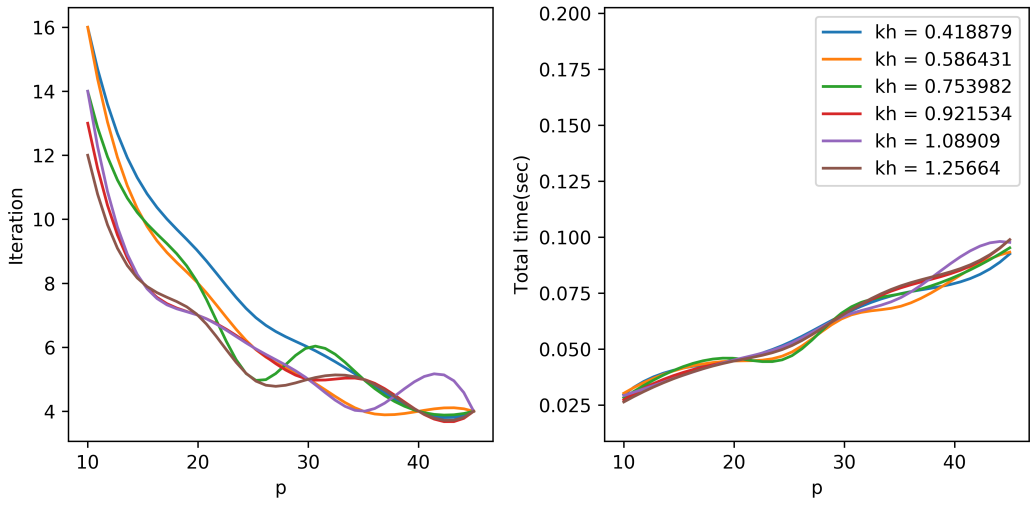


Figure 4.15: 100×100일 때, 각 kh에서 p에 따른 수렴까지 걸리는 반복 횟수와 총 시간

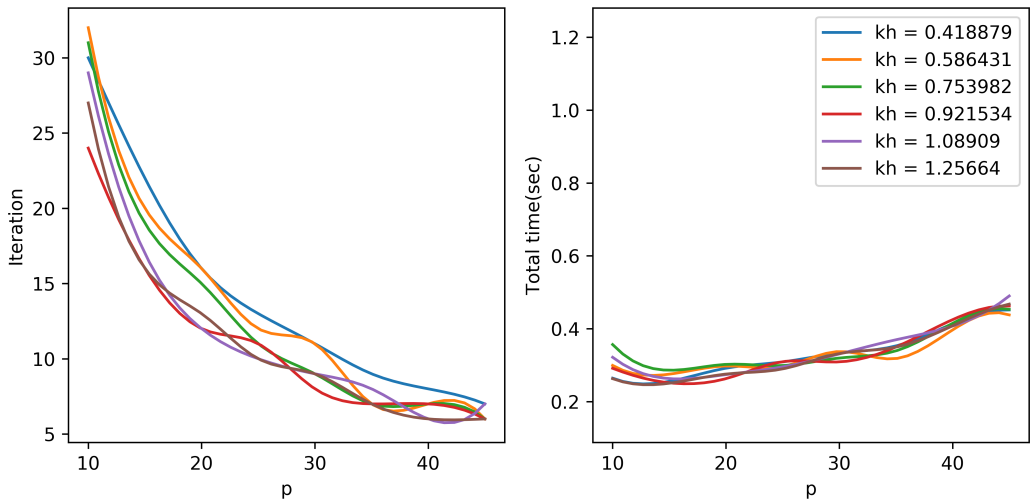


Figure 4.16: 200×200일 때, 각 kh에서 p에 따른 수렴까지 걸리는 반복 횟수와 총 시간



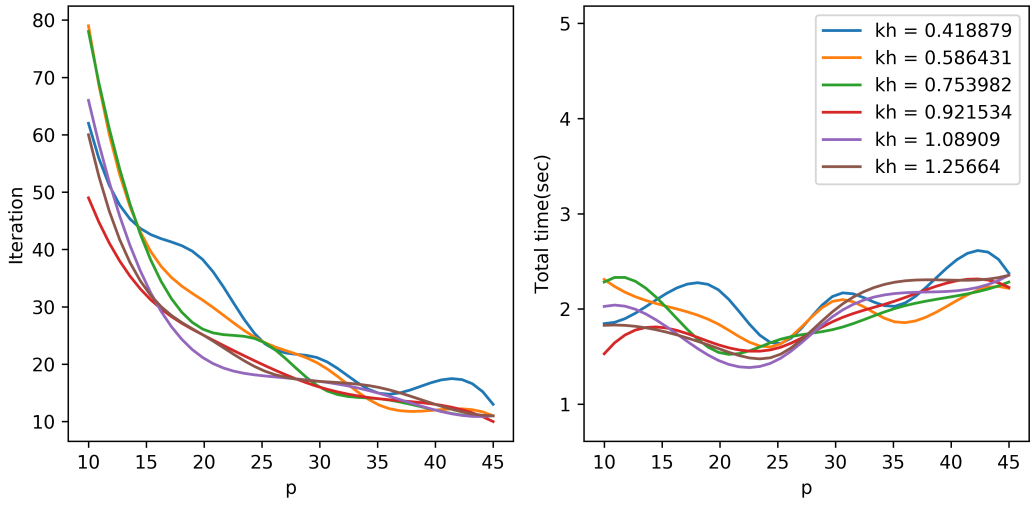


Figure 4.17:  $400 \times 400$ 일 때, 각  $kh$ 에서  $p$ 에 따른 수렴까지 걸리는 반복 횟수와 총 시간

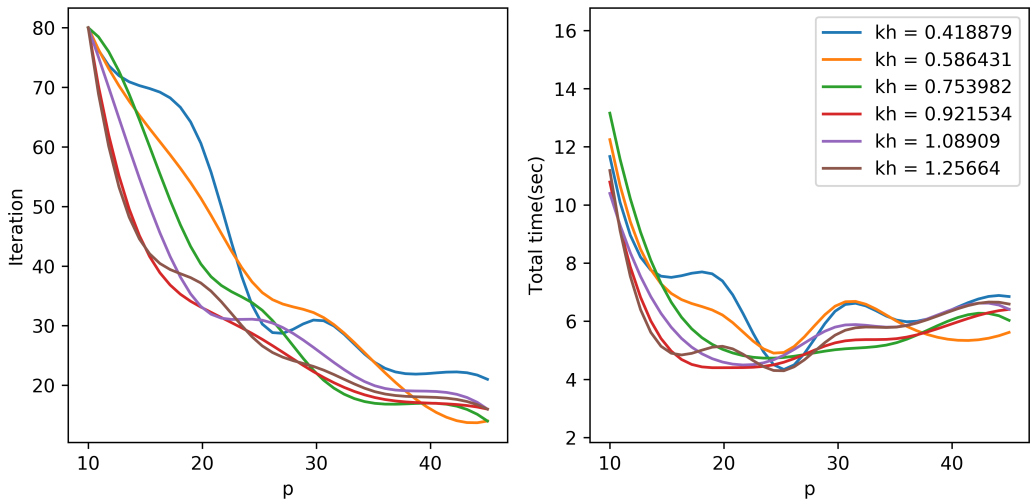


Figure 4.18:  $600 \times 600$ 일 때, 각  $kh$ 에서  $p$ 에 따른 수렴까지 걸리는 반복 횟수와 총 시간

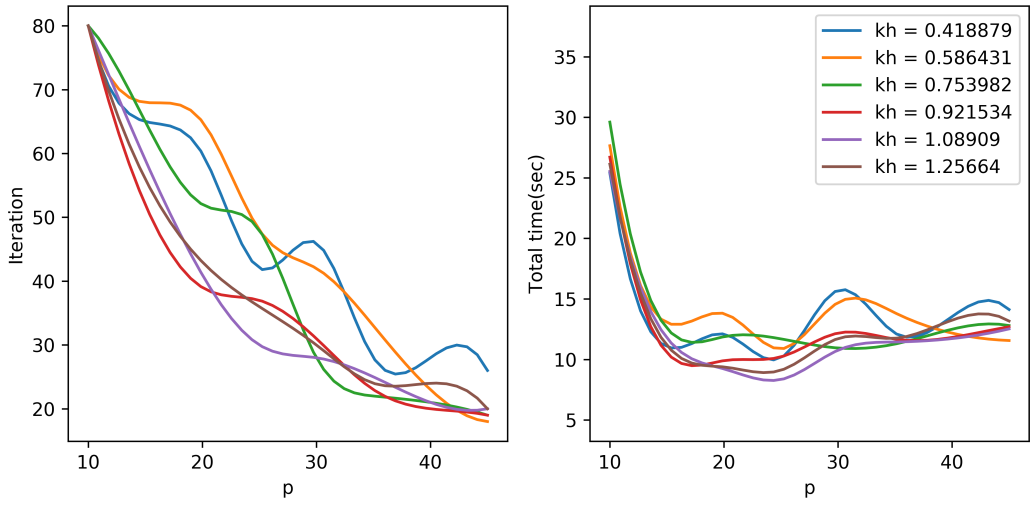


Figure 4.19:  $800 \times 800$ 일 때, 각 kh에서 p에 따른 수렴까지 걸리는 반복 횟수와 총 시간

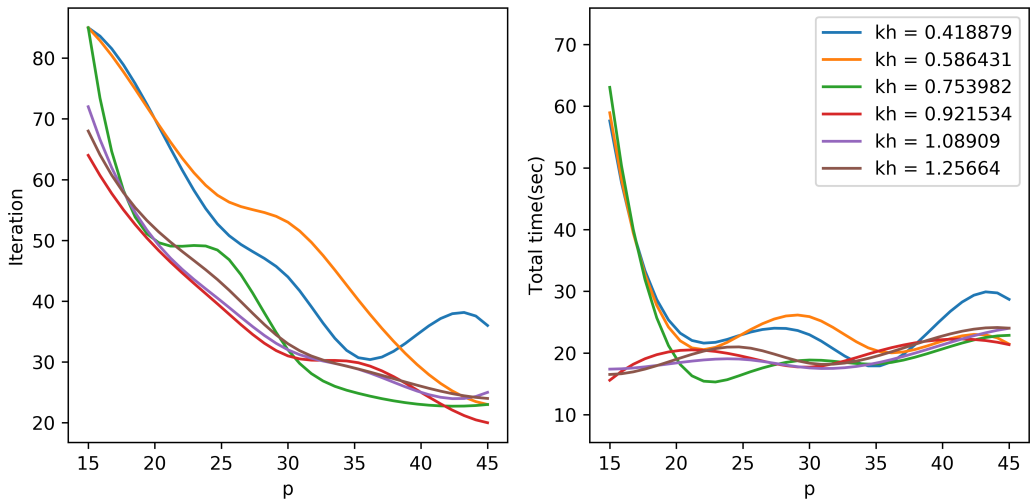


Figure 4.20:  $1000 \times 1000$ 일 때, 각 kh에서 p에 따른 수렴까지 걸리는 반복 횟수와 총 시간

### 4.1.3 Computational time

4.21에서 4.32은 2차원과 3차원 다양한 수치영역 크기에서의  $p$ 에 따른 수렴까지 소요되는 시간을 나타낸다. 불완전분해를 통해 SL preconditioner  $M_{SL}$ 을 구할 때 행렬의 비영 원소의 수는  $p$ 에 따라 증가하게 된다. 증가된 비영 원소수에 따라 불완전분해에 필요한 저장공간과 연산 시간이 늘어나게 된다. 이와 함께 반복법의 반복 횟수에 따른 비용도 증가하게 된다.  $p$ 의 값이 커질수록 불완전분해를 수행할 때 소요되는 시간이 증가하게 되어 일정 이상의 값을 사용할 경우 반복법에서 소요되는 시간보다 불완전분해에 소요되는 시간이 더 많이 걸리기 때문에 적절한  $p$ 를 사용해야 한다. 수치영역의 크기가 작을 경우  $p$ 가 일정이상 커지면 전체 소요시간에서 불완전분해를 수행하는 시간이 지배적이지만 수치영역의 크기가 커질수록  $p$ 가 커지더라도 불완전분해보다 반복법의 연산 시간이 지배적인 시간을 차지한다.

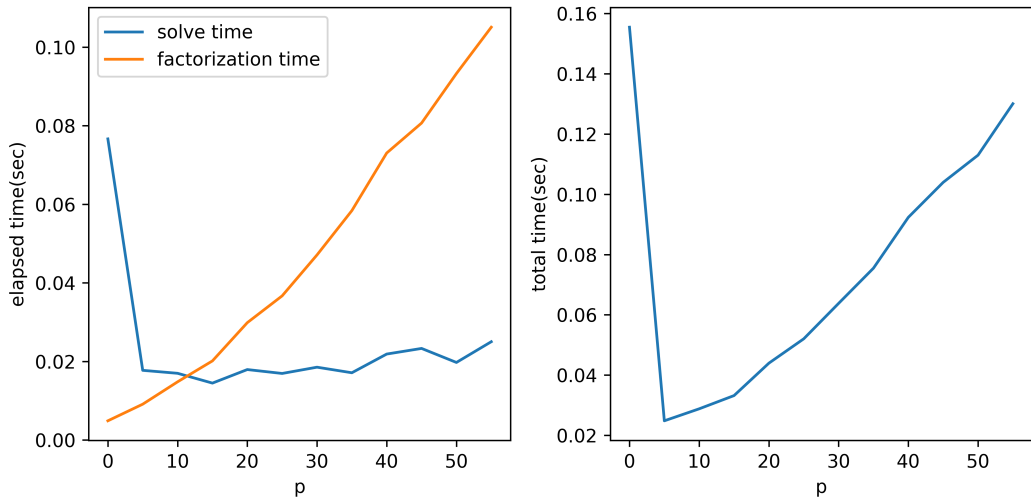


Figure 4.21: 2차원  $100 \times 100$ 일 때,  $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽)

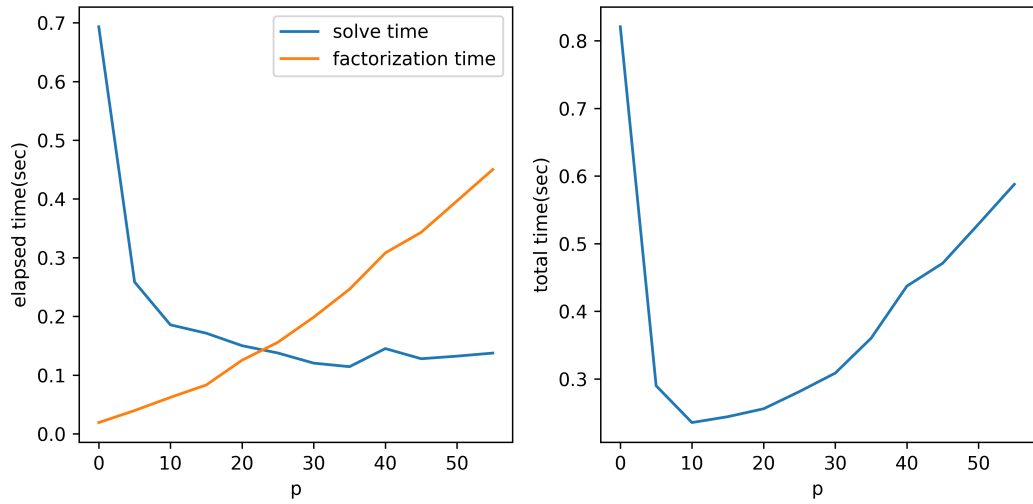


Figure 4.22: 2차원  $200 \times 200$ 일 때,  $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽)

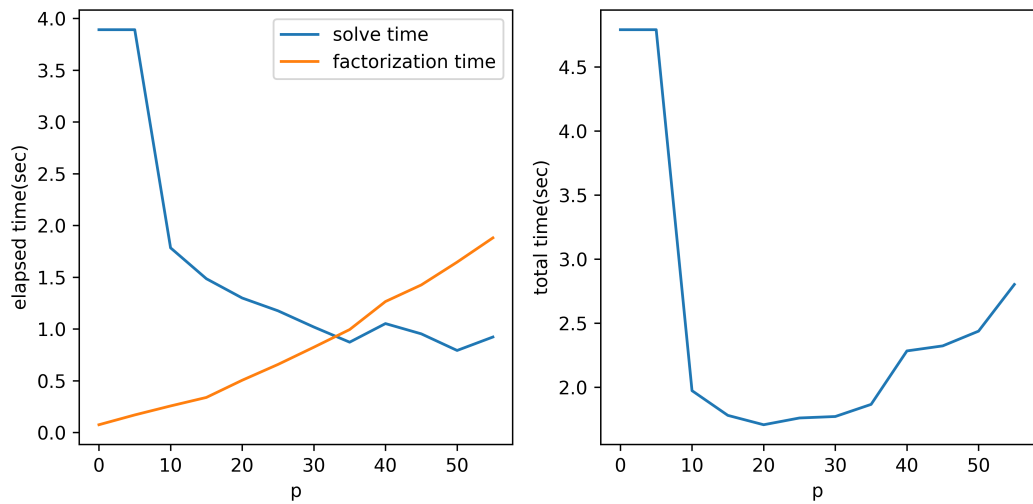


Figure 4.23: 2차원  $400 \times 400$ 일 때,  $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽)

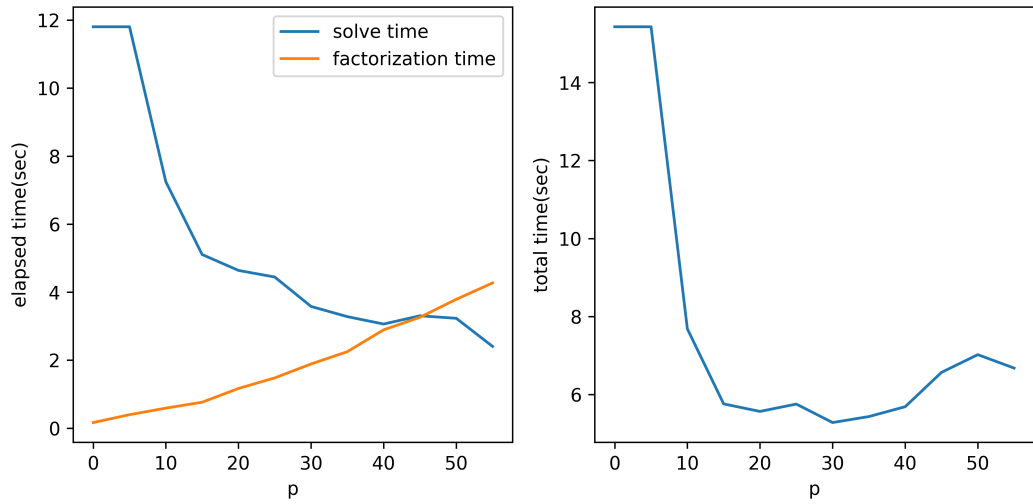


Figure 4.24: 2차원  $600 \times 600$ 일 때,  $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽)

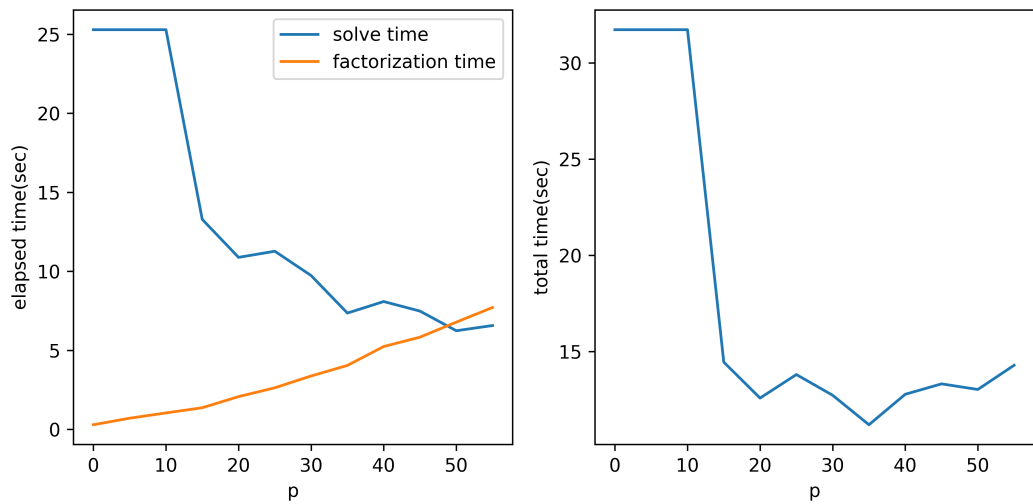


Figure 4.25: 2차원  $800 \times 800$ 일 때,  $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽)

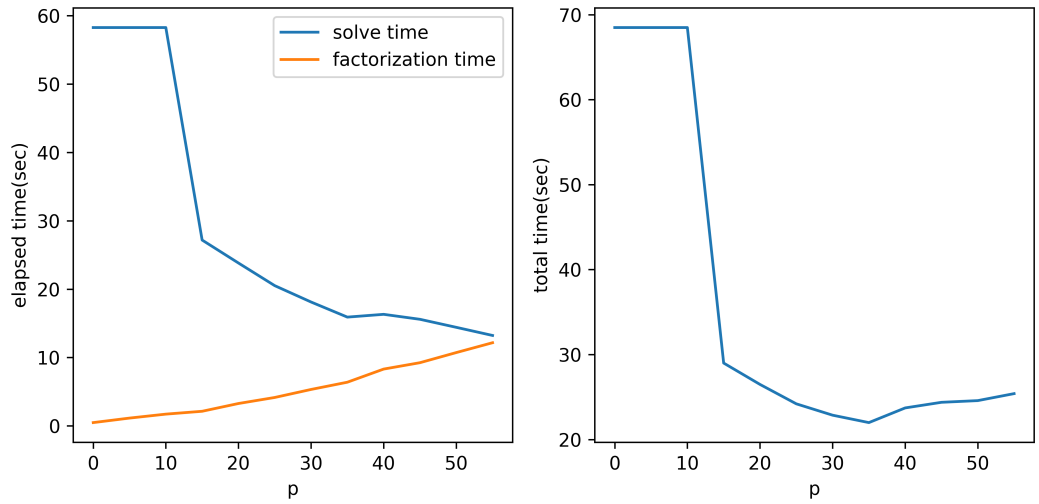


Figure 4.26: 2차원  $1000 \times 1000$ 일 때,  $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽)

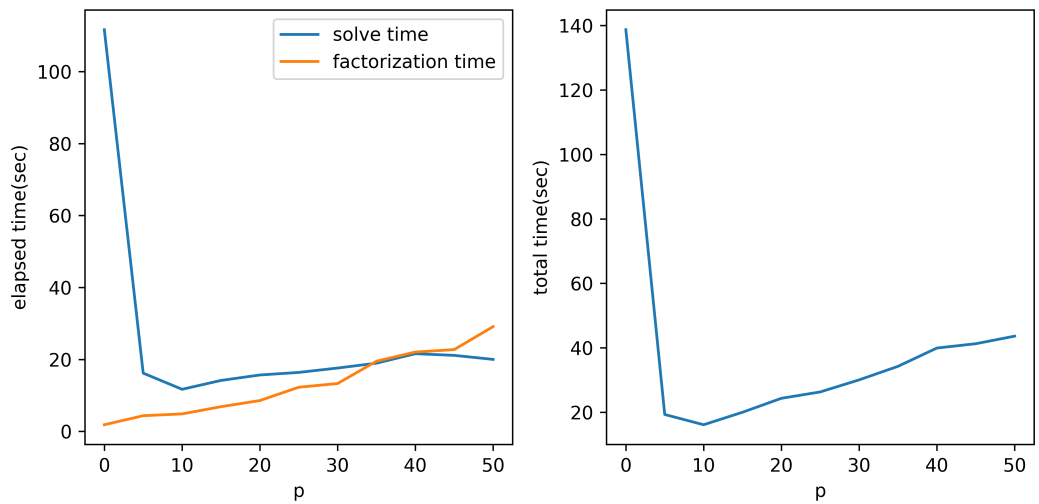


Figure 4.27: 3차원  $100 \times 100 \times 100$ 일 때,  $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽)

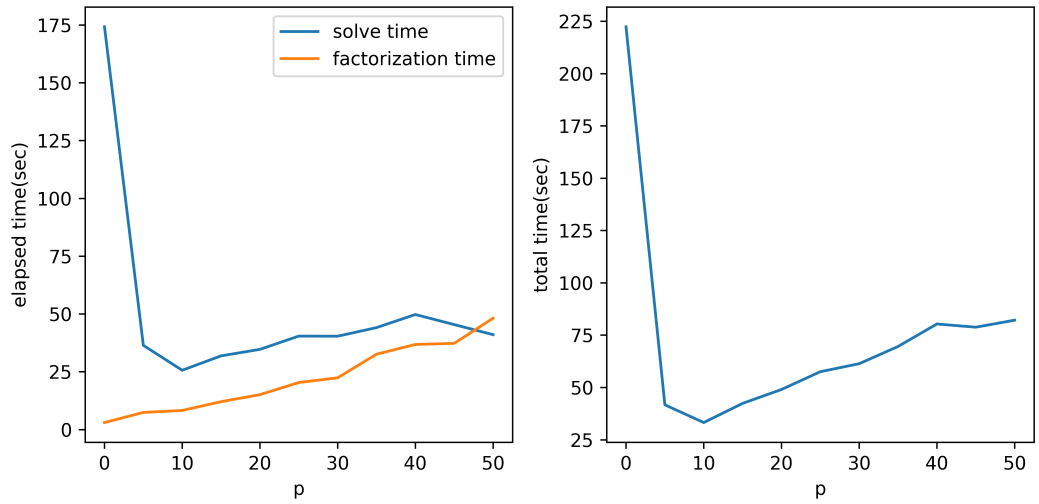


Figure 4.28: 3차원  $120 \times 120 \times 120$ 일 때,  $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽)

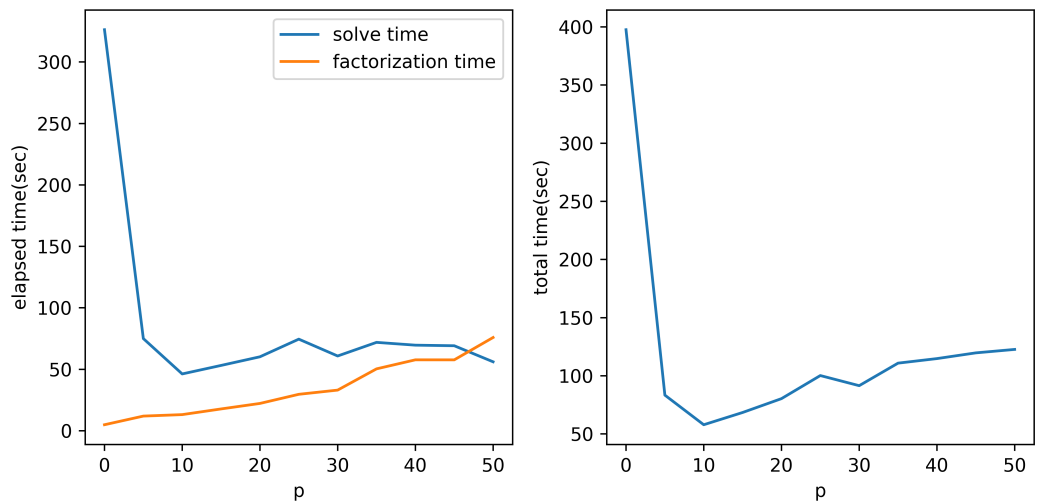


Figure 4.29: 3차원  $140 \times 140 \times 140$ 일 때,  $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽)

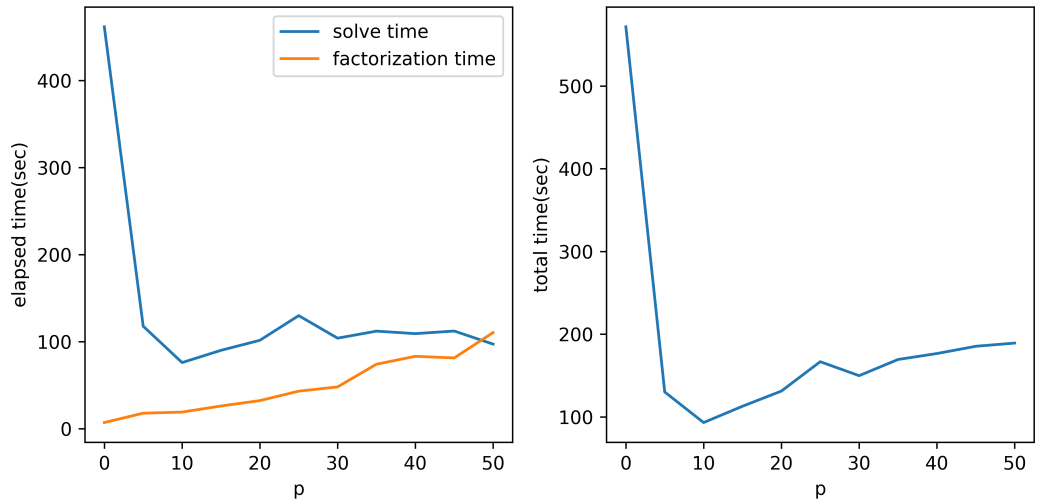


Figure 4.30: 3차원  $160 \times 160 \times 160$ 일 때,  $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽)

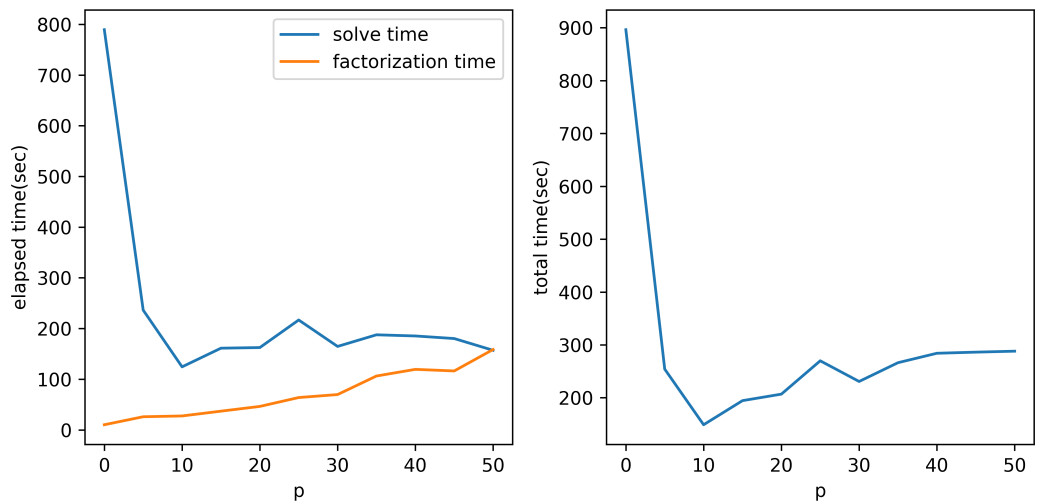


Figure 4.31: 3차원  $180 \times 180 \times 180$ 일 때,  $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽)



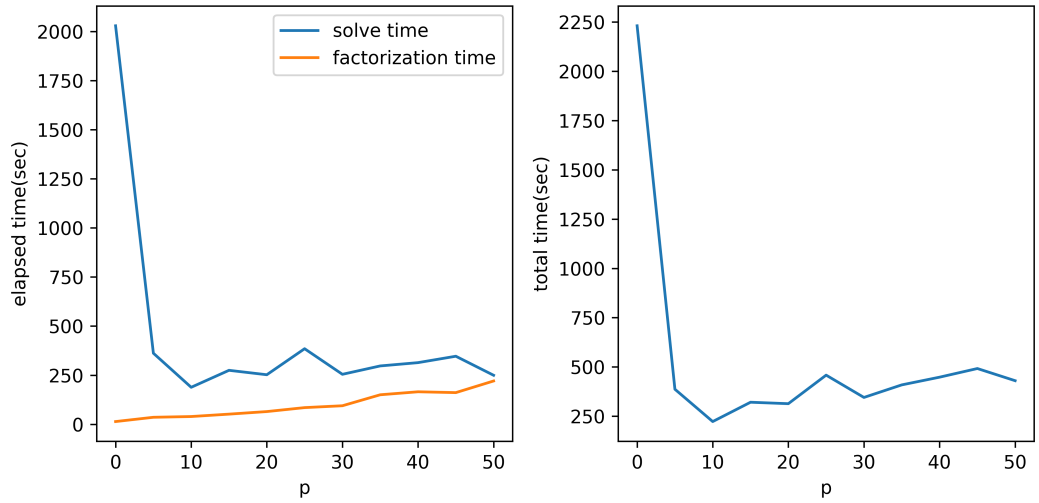


Figure 4.32: 3차원  $200 \times 200 \times 200$ 일 때,  $p$ 에 따른 수렴까지 반복법에 소요되는 시간과 불완전 분해에 소요되는 시간(왼쪽), 총 소요 시간(오른쪽)

### Complexity

SL preconditioner을 활용한 반복법의 이상적인 시간 복잡도와 저장공간 복잡도는 2차원일 때  $O(N^{3/2})$ ,  $O(N)$ , 3차원일 때  $O(N^{4/3})$ ,  $O(N)$ 이 나온다. 직접법의 경우 시간 복잡도와 저장공간 복잡도가 2차원 문제에서  $O(N^{3/2})$ ,  $O(N \log N)$  3차원 문제에서  $O(N^2)$ ,  $O(N^{4/3})$ 에 이르기 때문에 SL preconditioner을 활용한 방법이 직접법에 비해 계산 비용에서 매우 효율적인 방법이라는 것을 알 수 있다. 그림 4.33와 4.34는 위 3차원 실험에 대해 시간 복잡도와 저장공간 복잡도를 직접법과 SL preconditioner가 이상적일때를 비교한 그림이다. 시간 복잡도의 경우 실험값이 약간의 차이는 있지만 이상적인  $O(N^{4/3})$  복잡도를 만족하는 것을 확인할 수 있다. 저장공간 복잡도의 경우 이상적인 경우와 같은 복잡도를 갖는 것을 확인할 수 있다.

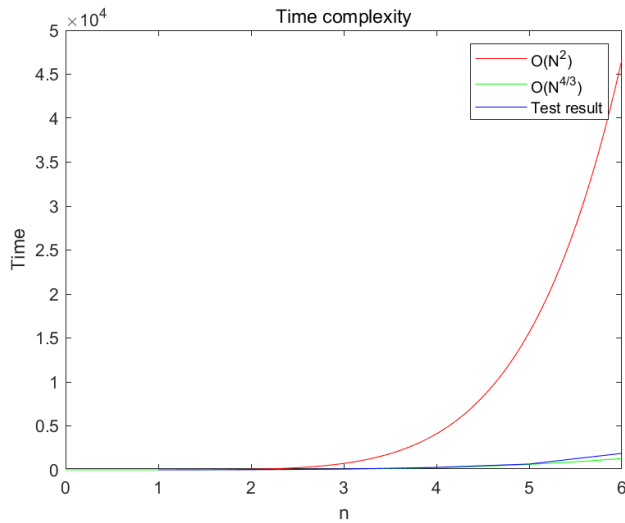


Figure 4.33: 3차원 실험 결과와 직접법과 이상적인 SL preconditioner 적용된 반복법의 시간 복잡도 비교

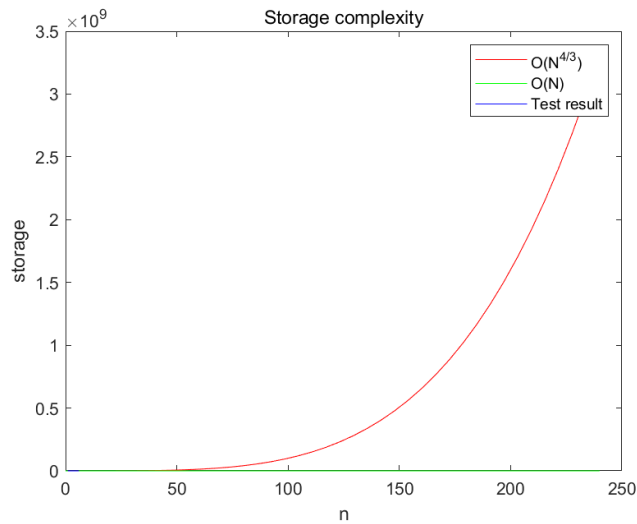


Figure 4.34: 3차원 실험 결과와 직접법과 이상적인 SL preconditioner 적용된 반복법의 저장공간 복잡도 비교

## 4.2 Heterogeneous model

지구물리 분야에서 검증자료로 많이 쓰이는 모델들로 불균질 매질에서의 성능을 실험하였다. 2차원 불균질 모델으로는 Marmousi 모델, Pluto 모델을 사용하였고, 3차원 불균질 매질의 실험으로는 SEG/EAGE 3D 모델을 사용하였다. 각각의 모델에 사용한  $p$ 와 시프트  $\epsilon$ 은 앞선 균질모델 실험에서 구한 수치영역 크기별 최적의 값을 사용하였다.

### 4.2.1 2D : Marmousi

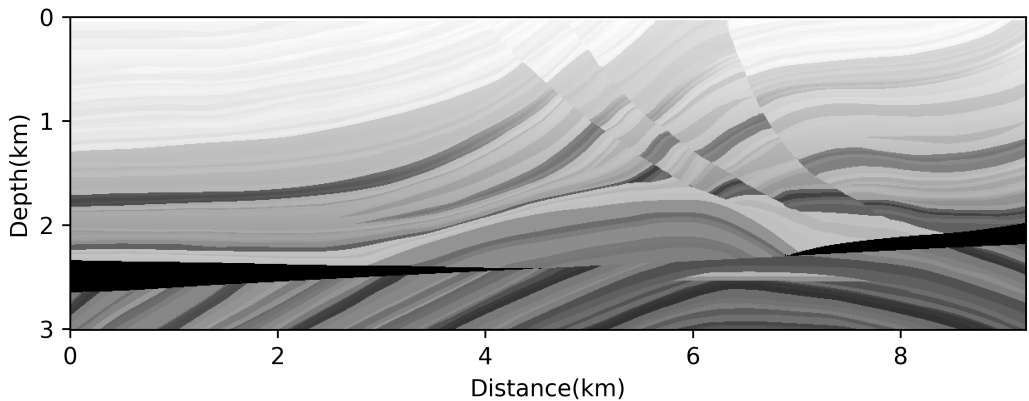


Figure 4.35: The Marmousi 속도 모델

사용한 Marmousi 모델은 직사각형에 물리영역  $9.2\text{km} \times 3\text{km}$ 이고 이를 수치영역  $230 \times 75$  노드,  $40\text{m}$  격자로 설정하였다. 경계조건으로 상부는 Dirichlet 경계조건, 나머지는 PML 경계조건을 사용하였다. 소스는 상부 경계 중심에서 바로 아래 노드에 위치하였다. 주파수는  $0\text{ Hz}$  부터  $7.5\text{ Hz}$  까지( $kh = 0$  부터  $kh = 1.25664$ 에 해당)  $0.25\text{ Hz}$  간격으로 실험하였다.

해의 정확도를 검증하기 위해 직접법으로 구한 해와 반복법으로 구한 해의 비교를 진행하였다. 그림 4.36 부터 그림 4.38는 각 kh에 따라 송신원 깊이에서의 해를 비교한 것이다. 비교를 위해 실험을 한 kh중 3개를 작은 값에서 부터 큰 값의 결과를 샘플링 하였다. 결과 이미지에서 파란 실선이 직접법의 해이고 빨간 점이 반복법의 해이다. 각각의 kh에서 직접법의 해와 반복법의 해가 일치하는 것을 볼 수 있다.

추가적으로 kh에 따른 결과를 종합하여 확인 할 수 있게 낮은 kh에서 부터 높은 kh까지의 결과를 종합한 시간영역 seismogram을 구하였다. 그림 4.39는 직접법으로 주파수 영역 실험을 통해 구한 seismogram이고, 4.40는 반복법을 사용하여 주파수 영역 실험을 통해 구한 seismogram이다. 육안으로 둘의 차이를 구분하기 힘들기 때문에 4.41로 두 결과의 차이를 이미지로 만들었다. Marmousi 모델의 경우 두 모델의 차이가 0으로 반복법과 직접법 계산상의 차이가 없는 것을 확인 할 수 있다.

kh	$n_x \times n_z$	iteration	Time
0.418879	230 × 75	8	0.043
0.837758	230 × 75	8	0.045
1.25664	230 × 75	9	0.048

Table 4.3: Marmousi 모델 각 kh별 반복 횟수와 소요 시간

## Frequency domain result

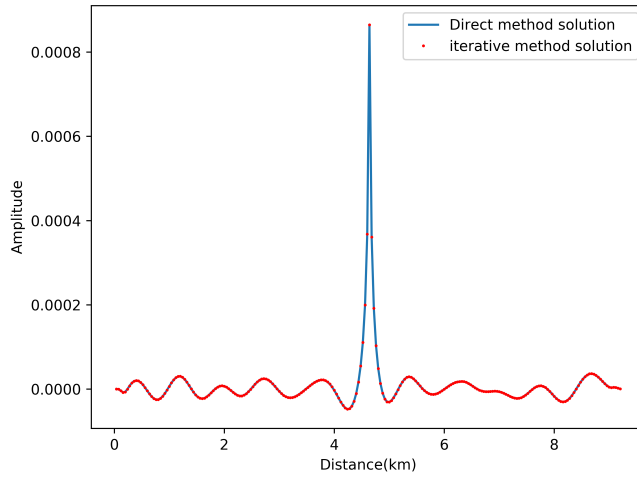


Figure 4.36: Marmousi 모델의  $kh=0.418879$ 일때 송신원 깊이에서의 직접법의 해와 반복법의 해 비교

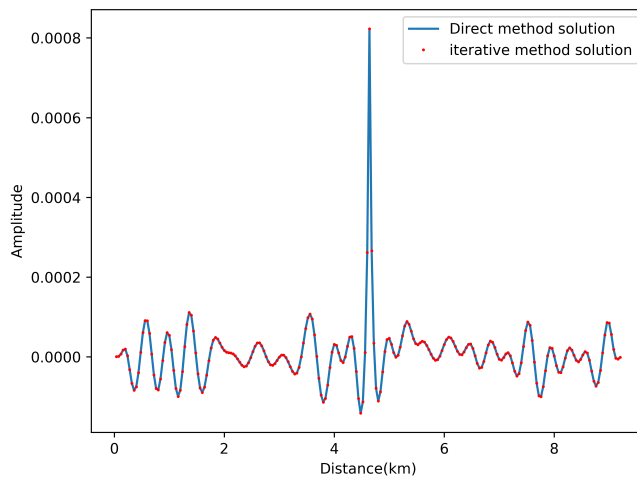


Figure 4.37: Marmousi 모델의  $kh=0.837758$ 일때 송신원 깊이에서의 직접법의 해와 반복법의 해 비교

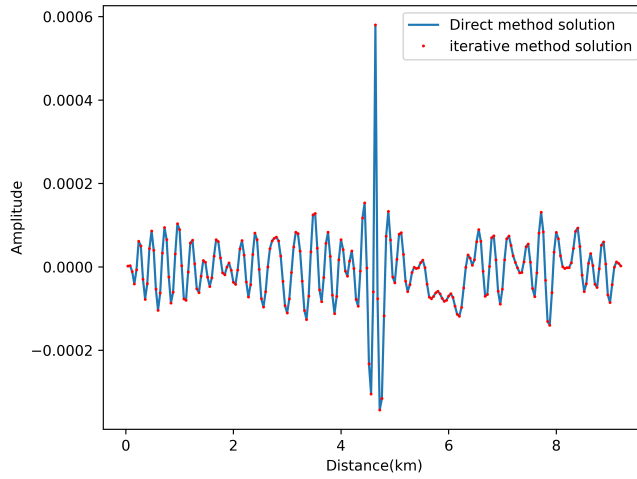


Figure 4.38: Marmousi 모델의  $kh=1.25664$ 일때 송신원 깊이에서의 직접법의 해와 반복법의 해 비교

## Seismogram result

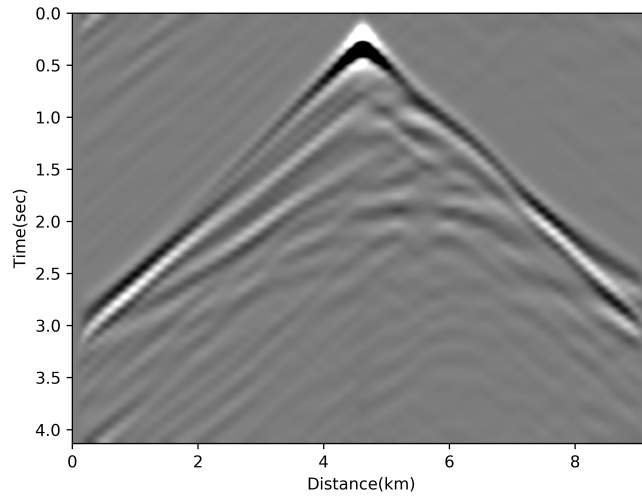


Figure 4.39: 직접법으로 구한 Marmousi 모델의 seismogram

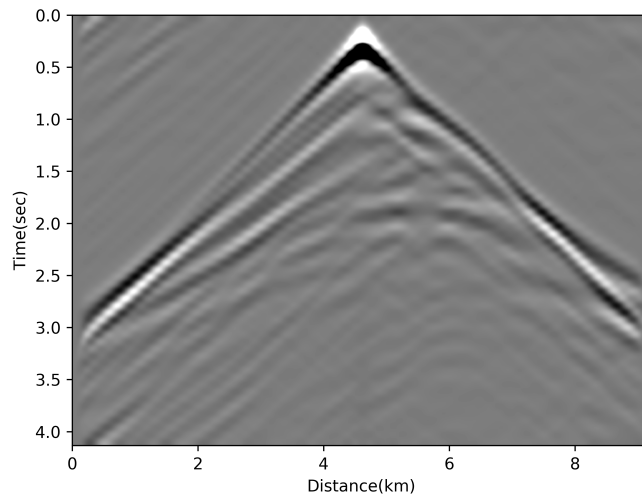


Figure 4.40: 반복법으로 구한 Marmousi 모델의 seismogram

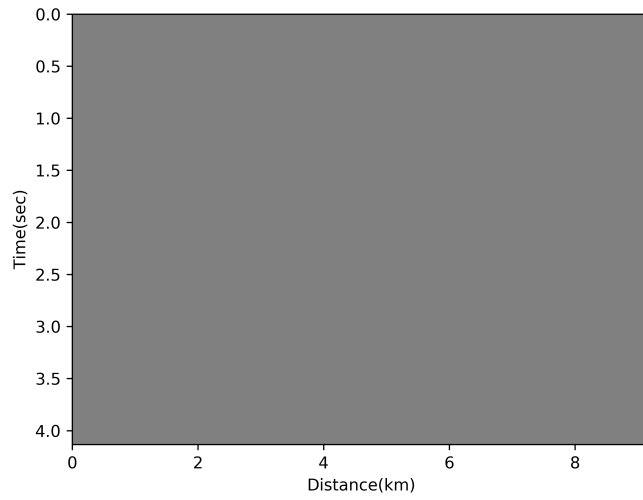


Figure 4.41: 직접법과 반복법으로 구한 Marmousi 모델의 seismogram의 차이



#### 4.2.2 2D : Pluto

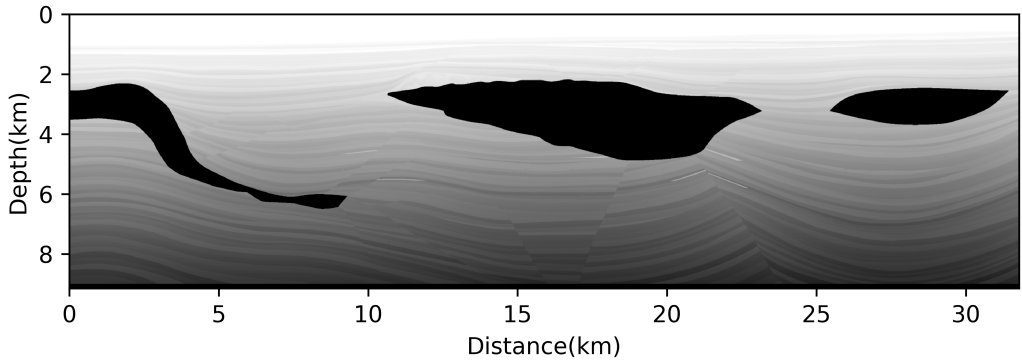


Figure 4.42: Pluto 속도 모델

사용한 Pluto 모델은 물리영역  $31.76\text{km} \times 9.16\text{km}$ 이고 이를 수치영역  $794 \times 229$  노드, 40m 격자로 설정하고, 주파수는 0 Hz 부터 7.5 Hz 까지( $kh = 0$  부터  $kh = 1.25664$ 에 해당) 0.1 Hz 간격으로 실험하였다.

해의 정확도를 검증하기 위해 직접법으로 구한 해와 반복법으로 구한 해의 비교를 진행하였다. 그림 4.43 부터 그림 4.45는 각  $kh$ 에 따라 송신원 깊이에서의 해를 비교한 것이다. 비교를 위해 실험을 한  $kh$ 중 3개를 작은 값에서 부터 큰 값의 결과를 샘플링 하였다. 결과 이미지에서 파란 실선이 직접법의 해이고 빨간 점이 반복법의 해이다. 각각의  $kh$ 에서 직접법의 해와 반복법의 해가 일치하는 것을 볼 수 있다. 이를 통해 반복법으로 구한 해가 직접법으로 구한 해만큼의 정확도를 가진다는 것을 확인 할 수 있다.

추가적으로  $kh$ 에 따른 결과를 종합하여 확인 할 수 있게 낮은  $kh$ 에서 부터 높은  $kh$  까지의 결과를 종합한 시간영역 seismogram을 구하였다. 우선 Pluto 모델의

경우 논문에서 사용한 다른 모델들에 비해 깊이가 깊어서 실험에서 사용한 주파수 간격에서는 wrapping around가 일어나 후기 신호가 초기 신호에 겹쳐서 나온 것을 확인할 수 있다. 이를 해결하기 위해서는 더 작은 간격으로 긴 시간의 시뮬레이션이 될 수 있게 실험 조건을 설정해야 한다. 그림 4.46는 직접법으로 주파수 영역 실험을 통해 구한 seismogram이고, 4.47는 반복법을 사용하여 주파수 영역 실험을 통해 구한 seismogram이다. 육안으로 둘의 차이를 구분하기 힘들기 때문에 4.48로 두 결과의 차이를 이미지로 만들었다. Pluto 모델의 경우 원 이미지들의 스케일로 볼 때 차이가 나지 않는 것을 확인할 수 있다. 차이를 수치적으로 보면 두 모델의 차이가 최소  $-3.0850e-07$ 에서 최대  $2.5425e-07$ 으로 반복법과 직접법의 계산값이 유의미한 차이가 없다는 것을 확인할 수 있다.

kh	$n_x \times n_z$	iteration	Time
0.418879	$794 \times 229$	21	1.320
0.837758	$794 \times 229$	19	1.203
1.24826	$794 \times 229$	22	1.366

Table 4.4: Pluto 모델 각 kh별 반복 횟수와 소요 시간

## Frequency domain result

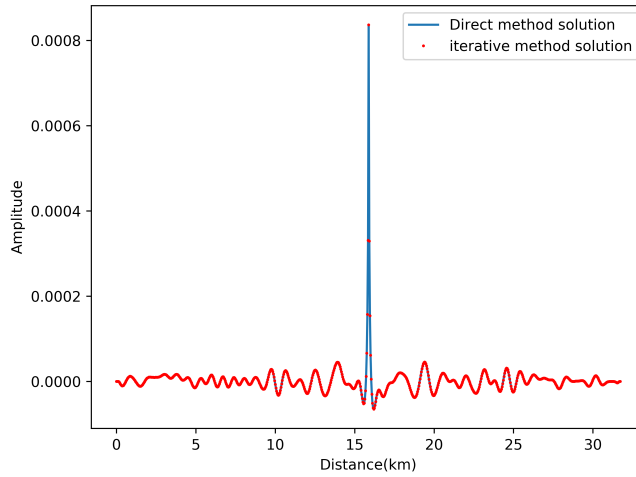


Figure 4.43: Pluto 모델의  $kh=0.418879$ 일때 송신원 깊이에서의 직접법의 해와 반복법의 해 비교

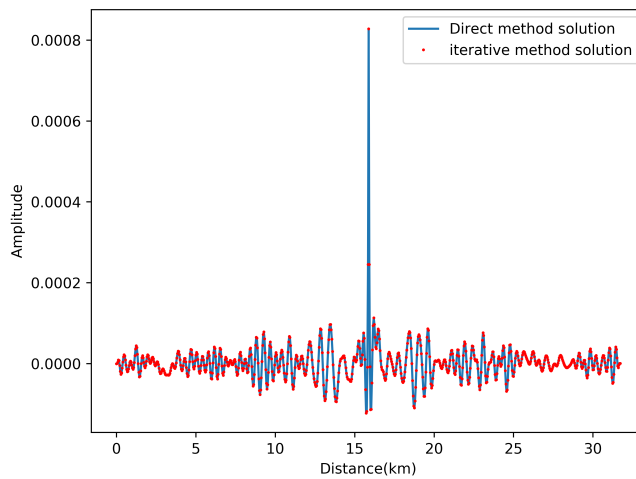


Figure 4.44: Pluto 모델의  $kh=0.837758$ 일때 송신원 깊이에서의 직접법의 해와 반복법의 해 비교

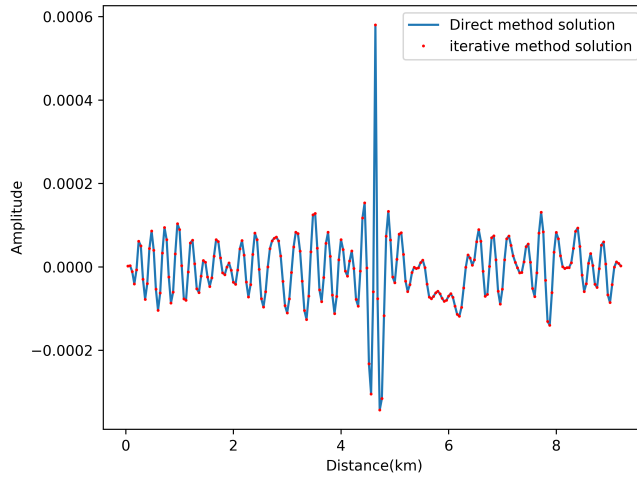


Figure 4.45: Pluto 모델의  $kh=1.25664$ 일때 송신원 깊이에서의 직접법의 해와 반복법의 해 비교

## Seismogram result

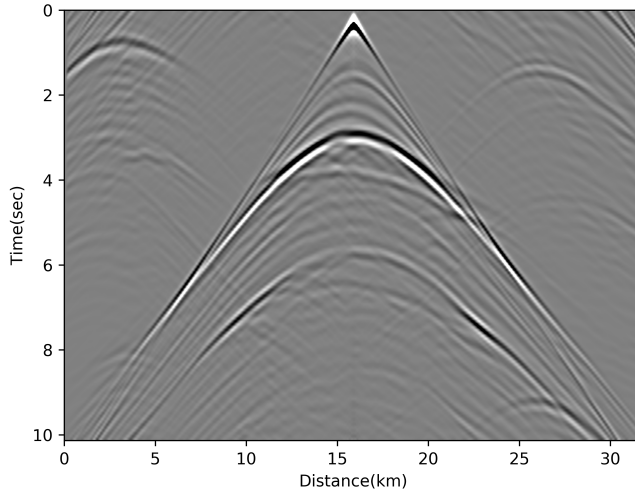


Figure 4.46: 직접법으로 구한 Pluto 모델의 seismogram

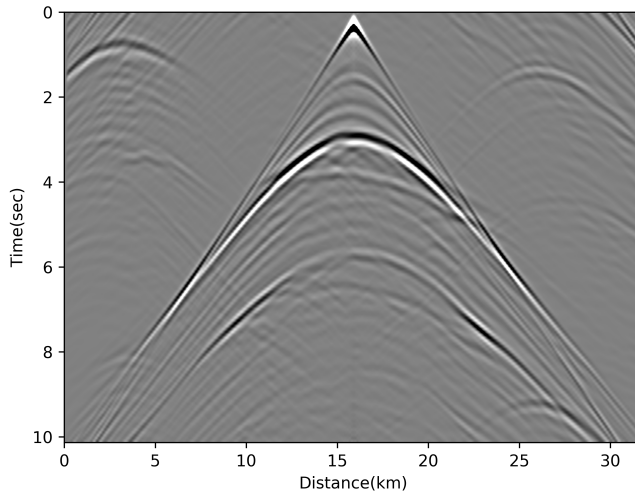


Figure 4.47: 반복법으로 구한 Pluto 모델의 seismogram

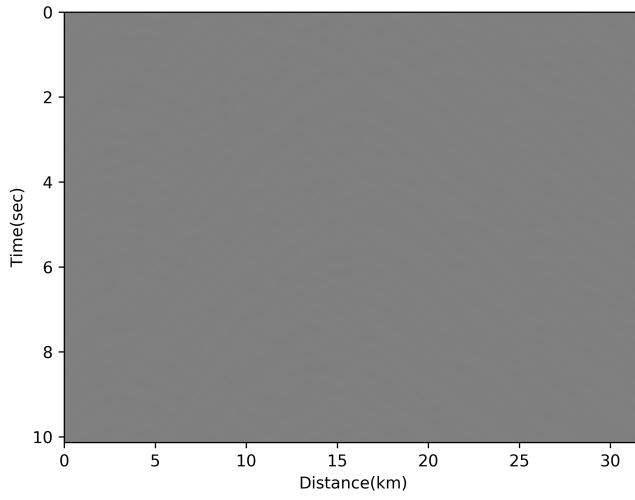


Figure 4.48: 직접법과 반복법으로 구한 Pluto 모델의 seismogram의 차이

### 4.2.3 3D : SEG/EAGE 3D

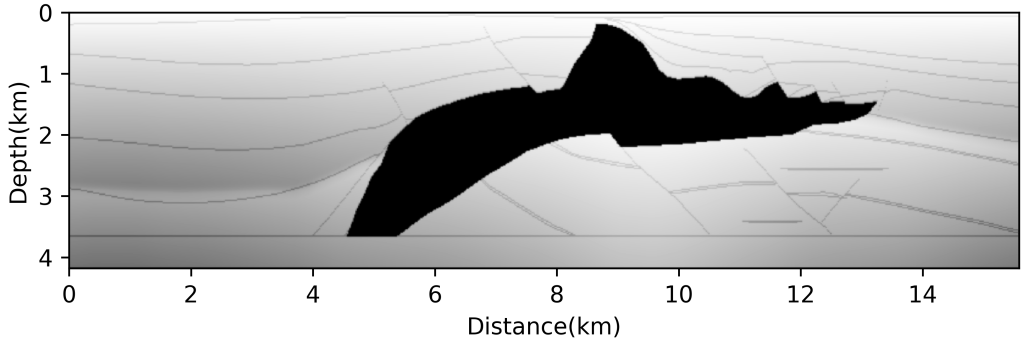


Figure 4.49: SEG/EAGE 3D 속도 모델의 단면

사용한 SEG/EAGE 3D 모델은 물리영역 13.52km x 13.52km x 4km이고 이를 수치영역 169 x 169 x 50 노드, 80m 격로 설정하고, 주파수는 0 Hz 부터 7.5 Hz 까지( $kh = 0$  부터  $kh = 1.25664$ 에 해당) 0.2 Hz 간격으로 실험하였다.

해의 정확도를 검증하기 위해 직접법으로 구한 해와 반복법으로 구한 해의 비교를 진행하였다. 그림 4.50 부터 그림 4.52는 각  $kh$ 에 따라 송신원 깊이에서의 해를 비교한 것이다. 비교를 위해 실험을 한  $kh$ 중 3개를 작은 값에서 부터 큰 값의 결과를 샘플링 하였다. 결과 이미지에서 파란 실선이 직접법의 해이고 빨간 점이 반복법의 해이다. 각각의  $kh$ 에서 직접법의 해와 반복법의 해가 일치하는 것을 볼 수 있다. 이를 통해 반복법으로 구한 해가 직접법으로 구한 해만큼의 정확도를 가진다는 것을 확인 할 수 있다.

추가적으로  $kh$ 에 따른 결과를 종합하여 확인 할 수 있게 낮은  $kh$ 에서 부터 높은  $kh$  까지의 결과를 종합한 시간영역 seismogram을 구하였다. 그림 4.53는 직접법

으로 주파수 영역 실험을 통해 구한 seismogram이고, 4.54는 반복법을 사용하여 주파수 영역 실험을 통해 구한 seismogram이다. 육안으로 둘의 차이를 구분하기 힘들기 때문에 4.55로 두 결과의 차이를 이미지로 만들었다. SEG/EAGE 3D 모델의 경우 원 이미지들의 스케일로 볼 때 차이가 나지 않는 것을 확인 할 수 있다. 차이를 수치적으로 보면 두 모델의 차이가 최소  $-7.0343e-08$ 에서 최대  $6.8857e-08$ 으로 반복법과 직접법의 계산값이 유의미한 차이가 없다는 것을 확인 할 수 있다.

kh	$n_x \times n_y \times n_z$	iteration	Time
0.837758	$169 \times 169 \times 50$	139	76.84
1.67552	$169 \times 169 \times 50$	177	93.39
2.51327	$169 \times 169 \times 50$	175	112.13

Table 4.5: SEG/EAGE 3D 모델 각 kh별 반복 횟수와 소요 시간



## Frequency domain result

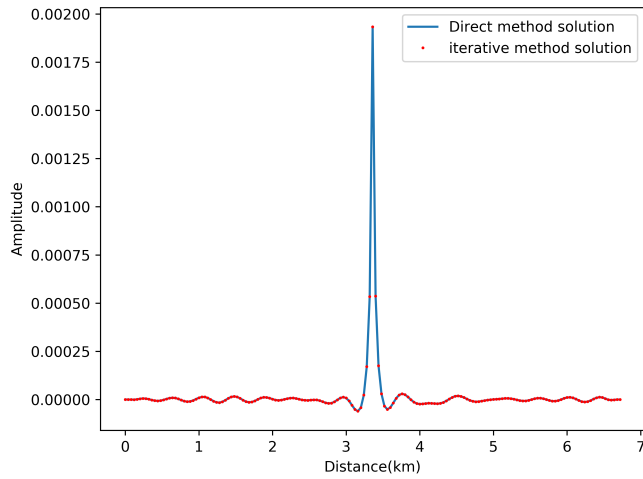


Figure 4.50: SEG/EAGE 3D 모델의  $kh=0.837758$ 일때 송신원 깊이에서의 직접법의 해와 반복법의 해 비교

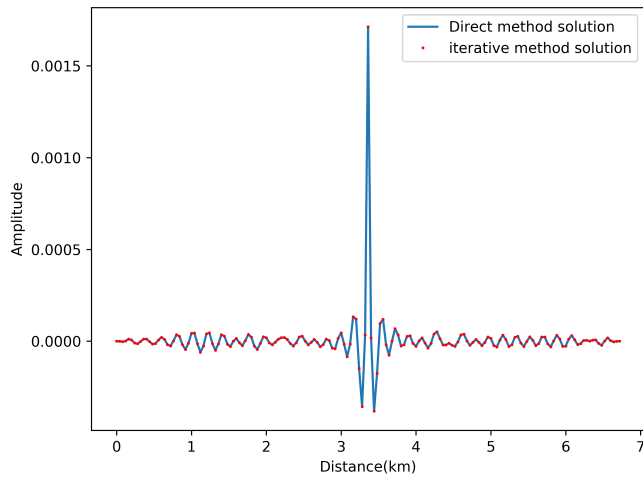


Figure 4.51: SEG/EAGE 3D 모델의  $kh=1.67552$ 일때 송신원 깊이에서의 직접법의 해와 반복법의 해 비교

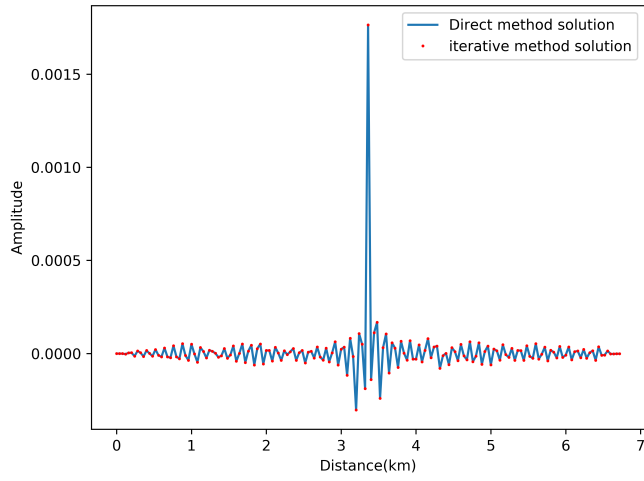


Figure 4.52: SEG/EAGE 3D 모델의  $kh=2.51327$ 일때 송신원 깊이에서의 직접법의 해와 반복법의 해 비교

## Seismogram result

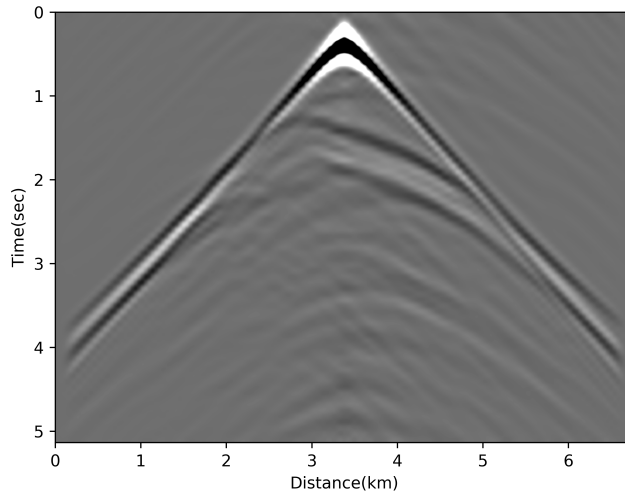


Figure 4.53: 직접법으로 구한 SEG/EAGE 3D 모델의 seismogram

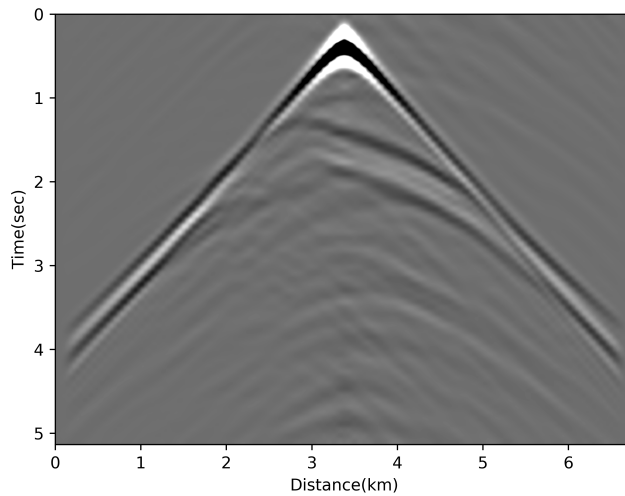


Figure 4.54: 반복법으로 구한 SEG/EAGE 3D 모델의 seismogram

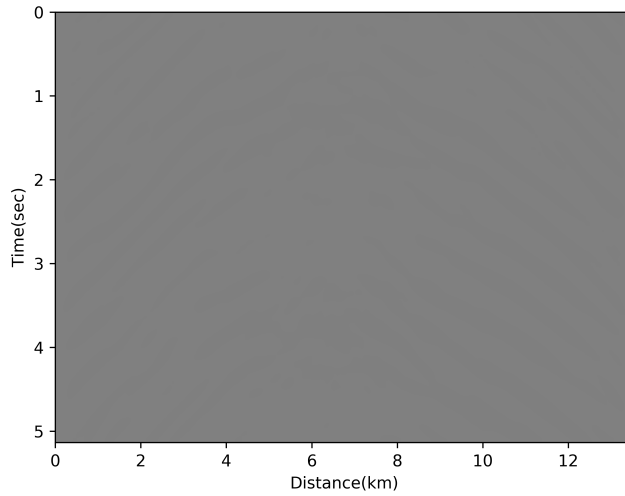


Figure 4.55: 직접법과 반복법으로 구한 SEG/EAGE 3D 모델의 seismogram의 차이

앞의 2개의 2차원 불균질 모델과 1개의 3차원 불균질 모델을 통해 반복법의 성능을 확인하였다. SL preconditioner가 적용된 크릴로브 부분공간법을 통해 직접법 정확도에 준하는 결과를 도출해 냈으므로 다양한 조건의 불균질 모델에서 성공적으로 적용 가능한 것을 보여주었다. 이를 통해 SL preconditioner를 사용한 크릴로브 부분공간법으로 직접법으로 해를 구하는데 어려움이있던 큰 규모의 문제에도 적용 가능할 것으로 생각된다.

# Chapter 5

## Conclusion

본 연구에서는, 헬름홀츠 방정식을 PML 경계조건을 포함한 복소 대칭 이산화하였으며, 반복법의 수렴속도를 개선시키기 위해 preconditioner 행렬의 복소 시프트를 조절하여 최적의 고유값 군집화를 설계할 수 있음을 보여주는 Shifted Laplace preconditioner가 사용되었다. preconditioner 또한 대칭 이산화를 적용하였고, 이를 통해 반복법에 preconditioner를 적용하기 위해 행렬을 불완전 분해시 대칭 행렬에 사용할 수 있는 불완전 Cholesky 분해(IC)를 사용하여 저장공간과 연산에서 기존 연구들에 비해 2배 가량의 효율을 얻을 수 있었다. 또한 불완전 분해시 탈락 기준에서 더 효율적인 ICT(p)방법을 도입하였다. 이러한 기술들의 조합을 통해 헬름홀츠 방정식을 해를 빠르고 정확하게 구할 수 있음을 확인하였다. 본 연구에서는 다음과 같은 실험들을 진행하고 그에 따른 결과를 얻었다.

1. 헬름홀츠 방정식을 풀때 효율적인 반복법을 찾기위해 크릴로브 부분공간법들의 연산에 필요한 저장 공간 및 반복에 따른 연산 횟수의 비교를 하였다. 여기서,

본 연구의 조건에서 CR이 반복법의 수렴때 까지 다른 반복법들에 비해 더 적은 연산 비용이 필요하여 효율적이라는 것을 확인하였다. 하지만 다른 반복법들도 큰 성능의 차이가 있진 않았고, 반복법의 성능은 preconditioner에 영향을 많이 받기 때문에 본 논문에서 얻은 결과를 활용한다면 다른 크릴로브 부분공간법들도 좋은 성능을 보여줄 것으로 사료된다.

2. SL preconditioner의 근사 역행렬을 구할 때 탈락 기준에 따른 ICT(p)와 IC(k)의 사용 결과 비교를 통해 IC(k)가 계산 시간과 저장공간에서 개선된 성능을 보여주었다. 같은 문제 조건에서 성능의 차이는 최대 계산 시간에서는 약 3.5배, 저장공간에서는 9.5배의 차이가 나는 것을 확인하였다. 이는 시스템의 크기가 커질수록 이에 비례하여 차이가 더 커질 수 있기 때문에 ICT(p)로 인해 확실한 효율의 증대를 기대할 수 있다.

3. SL preconditioner와 ICT(p)를 사용할 때 수치 영역에 따른 시프트 $\epsilon$ 과 탈락 기준  $p$ 의 경향성을 확인하는 실험을 하였다. 시프트 $\epsilon$ 의 경우 너무 작거나 큰 값을 사용하면 수렴하지 못하는 것을 확인하였다. 반면 최적의 시프트 $\epsilon$ 를 사용할 경우 수렴속도가 매우 향상된 것을 확인할 수 있었다. 또한 2차원일 경우와 3차원일 경우에는 최적의 시프트는 다른 값을 갖게 되지만 같은 차원일 경우 작은 범위 안의 값을 상수로 사용할 수 있는 것을 확인하였다. 하지만 수치영역의 크기가 커질수록 탈락 기준  $p$ 가 일정 이상 커져야 시프트 $\epsilon$ 의 경향이 보장되었다. 하지만 큰 탈락 기준  $p$ 를 사용할 경우 사용할 수 있는 시프트 $\epsilon$ 의 범위가 줄어드는 경우도 확인할 수 있었다. 또한 탈락 기준  $p$ 가 커짐에 따라 행렬의 분해시 소요되는 시간이 증가한다. 따라서, 안정적인 시프트 $\epsilon$  값의 사용과 컴퓨터 자원의 효율적인 활용을 위해서 탈락 기준  $p$  값의 유동적인 조절이 필요할 것으로 보인다.

4. SL preconditioner와 ICT(p)를 사용할 때  $kh$ 에 따른 시프트 $\epsilon$ 과 탈락 기준  $p$ 의 경향성을 확인하는 실험을 하였다. 여기서  $kh$ 에 따라 시프트 $\epsilon$ 과 탈락 기준  $p$ 가 크게 영향을 받지 않는 것을 확인하였다. 불균질 모델의 경우, 일정한  $kh$ 가 아니기 때문에  $kh$ 에 따른 영향이 크다면 일관적으로 사용하는데 어려움이 있을 수 있지만, 그 영향이 크지 않기 때문에 균질 모델에서 찾은 최적의 시프트 $\epsilon$ 과 탈락 기준  $p$ 를 제약없이 사용할 수 있다는 것을 알 수 있다.

5. 상기 결과들을 통해 얻은 최적의 시프트 $\epsilon$ 과 탈락 기준  $p$ 를 3가지 비균질 모델에 적용하였다. 비교적 단순한 모델에서 고속도층이 있는 복잡한 모델까지 2차원과 3차원 문제 모두에서 잘 작동하였다. 또한 주파수 영역 해와 여러 주파수들을 종합하여 구한 seismogram의 결과를 직접법의 결과와 비교시 허용범위 내의 오차 수준의 정확도를 보여주며 본 논문에서 조합한 기법들을 통해 빠르고 정확한 해를 구할 수 있음을 확인하였다.

본 연구에서 실험한 169X169X50 크기의 3차원 문제의 경우, 고도로 최적화된 프론탈 솔버인 UMFPACK의 연산 시간의 약 5%인 것으로 나타났다. 이 실험은 동일한 성능의 컴퓨터에서 30 코어를 사용한 최적화된 직접법 패키지와 아직 최적화가 진행되지 않은 본 연구에서 개발한 방법을 1 코어를 사용했을 때의 성능 차이이기 때문에 보다 불리한 방법에서도 SL preconditioner를 사용한 반복법이 월등한 성능을 보여주는 것을 확인할 수 있는 결과이다. 본 연구에서 활용한 SL preconditioner는 deflation 기법을 통해 고유값 스펙트럼의 개선하여 추가적인 수렴 속도 개선이 가능하며, preconditioner를 근사하는 ICT(p) 과정중 행렬의 스케일링, 치환(permutaion) 등을 통해 보다 수치적인 성질을 개선 시켜 계산 효율을

증대시킬 수 있다. 또한 행렬의 분해 과정에서 Openmp를 사용한 병렬화, 행렬의 계산 중 행렬-벡터 연산을 GPU kernel - cusparse level2 등을 활용하여 가속시킬 수 있는 여지가 있다. 이를 통해, SL preconditioner와 ICT(p)를 활용한다면 직접법 또는 반복법으로 적용 실효성이 낮았던 응용 분야(완전파형역산, 역시간구조보정 등)의 3차원 문제들을 효과적으로 해결할 수 있을 것으로 사료된다.



# Bibliography

- [1] W. E. Arnoldi. The principle of minimized iterations in the solution of the matrix eigenvalue problem. *Quarterly of Applied Mathematics*, 9(1):17–29, 1951.
- [2] I. M. Babuska and S. A. Sauter. Is the pollution effect of the FEM avoidable for the Helmholtz equation considering high wave numbers? *SIAM Journal on Numerical Analysis*, 34(6):2392–2423, 1997.
- [3] A. Bamberger, P. Joly, and J. E. Roberts. Second-order absorbing boundary conditions for the wave equation: a solution for the corner problem. *SIAM Journal on Numerical Analysis*, 27(2):323–352, 1990.
- [4] A. Bamberger, R. Glowinski, and Q. H. 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.
- [5] A. Bayliss, M. Gunzburger, and E. Turkel. Boundary conditions for the numerical solution of elliptic equations in exterior regions. *SIAM Journal on Applied Mathematics*, 42(2):430–451, 1982.

- [6] A. Bayliss, C. I. Goldstein, and E. Turkel. An iterative method for the Helmholtz equation. *Journal of Computational Physics*, 49(3):443–457, 1983.
- [7] A. Bayliss, C. I. Goldstein, and E. Turkel. On accuracy conditions for the numerical computation of waves. *Journal of Computational Physics*, 59(3):396–404, 1985.
- [8] J.-P. Bérenger. Perfectly matched layer (PML) for computational electromagnetics. *Synthesis Lectures on Computational Electromagnetics*, 2(1):1–117, 2007.
- [9] J.-P. Bérenger et al. A perfectly matched layer for the absorption of electromagnetic waves. *Journal of Computational Physics*, 114(2):185–200, 1994.
- [10] A. Brandt. Multi-level adaptive solutions to boundary-value problems. *Mathematics of Computation*, 31(138):333–390, 1977.
- [11] A. Brandt and I. Livshits. Wave-ray multigrid method for standing wave equations. *Electron. Trans. Numer. Anal*, 6(162-181):91, 1997.
- [12] A. Brandt and S. Ta’asan. Multigrid method for nearly singular and slightly indefinite problems. In *Multigrid Methods II*, pages 99–121. Springer, 1986.
- [13] W. L. Briggs, V. E. Henson, and S. F. McCormick. *A multigrid tutorial*. SIAM, 2000.

- [14] 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. *Numerical Linear Algebra with Applications*, 20(4):663–688, 2013.
- [15] C. Cerjan, D. Kosloff, R. Kosloff, and M. Reshef. A nonreflecting boundary condition for discrete acoustic and elastic wave equations. *Geophysics*, 50(4):705–708, 1985.
- [16] Z. Chen, D. Cheng, W. Feng, T. Wu, and H. Yang. A multigrid-based preconditioned Krylov subspace method for the Helmholtz equation with PML. *Journal of Mathematical Analysis and Applications*, 383(2):522–540, 2011.
- [17] D. Cheng, X. Tan, and T. Zeng. A dispersion minimizing finite difference scheme for the Helmholtz equation based on point-weighting. *Computers & Mathematics with Applications*, 73(11):2345–2359, 2017.
- [18] W. C. Chew and W. H. Weedon. A 3D perfectly matched medium from modified Maxwell’s equations with stretched coordinates. *Microwave and Optical Technology Letters*, 7(13):599–604, 1994.
- [19] S.-C. T. Choi. Minimal residual methods for complex symmetric, skew symmetric, and skew hermitian systems. *arXiv preprint arXiv:1304.6782*, 2013.
- [20] E. Chow and Y. Saad. ILUS: an incomplete LU preconditioner in sparse

- skyline format. *International Journal for Numerical Methods in Fluids*, 25(7):739–748, 1997.
- [21] R. Clayton and B. Engquist. Absorbing boundary conditions for acoustic and elastic wave equations. *Bulletin of the Seismological Society of America*, 67(6):1529–1540, 1977.
- [22] P.-H. Cocquet and M. J. Gander. On the minimal shift in the shifted Laplacian preconditioner for multigrid to work. In *Domain Decomposition Methods in Science and Engineering XXII*, pages 137–145. Springer, 2016.
- [23] 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.
- [24] E. d’Azevedo, P. Forsyth, and W.-P. Tang. Towards a cost-effective ILU preconditioner with high level fill. *BIT Numerical Mathematics*, 32(3):442–463, 1992.
- [25] B. Després. Domain decomposition method and the Helmholtz problem. *Mathematical and Numerical Aspects of Wave Propagation Phenomena*, pages 44–52, 1991.
- [26] A. P. Dowling and J. E. Ffowcs Williams. *Sound and sources of sound*. Horwood, 1983.
- [27] H. C. Elman. A stability analysis of incomplete LU factorizations. *Mathematics of Computation*, pages 191–217, 1986.

- [28] H. C. Elman, O. G. Ernst, and D. P. O’leary. A multigrid method enhanced by Krylov subspace iteration for discrete Helmholtz equations. *SIAM Journal on Scientific Computing*, 23(4):1291–1315, 2001.
- [29] B. Engquist and A. Majda. Absorbing boundary conditions for numerical simulation of waves. *Proceedings of the National Academy of Sciences*, 74(5):1765–1766, 1977.
- [30] 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.
- [31] Y. A. Erlangga and R. Nabben. On a multilevel Krylov method for the Helmholtz equation preconditioned by shifted Laplacian. *Electronic Transactions on Numerical Analysis*, 31(403-424):3, 2008.
- [32] Y. A. Erlangga, C. Vuik, and C. W. Oosterlee. On a class of preconditioners for solving the Helmholtz equation. *Applied Numerical Mathematics*, 50(3-4):409–425, 2004.
- [33] Y. A. Erlangga, C. W. Oosterlee, and C. Vuik. A novel multigrid based preconditioner for heterogeneous Helmholtz problems. *SIAM Journal on Scientific Computing*, 27(4):1471–1492, 2006.
- [34] Y. A. Erlangga, C. Vuik, and C. W. Oosterlee. Comparison of multigrid and incomplete LU shifted-Laplace preconditioners for the inhomogeneous Helmholtz equation. *Applied Numerical mathematics*, 56(5):648–666, 2006.

- [35] O. G. Ernst and M. J. Gander. Why it is difficult to solve Helmholtz problems with classical iterative methods. In *Numerical Analysis of Multiscale Problems*, pages 325–363. Springer, 2012.
- [36] 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.
- [37] R. W. Freund. Conjugate gradient-type methods for linear systems with complex symmetric coefficient matrices. *SIAM Journal on Scientific and Statistical Computing*, 13(1):425–448, 1992.
- [38] 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. *Numerische Mathematik*, 2014.
- [39] M. J. Gander and F. Nataf. AILU: a preconditioner based on the analytic factorization of the elliptic operator. *Numerical Linear Algebra with Applications*, 7(7-8):505–526, 2000.
- [40] M. J. Gander and F. Nataf. AILU for Helmholtz problems: a new preconditioner based on the analytic parabolic factorization. *Journal of Computational Acoustics*, 9(04):1499–1506, 2001.
- [41] M. J. Gander and F. Nataf. An incomplete LU preconditioner for problems in acoustics. *Journal of Computational Acoustics*, 13(03):455–476, 2005.
- [42] M. J. Gander, I. G. Graham, and E. A. Spence. Applying GMRES to the Helmholtz equation with shifted Laplacian preconditioning: what is

- the largest shift for which wavenumber-independent convergence is guaranteed? *Numerische Mathematik*, 131(3):567–614, 2015.
- [43] A. Greenbaum, V. Pták, and Z. e. k. Strakoš. Any nonincreasing convergence curve is possible for GMRES. *SIAM Journal on Matrix Analysis and Applications*, 17(3):465–469, 1996.
- [44] I. Gustafsson. A class of first order factorization methods. *BIT Numerical Mathematics*, 18(2):142–156, 1978.
- [45] W. Hackbusch. *Multi-grid methods and applications (2nd Printing)*. Berlin: Springer, 2003.
- [46] G. R. Hadley. A complex Jacobi iterative method for the indefinite Helmholtz equation. *Journal of Computational Physics*, 203(1):358–370, 2005.
- [47] C.-H. Jo, C. Shin, and J. H. Suh. An optimal 9-point, finite-difference, frequency-space, 2-D scalar wave extrapolator. *Geophysics*, 61(2):529–537, 1996.
- [48] M. T. Jones and P. E. Plassmann. An improved incomplete Cholesky factorization. *ACM Transactions on Mathematical Software (TOMS)*, 21(1):5–17, 1995.
- [49] J.-H. Kimn and M. Sarkis. Shifted Laplacian RAS solvers for the Helmholtz equation. In *Domain Decomposition Methods in Science and Engineering XX*, pages 151–158. Springer, 2013.

- [50] A. L. Laird and M. Giles. Preconditioned iterative solution of the 2D Helmholtz equation. 2002.
- [51] C. Lanczos. *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*. United States Governm. Press Office Los Angeles, CA, 1950.
- [52] B. Lee, T. A. Manteuffel, S. F. McCormick, and J. Ruge. First-order system least-squares for the Helmholtz equation. *SIAM Journal on Scientific Computing*, 21(5):1927–1949, 2000.
- [53] C.-J. Lin and J. J. Moré. Incomplete Cholesky factorizations with limited memory. *SIAM Journal on Scientific Computing*, 21(1):24–45, 1999.
- [54] J. Liu and A. George. Computer solution of large sparse positive definite systems. *Prentice-Hall, Inc. Englewood Cliffs. NJ*, 7632:1981, 1981.
- [55] M. Magolu Monga Made. Incomplete factorization-based preconditionings for solving the Helmholtz equation. *International Journal for Numerical Methods in Engineering*, 50(5):1077–1101, 2001.
- [56] K. J. Marfurt. Accuracy of finite-difference and finite-element modeling of the scalar and elastic wave equations. *Geophysics*, 49(5):533–549, 1984.
- [57] J. A. Meijerink and H. A. Van Der Vorst. An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix. *Mathematics of Computation*, 31(137):148–162, 1977.



- [58] C. Oosterlee, C. Vuik, W. Mulder, and R.-E. Plessix. Shifted-Laplacian preconditioners for heterogeneous Helmholtz problems. In *Advanced Computational Methods in Science and Engineering*, pages 21–46. Springer, 2009.
- [59] H. Park, H. Song, Y. Park, and C. Shin. A symmetric discretization of the perfectly matched layer for the 2D Helmholtz equation. *Journal of Seismic Exploration*, 26:541–560, 12 2017.
- [60] R. Plessix and W. Mulder. Separation-of-variables as a preconditioner for an iterative Helmholtz solver. *Applied Numerical Mathematics*, 44(3):385–400, 2003.
- [61] A. C. Reynolds. Boundary conditions for the numerical solution of wave propagation problems. *Geophysics*, 43(6):1099–1110, 1978.
- [62] Y. Saad. ILUT: A dual threshold incomplete LU factorization. *Numerical Linear Algebra with Applications*, 1(4):387–402, 1994.
- [63] Y. Saad and M. H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 7(3):856–869, 1986.
- [64] A. Sheikh, C. Vuik, and D. Lahaye. Fast iterative solution methods for the Helmholtz equation. *Reports of the Department of Applied Mathematical Analysis*, 09-11, 2009.
- [65] T. Sogabe and S.-L. Zhang. A COCR method for solving complex sym-

- metric linear systems. *Journal of Computational and Applied Mathematics*, 199(2):297–303, 2007.
- [66] C. C. Stolk. A dispersion minimizing scheme for the 3-D Helmholtz equation based on ray theory. *Journal of Computational Physics*, 314:618–646, 2016.
- [67] R. Tezaur, A. Macedo, and C. Farhat. Iterative solution of large-scale acoustic scattering problems with multiple right hand-sides by a domain decomposition method with Lagrange multipliers. *International Journal for Numerical Methods in Engineering*, 51(10):1175–1193, 2001.
- [68] E. Turkel, A. Yefet, et al. Absorbing PML boundary layers for wave-like equations. *Applied Numerical Mathematics*, 27(4):533, 1998.
- [69] H. A. Van der Vorst. Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 13(2):631–644, 1992.
- [70] M. B. van Gijzen, Y. A. Erlangga, and C. Vuik. Spectral analysis of the discrete Helmholtz operator preconditioned with a shifted Laplacian. *SIAM Journal on Scientific Computing*, 29(5):1942–1958, 2007.
- [71] P. Vanek, J. Mandel, and M. Brezina. Two-level algebraic multigrid for the Helmholtz problem. *Contemporary Mathematics*, 218(349-356):187, 1998.
- [72] X. I. Yang and R. Mittal. Acceleration of the Jacobi iterative method by factors exceeding 100 using scheduled relaxation. *Journal of Computational Physics*, 274:695–708, 2014.

## 초록

The Helmholtz equation is a partial differential equation (PDE) that governs the wavefield of a frequency component. Since it is an elliptic type PDE, it is expressed in the form that the solution can be obtained by solving a linear system, which is composed of the impedance matrix and a given right hand side including source information. Due to the reciprocity property of the Helmholtz equation, the discretized impedance matrix can be naturally expressed as a complex symmetric matrix that can reduce a memory requirement and the number of arithmetic operations by half by applying the Cholesky factorization. However, it is not widely disseminated how to reserve such symmetric property of the matrix using the PML boundary condition. Therefore, firstly, we propose a method that discretizes the Helmholtz equation as a symmetric matrix for the entire spatial nodes including the PML layer. We also suggest the optimized discrete coefficient to generate impedance matrix which satisfies the dispersion error of less than 0.5% within the range that the number of grids per wavelength is greater than 4. The system can be solved by using the direct matrix solver for 2D or small 3D problems, however, time taken and memory requirement to factorize and solve the impedance matrix is extremely high to solve general 3D problems considering the complexities. In this case, iterative methods, in particular, the Krylov subspace methods, such as conjugate gradient method, general minimal residual method, etc., can be effectively applied. In this study,

many Krylov iterative solvers that can be applied to the symmetric matrix are implemented to experimentally investigate which method is advantageous for solving the impedance matrix. As a result, it was observed that for various kh conditions, the conjugate residual method solves the matrix in the fastest time consistently. To shorten the convergence speed, we applied the shifted Laplacian preconditioner that can optimally modify the eigenvalue clustering by adjusting the complex shift( $\epsilon$ ) of the preconditioner matrix. To apply this preconditioner, incomplete factorization method is generally used. In this study, we suggest incomplete Cholesky decomposition, especially ICT(p) method, that leaves only p largest values of the factorized matrix. Optimal pairs of  $\epsilon$  and p for given kh condition are also suggested, which is validated by numerical examples using the synthetic velocity models. It is shown that only few minutes are taken to solve the problem to yield the wavefield that has negligibly small error compared to that obtained by the direct method, the error is negligibly small. Especially, in the case of a 3D problem with a size of 169X169X50, it is shown that it is about 5% of the computation time of UMFPACK, a highly optimized directive multifrontal solver.

**주요어:** 헬름홀츠 방정식, 크릴로브 부분공간법, preconditioner, shifted Laplace  
**학번:** 2014-21329