Important Notice

UNIVERSITY OF CALGARY

Seismic depth migration using the Gabor imaging theories

by

Yongwang Ma

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE

DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF GEOSCIENCE

CALGARY, ALBERTA

December, 2008

THE UNIVERSITY OF CALGARY

FACULTY OF GRADUATE STUDIES

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies for acceptance, a thesis entitled "Seismic depth migration using the Gabor imaging theories" submitted by Yongwang Ma in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Supervisor, Dr. Gary F. Margrave, Department of Geoscience

Dr. Edward S. Krebes, Department of Geoscience

Dr. John C. Bancroft, Department of Geoscience

Dr. Elise Fear, Department of Electrical and Computer Engineering

---

Dr. Michael A. Slawinski, External examiner, Department of Earth Sciences
Memorial University

Date

Abstract

Seismic depth migrations are used to image the earth with more plausible details; they become important tools to obtain 'true' pictures of the subsurface and attract more and more attention. This thesis is dedicated to developing a wave-equation depth migration method, called Gabor depth migration, where the Gabor transform is employed.

The Gabor depth migration method is developed to image structures with strong lateral velocity variations. The wavefield is localized by narrow and uniform partitions such that the most rapidly varying velocity can be accommodated to get accurate images. However, such narrow and uniform partitions introduce inefficient depth migrations, which is not always necessary. Instead, wide and optimized partitions should be utilized according to lateral velocity variations given accuracy criteria.

Adaptive partitioning algorithms are developed to consolidate narrow, small partitions where lateral variations are considered as minor given accuracy criteria. Such a consolidation process usually results in fewer and wider partitions, leading to fewer Fourier transforms in the wavefield extrapolation. Therefore, computational cost is minimized under given accuracy thresholds. Three adaptive partitioning algorithms described in this thesis help to achieve efficient Gabor depth migrations at the cost of losing some imaging accuracy. The Gabor depth migration method can achieve accurate images by increasing accuracy thresholds. An optimized imaging process can be obtained by using trade-off between efficiency and accuracy in the migration method.

The Gabor depth migration method is made more efficient using the spatial resampling without losing imaging accuracy. The depth migration method reduces computational cost by down-sampling wavefields at low frequencies, giving a faster imaging process.

Depth images from the Marmousi data verify the improvement of migration speed without obvious loss of imaging quality.

In addition, the Gabor depth migration method is modified to work with topographic seismic data. Tests on a synthetic 2D topographic seismic data set show accurate images.

The Gabor depth migration method can be extended straightforward from 2D to 3D. The adaptive partitioning algorithm using lateral position errors is easy to extend to higher dimensions, which simplifies the implementation of the adaptive Gabor depth migration in 3D.

Acknowledgements

During my four-year research and study in the Consortium of Research in Elastic Wave Exploration Seismology (CREWES) project in the Department of Geoscience, I have met many talented and kind people. Their help and support has contributed to my successes in the CREWES project as a graduate student. It has been stimulating and rewarding to be here and to travel around with the CREWES faculty members, staff and students to various conferences around the world. I would like to thank the following persons.

The most important thing my thesis supervisor, Dr. Gary F. Margrave, has passed on to me (and I am sure, other students) is the true scientific ethic and spirit, where honesty and integrity are a must. I am very grateful for Gary who has encouraged me to explore the unknown in my areas of study. Many discussions with him have cracked puzzles in my mind, which have been very helpful and have led to my successes during the proceeding four fruitful years.

I thank Dr. E. D. Krebes and Dr. John Bancroft for kindly being my supervisory committee members. I thank Dr. Don Lawton for allowing me to use the real 2D topographic seismic data from the Canadian Foothills in this thesis. I thank Dr. Larry Lines, Dr. Don Lawton and Dr. Michael P. Lamoureux (Department of Mathematics and Statistics) for being in my comprehensive examination committee. I thank Cathy Hubbell (graduate program administrator in this department) for her kind help and assistance during my study in the Department of Geoscience. I thank all other faculty, staff and students in this Department who have helped me in the past four years.

I thank all the CREWES staff members for their support. Kevin Hall (systems manager) and Dr. Rolf Maier are so kind that I was saved by them many times from computer

problems. I thank Louise Forgues (administrative manager), for her kind help and assistance to make life easier for us students. I sincerely thank Han-xing Lu (applications geoscientist), who has helped me many times to prepare seismic data for testing my migration codes. I thank Dr. Peter Manning and Kayla Bonham (HPC programmer), for their great effort to make this thesis read more smoothly and easy to understand. I thank David Henley (research geophysicist) for reviewing my papers devoted to the sponsor meeting every year, which has been a critical training process honing my writing skills in English.

I have had great time to work and enjoy myself in the CREWES project and the University of Calgary with all my fellow students. Especially, I would like to thank Jianlin (John) Zhang, Xiang Du, Zaiming Jiang, Chad Hogan, Carlos Montana, Jason McCrank, Kelley Classen, Joanna Cooper, Yong Hua, Fuju Chen, Zhihong Cao, Chunyan Xiao, Ben Wards, Zimin Zhang, Linping Dong, Hong Feng, Peng Cheng, and all other students.

I thank Shell Canada Limited for offering me a summer internship in 2006, which has been a valuable experience in my personal career development. I also thank Chevron Energy Company (Houston), who offered me an opportunity in 2007 to learn about different working environments and explore their technology of velocity model building for depth imaging, which has broadened my views outside of the seismic imaging.

To my family, I am grateful for the support and understanding from my wife during last four years. I thank my parents, my sisters and brothers for their faith in me all these years.

Thanks go to the following organizations for their generous financial supports throughout my thesis research. They are the sponsors of the CREWES project, the sponsors of the Pseudodifferential Operator Theory in Seismic Imaging (POTSI), the Natural Sciences and Engineering Research Council (NSERC) of Canada, the Mathematics of Information

Technology and Complex Systems (MITACS) of Canada, the Pacific Institute of Mathematical Sciences (PIMS), the Canadian Society of Exploration Geophysicists (CSEG) and the European Association of Geoscientists & Engineers (EAGE).

Dedication

To my family.

Table of Contents

Approval Page ii
Abstract iii
Acknowledgements v

Dedication viii

Table of Contents ix

List of Tables xii

List of Figures xiii
Chapter 1 Introduction 1
1.1 Necessity of seismic migrations 1
1.2 Evolution of seismic migration algorithms 2
1.3 Review on seismic migration algorithms 4
1.3.1 Kirchhoff migrations 4
1.3.2 Finite difference migrations 5
1.3.3 Frequency-wavenumber-domain migrations 6
1.3.4 Recent developments and trends of seismic migrations . . . 7
1.4 Motivations of the research in this thesis 8
1.5 Objective of the thesis 9
1.6 Claim of originality 10
1.7 Summary of the thesis 11
1.8 Summary of chapters 12
Chapter 2 Explicit Wavefield Extrapolation and the Gabor Transform 15
2.1 Explicit wavefield extrapolation 15
2.2 Explicit wavefield extrapolation with phase shift 17
2.2.1 Review of phase-shift methods 17
2.2.2 Fourier transforms 18
2.2.3 Wavefield extrapolation with the phase-shift method 20
2.3 Gabor transforms (windowed Fourier transforms) 26
2.3.1 Continuous Gabor transforms 26
2.3.2 Discrete Gabor transforms 28
2.4 Explicit wavefield extrapolation with the Gabor transform 29
2.4.1 Gabor extrapolation schemes simulating the PSPI and the NSPS 35
2.4.2 Gabor extrapolation schemes using intermediate p values . . 38
2.5 Implementation of the Gabor wavefield extrapolation 39
2.5.1 Gabor wavefield extrapolation 39

2.5.2 Impulse responses with the Gabor wavefield extrapolator . . 40
2.6 Chapter summary . . . 47
Chapter 3 Fast Gabor Depth Migration . . . 48
3.1 Computational cost of the Gabor depth migration method . . . 48
3.2 Adaptive Gabor wavefield extrapolation . . . 50
3.3 Adaptive partitioning algorithms . . . 51
3.3.1 Adaptive partitioning with velocity gradients . . . 52
3.3.2 Adaptive partitioning with phase errors . . . 55
3.3.3 Adaptive partitioning with lateral position errors . . . 61
3.4 Spatial Resampling . . . 70
3.4.1 Spatial resampling theory . . . 70
3.4.2 Spatial resampling examples . . . 71
3.5 Chapter summary . . . 73
Chapter 4 Depth Migration with 2D Synthetic Data . . . 75
4.1 Background of the Marmousi model . . . 75
4.2 Marmousi data set . . . 77
4.3 Prestack Gabor depth migration with the Marmousi data . . . 77
4.4 Efficient Gabor depth migrations . . . 79
4.4.1 Fast Gabor depth migration using the adaptive partitioning . 79
4.4.2 Fast Gabor depth migration using the spatial resampling . . 84
4.5 Trade-off between imaging accuracy and imaging efficiency . . . 87
4.6 Gabor depth migrations with various p values . . . 93
4.7 Chapter summary . . . 95
Chapter 5 Depth Migration with Topography . . . 97
5.1 Topographic seismic migrations using regular imaging methods . 98
5.2 Topographic seismic migrations using the Gabor method . . . 99
5.3 Depth migration with 2D Synthetic Data from Topography . . . 102
5.3.1 Velocity Model . . . 102
5.3.2 Model 94 synthetic data . . . 105
5.3.3 Initial imaging results and analysis . . . 107
5.3.4 Problems in depth migrations and the solutions to them . . . 110
5.3.5 Final imaging results for the synthetic data set . . . 116
5.4 Imaging 2D Real Data from Topography . . . 118
5.4.1 Acquisition parameters . . . 118
5.4.2 Processing before migration . . . 119
5.4.3 Tests with the chosen migration velocities . . . 119
5.4.4 Imaging results from the real data set . . . 120
5.5 Chapter summary . . . 121
Chapter 6 3D Gabor Depth Imaging Theory and Initial Imaging Tests 125
6.1 Wavefield extrapolation with the phase shift in 3D . . . 126

6.2 3D Gabor extrapolation theory 129
6.3 2D adaptive partitioning algorithm for 3D Gabor extrapolation . . 131
6.4 2D adaptive partitioning examples 137
6.5 3D impulse response 144
6.6 Chapter summary 156
Chapter 7 Conclusions and Future Directions 158
7.1 Conclusions 158
7.1.1 Gabor depth imaging theories 158
7.1.2 Accuracy and efficiency of the Gabor depth migration . . . 159
7.1.3 2D Gabor depth migration results 160
7.1.4 3D Gabor depth migration tests 160
7.2 Future directions 161
7.2.1 Adaptive partitioning algorithms 162
7.2.2 3D Gabor depth migration 162
7.3 Final remarks 162

Bibliography 164

List of Tables

3.1 Statistics of spatial samples in frequency bands. 72

4.1 Acquisition parameters of the Marmousi data set. 77

5.1 Acquisition parameters of the Grizzly seismic line. 118

List of Figures

2.1 Explicit wavefield extrapolation using recursive depth marching scheme. Δz often denotes a small step (e.g., 10 m). 16

2.2 A Fourier transform of a seismic trace. (a) Input signals. (b) Amplitude spectrum. 19

2.3 Numerical tests with Gabor transforms. (a) Input signal (the same one as shown in Figure 2.2). (b) Gabor spectrum of the input signal. (c) Recovered signal from the Gabor spectrum shown in (b) using the inverse Gabor transform. (d) Difference between the original (a) and the recovered (c) signals. 30

2.4 A flowchart showing the Gabor wavefield extrapolation (simulating the PSPI) on a shot record. 41

2.5 A flowchart showing the Gabor wavefield extrapolation (simulating the NSPS) on a shot record. 42

2.6 A flowchart showing the Gabor wavefield extrapolation (using the intermediate p) on a shot record. 43

2.7 Impulse responses in a homogeneous half space with a velocity of 2000 m/s. (a) Impulses. (b) Impulse responses when extrapolated 200 m downward from their original positions. (c) Impulse responses when extrapolated 100 m upwards from 200 m. (d) Impulses extrapolated back to their original positions. 44

2.8 Impulse responses in an inhomogeneous half space. (a) Impulses extrapolated down to depth of 100 m. (b) Impulses extrapolated down to depth of 200 m. (c) Impulses extrapolated 100 m upward from depth 200 m. (d) Impulses extrapolated back to their original positions. 46

3.1 Partitions for the Gabor wavefield extrapolation. (a) Atomic windows. Partitions given by the VGAP algorithm using a partitioning parameter of (b) 5% (from left to right), (c) 50% (from left to right) and (d) 5% (from right to left). 53

3.2 Partitions created by the PEAP algorithm for the Gabor wavefield extrapolation using various (relative) phase-error criteria. (a) 60% . (b) 40%. (c) 10% (from left to right). (d) 10% (from right to left). 60

3.3 Geometry relationships of a local ray path between the vertical and lateral components in the (a) spatial domain and (b) wavenumber domain. . 61

3.4 The trigonometric function of θ used in equation (3.18). 63

3.5 Reference velocities selected using equations (3.18), (3.23) and (3.24). Statistics of the velocities (histogram shown in red bars) from the Marmousi velocity model (Bourgeois et al., 1991) at depth of 750 m and the reference velocities (shown in blue-dashed vertical lines) selected using a lateral position error (δx) of 2.5 m. 64
3.6 The reference velocities from Figure 3.5 (blue-dashed horizontal lines) superimposed on the exact velocity profile (solid red line) from the Marmousi velocity model at depth 750 m. These reference velocities will be used as measures to create indicator functions for adaptive partitions. 65
3.7 Partitions created by the LPEAP algorithm. Partitions created using a lateral position error of (a) 15 m, (b) 10 m, (c) 5 m and (d) 2 m. 67
3.8 Partitions created by the LPEAP algorithm over the same velocity profile shown in Figures 3.1 and 3.2. The lateral position error used in this example is 2 m. 68
3.9 Comparison of samples with and without the spatial sampling. (a) Samples versus frequency: the lower the frequency, the fewer the samples in x; (b) change in samples (in percentage compared to no spatial resampling) versus frequency. Each stair-step (in blue) in (a) or (b) corresponds to a frequency band. 72

4.1 The Marmousi velocity model. The colour bar on the right codes velocities in km/s. 76
4.2 Prestack shot migrations from shot record 200 using various partitioning schemes. (a) Atomic windows. (b) Partitions from the VGAP method using a partition parameter of 5%. (c) Partitions from the PEAP method using a relative phase error of 5%. (d) Partitions from the LPEAP method using a lateral position error of 2.5 m. 80
4.3 Statistics of run times. (a) Shot migration run times from shot record 200. (b) Projected run times for all 240 shot migrations in the Marmousi data set. 81
4.4 Relative difference between depth images (from shot record 200) given by the Gabor depth migration using atomic windows and using adaptive partitions. 82
4.5 Comparison of the numbers of partitions (windows) used in various partitioning schemes in the Gabor depth migration. 83
4.6 Depth images from shot record 200: (a) Spatial resampling is not used. (b) Spatial resampling is used. The lateral position error used in the migrations is 2.5 m. 85

4.7 Prestack depth migrations with all 240 shots in the Marmousi data set. The lateral position error used in 2.5 m. (a) No spatial resampling. (b) Spatial resampling. 86
4.8 Prestack depth migrations from the Marmousi data set using lateral position errors of (a) 5 m and (b) 10 m. 88
4.9 Run times used by Gabor depth migrations with various lateral position errors. A: 2.5 m (no spatial resampling). B: 2.5 m. C: 5 m. D: 10 m. 90
4.10 The Marmousi velocity model and the constructed image given by the Gabor depth migration method. (a) Reflectivity image (from the velocity model). (b) Gabor depth image ($\delta x = 2.5\,\mathrm{m}$) of the Marmousi model. 91
4.11 Depth migrations from the Marmousi data set. (a) Gabor depth imaging using δx =2.5 m. (b) FOCI depth imaging using a 51-point operator. (c) Zoomed-in part from (a). (d) Zoomed-in part from (b). 92
4.12 Depth migrations from the Marmousi data set using Gabor depth migration schemes with various p values (1). (a) $p = 0$ (PSPI). (b) $p = 0.1$. (c) $p = 0.25$. (d) $p = 0.5$. 93
4.13 Depth migrations from the Marmousi data set using the Gabor depth migration schemes with various p values (2). (a) $p = 0.5$. (b) $p = 0.75$. (c) $p = 0.9$. (d) $p = 1$ (NSPS). 94

5.1 Step sizes created by equation (5.5). (a) Laterally variable step sizes for Model 94 topography. (b) Closeup of the region in white box shown in (a). Depth steps are color-coded in m. 102
5.2 Model 94 velocity model. The colour bar codes velocities in km/s. The dark blue regions shown on top of the pictures are the air with a velocity of 0 km/s. Two blue horizontal lines at depths about 2 km and 3 km are where the example velocity slices are selected (see Figure 5.3). . . . 104
5.3 Velocity slices from Model 94. (a) A slice adapted at depth of 2 km, from x = 4.5 km to x = 10.5 km. (b) A slice adapted at depth of 2.8 km, from x = 16.5 km to x = 23.25 km. The blue lines on top of (a) and (b) show velocities in km/s. The partitions in the lower parts of (a) and (b) in colours are created by the adaptive partitioning algorithm using lateral position errors, corresponding to various velocity segments shown above. Those partitions in both (a) and (b) always sum to 1. 104
5.4 Seismic data acquired from flat and topographic surfaces. (a) A shot record from Marmousi data set (flat surface). (b) A shot record from Model 94 data set (topographic surface). 106
5.5 Two shot records selected from Model 94 data set. (a) Shot record 30. (b) Shot record 150, a full spread containing 480 channels. 106

5.6 Selected FK spectra of the shot records from Model 94 data set. (a) FK spectrum of shot record 30. (b) FK spectrum of shot record 150. . . 107
5.7 The temporal spectra of selected traces from Model 94 data set. . . . 108
5.8 Shot migrations and reflectivity image pieces. (a) Image with shot record 30. (b) Reflectivity image piece related to shot 30 migration. (c) Image with shot record 150. (d) Reflectivity image piece related to shot 150 migration. 109
5.9 Model 94 images. (a) Stacking of 278 shot migrations. (b) Reflectivity image. 111
5.10 A wavenumber (dip) filter used in the Gabor imaging of Model 94. . 112
5.11 Dip filtering test in depth imaging of Model 94. (a) Depth image after dip filtering applied. (b) Difference taken between the image in (a) and the one in Figure 5.9 (a). 114
5.12 Depth imaging of Model 94. (a) Image with low-cut frequency filtering. (b) Image from data with direct arrivals muted. 115
5.13 Test of direct arrival muting in depth migration of shot record 150. (a) Original shot record. (b) Image of shot record in (a). (c) Shot record 150 with direct arrivals muted. (d) Image of shot record in (c). 116
5.14 Depth images of Model 94: Gabor depth imaging with both direct-arrival muting and dip filtering. 117
5.15 Topography profile for the 2D real data acquisition. 118
5.16 Two shot records from the Grizzly real data set (gaps in the shot records are bad traces removed and left blank). (a) Shot record 1. (b) Shot record 100. 120
5.17 Shot migrations from the Grizzly real data set. Shot migrations from (a) shot record 1 and (b) shot record 100, no muting applied in the shot images. (c) Muted shot image corresponding to (a). (d) Muted shot image corresponding to (b). 122
5.18 Depth imaging results of the Grizzly data set. (a) Imaging without muting in shot migrations. (b) Imaging with muting in shot migrations. . . . 123

6.1 Geometry of a local ray path in the (a) spatial domain and (b) wavenumber domain. 132
6.2 Reference velocities determined by the 2D adaptive partitioning algorithm using the lateral position error. (a) A 2D velocity slice. The colour bar on the right codes velocities in km/s. (b) A 3D view of the velocity model shown in (a). (c) Statistics of velocities from the model in (a) with reference velocities (blue-dashed lines) superimposed. (d) A 3D view of the velocity with 2D reference planes (velocities) superimposed. 135
6.3 2D atomic window (Gaussian). 137

6.4 A set of reference velocities created by the 2D adaptive partitioning algorithm using a lateral position error of 3 m. (a) A top view of the 2D bump velocity model. Velocities are coded in km/s. (b) A 3D view of the 2D bump velocity slice. (c) Statistics of the velocities in the velocity model shown in (a) or (b) with references velocities superimposed. 138
6.5 Adaptive partitioning on a bump velocity model using a lateral position error of 3 m. (a) The first partition related to the left plateau (see Figure 6.4 (b)). (b) The second partition related to the plateau in the middle. (c) The third partition related to the right plateau. (d) All partitions superimposed. (e) 2D POU. (f) 1D POU in the x direction. 139
6.6 Adaptive partitioning on a bump velocity model with random fluctuations. (a) A 2D view of the velocity model. (b) A 3D view of the velocity model. (c) Velocity statistics with the reference velocities selected by the partitioning algorithm superimposed. 140
6.7 Adaptive partitioning on a bump velocity model with random fluctuations using a lateral position error of 3 m. (a) The first partition. (b) The second partition. (c) The third partition. (d) The fourth partition. (e) Partitions superimposed. (f) 2D POU from summing all partitions. 141
6.8 1D POU's. (a) in the y direction at x = 650 m. (b) 1D POU in the x direction at y = 300 m. 142
6.9 Adaptive partitioning on the mosaic velocity model using a lateral position error of 1 m. (a) The mosaic velocity model. (b) The first partition: triangles in dark blue in (a). (c) The second partition: sky blue chunk adjacent to the lower triangle in (a). (d) The third partition: yellow chunk in (a). (e) The fourth partition: light-blue chunk at the bottom of (a). (f) The fifth partition: light-green chunk in the upper right corner of (a). (g) The sixth partition: dark-red chunk in (a). (h) All the partitions superimposed together. (i) 2D POU from summing of the 6 partitions. 143
6.10 1D POU from the 2D partitioning of the mosaic velocity model. (a) 1D POU in the y direction at x = 650 m. (b) 1D POU in the x direction at y = 300 m. 144
6.11 Modelling geometry and the velocity model: the point source is positioned at the centre of the surface. 145
6.12 3D impulse responses in a homogeneous medium. (a) Impulse before extrapolation. Impulse responses when extrapolated from surface to depths: (b) 20 m, (c) 40 m, (d) 60 m, (e) 80 m and (f) 100 m. 146
6.13 2D vertical profiles through the point source ($y = 0$) taken from 3D impulse responses shown in Figure 6.12. (a) Impulse before extrapolation. Impulse responses when extrapolated from surface to depths: (b) 20 m, (c) 40 m, (d) 60 m, (e) 80 m and (f) 100 m. 148

6.14 2D vertical profiles with varying offsets from the source point and time slices at various times from 3D impulse response shown in Figure 6.12 (f). (a) Vertical profile at offset 0 m. (b) Time slice at 52 ms. (c) Vertical profile at offset 160 m. (d) Time slice at 82 ms. (e) Vertical profile at offset 240 m. (f) Time slice at 112 ms. 149
6.15 3D impulse responses using five impulses in a homogeneous medium. (a) Impulses before extrapolation. Impulse responses at depths: (b) 20 m, (c) 40 m, (d) 60 m, (e) 80 m and (f) 100 m. 150
6.16 2D vertical profiles through the central point source ($y = 0$) taken from 3D impulse responses shown in Figure 6.15. (a) Impulses before extrapolation. Impulse responses when extrapolated from surface to depths: (b) 20 m, (c) 40 m, (d) 60 m, (e) 80 m and (f) 100 m. 152
6.17 2D vertical profiles at varying offsets from the central source point and time slices at various times from 3D impulse responses shown in Figure 6.15 (f). (a) Vertical profile at offset 0 m. (b) Time slice at 52 ms. (c) Vertical profile at offset 160 m. (d) Time slice at 82 ms. (e) Vertical profile at offset 240 m. (f) Time slice at 112 ms. 153
6.18 A horizontal slice from the 3D velocity volume. In the x direction, v = 2000 m/s, when $x \in$(210,420); otherwise, v = 4000 m/s. In the y direction, v = 2000 m/s, when $y \in$(210,420); otherwise, v = 4000 m/s. 154
6.19 3D impulse responses in an inhomogeneous medium with two distinct velocity regions indicated in Figure 6.18. Notice that the central impulse responses depart from the ones surrounding it. 155
6.20 2D vertical profiles from 3D impulse responses at a zero offset from the central point source (a) in an inhomogeneous model and (b) In a homogeneous model. 156

Chapter 1: Introduction

1.1 Necessity of seismic migrations

In seismic explorations for oil and minerals, artificial sources are used to generate seismic waves sent into the earth to illuminate the subsurface. Seismic waves reflected or diffracted from the subsurface are recorded by instruments (e.g., geophones) deployed on the surface. These recorded signals are known as *seismic data*.

Seismic data usually have to be processed before they can be used for interpretation of structures (subsurfaces) in the earth. In traditional seismic processing, we assume that the earth has horizontally layered structures or contains only mild dipping layers, and these layers are homogeneous and isotropic. If such assumptions were always true, seismic data processing would be so easy that we could obtain structural images of the earth after stacking seismic data with normal moveout (NMO) (e.g., Brown, 1969; de Bazelaire, 1988) and dip moveout (DMO) (e.g., Hale, 1984) corrections. However, these assumptions are not always true. Moreover, with other issues, such as noises, irregular surface and weathered layers with the laterally varying thickness, images related to true geological structures can be severely distorted. Therefore, seismic data without further processing may be useless.

Seismic data are processed through all major steps such as noise attenuation, deconvolution, NMO (sometimes DMO) corrections, and stacked as zero-offset [1] profiles. However, these processed data may still contain diffractions and dipping events that are not correctly positioned according to the true geological structures. As a result, further processing is needed to collapse those diffractions to their scattering points, and seismic events related

[1] Offset is the distance between a source point and a receiver point. Zero-offset means that the source and the receiver are placed at the same position.

to dipping reflectors should be moved to their correct spatial positions. These can be accomplished by *seismic migration*, which is one of the most important steps in seismic data processing.

The main purposes of seismic data processing include: enhancing signals and attenuating noises (e.g., random noises, surface waves and multiples) using stacking and other techniques, increasing temporal resolution using deconvolution, and moving dipping events to correct spatial positions and collapsing diffractions using migration (Yilmaz, 2001). Even in the early 1990s, seismic migrations were often optional, and if chosen they were final steps in seismic processing (Gray et al., 2001). However, seismic migrations have become necessary and important in seismic data processing and they have attracted more and more attention since last decade.

1.2 Evolution of seismic migration algorithms

The seismic migration concept was well established as early as in the 1920s when seismic explorations began to be used for commercial purposes; and seismic migration was originally used for structure mapping in those early days (Bednar, 2005). In the 1950s, Hagedoorn (1954) gave a comprehensive description of seismic migration using geometry of seismic rays and wavefronts, and he set a theoretical foundation for the coming seismic migration algorithms which were later implemented on computers and called *classical* seismic migration, the Kirchhoff migration (Schneider, 1978; Miller et al., 1987).

Since the late 1960s and the early 1970s, seismic migration methods had been flourishing thanks to the emergence of digitally recorded seismic data and data processing on digital computers. Seismic migration methods have been developed for earth models from

slowly dipping layers and smooth velocity variations to ones with complex geological structures and abrupt (strong) velocity variations in both vertical and lateral directions by using wave-equation based migration algorithms that can deal with multi-path problems in complex media (Gray and May, 1994). In such a historical context, we have experienced the development and application of poststack time migration, prestack time migration, poststack depth migration and prestack depth migration utilized in many ways in terms of algorithms.

Poststack time migration is the cheapest migration scheme; however, it can only deal with slowly, laterally varying velocities. To the other extreme, prestack depth migration is the most expensive but preferable method to image the earth with complex geology and strong velocity contrasts that we often encounter in today's explorations. Depth migration algorithms account for heterogeneity of the earth, however we still may have anisotropy issues. To address this, we need to integrate anisotropy parameters into the wave equation, which requires more sophisticated thus complicated migration algorithms.

Seismic migration techniques have become more advanced, and their range has been vastly broadened since the 1970s. Though seismic migration has experienced a long history, not very many numerical migration algorithms had been developed by the 1970s (Gray et al., 2001). Proliferation of numerical seismic migration algorithms resulted from advancement in computer power.

In the next section, I will review seismic migration algorithms. From there, original motivations of the research in this thesis are introduced.

1.3 Review on seismic migration algorithms

Usually seismic migration algorithms can be characterized as belonging to three main categories according to methods used to solve the scalar wave equation: integral (also known as Kirchhoff) migrations, finite-difference migrations, and frequency-wavenumber migrations (Yilmaz, 2001).

1.3.1 Kirchhoff migrations

Diffraction summation methods rest on the firm theoretical background built by Hagedoorn (1954). This method was presented in the late 1960s as the predecessor of Kirchhoff migration algorithm (Schneider, 1971). Incorporating obliquity, spherical divergence and wavelet shape factors in diffraction summation, we have Kirchhoff migration (Yilmaz, 2001). Kirchhoff migration algorithm uses rigourous formulae to fulfill the *Huygens principle* (Schneider, 1978; Baker and Copson, 1987), which is well known in physics communities.

Following French (1974) and French (1975), Kirchhoff migration was soon applied in the depth domain (Schneider, 1978). Kirchhoff migration method can also be integrated with the finite difference scheme to give a better way of calculating travel times using the eikonal equation (Gray and May, 1994).

The main advantage of the Kirchhoff migration method is its imaging efficiency compared to other methods. Another advantage of Kirchhoff migration is the flexibility of handling any subset of seismic data in various input and output domains. The Kirchhoff migration method can also treat traces with irregular spacing, which is very useful in 3D seismic imaging, where irregular trace spacing is common (Audebert et al., 1999). All

these advantages account for the reason why the Kirchhoff migration has been acting as a work horse in the industry for so many years. With a reasonable imaging accuracy, it is still the first choice when a quick turn around is needed.

The disadvantage of this method can be seen when imaging complex geological structures with highly varying velocities (Geoltrain and Brac, 1993). Extra care should be taken to obtain accurate images using Kirchhoff migration with seismic data from such complex structures.

1.3.2 Finite difference migrations

Claerbout (1971) opened another avenue for seismic migration in addition to the classic diffraction summation given before the 1970s by others. In his method, the finite difference (FD) method is used to approximate the solution to the scalar wave equation.

The Claerbout (1971) migration method is always referred to as 'wave-equation migration'. By contrast, the Kirchhoff migration is known as a migration method based on ray theories. However, both migration methods and the frequency-wavenumber migration to be introduced are all based on the wave equation.

The innovative work by Claerbout (1971) and Claerbout and Doherty (1972) is recognized as a one-way wave equation migration method for isotropic media. Their work has inspired various approaches developed by others (e.g., Pann et al., 1979; Clayton and Engquist, 1980; Schultz and Sherwood, 1980; Kosloff and Baysal, 1983; Li, 1991; Ristow and Rühl, 1997; Fei and Liner, 2008), who made the finite-difference migration method applicable in complex media (e.g., anisotropy, elastic) with both one-way and two-way schemes. And the finite-difference migration method has also been improved to be more accurate, with the ability to obtain images from higher geological dips.

The most attractive parts of the finite difference migration method are that it can handle media with strong velocity variations in a natural way, and the algorithm is straightforward to code on computers by using the explicit finite difference. The explicit finite-difference scheme is not unconditional stable (Ji and Biondi, 1991). So this method usually requires smaller grid sizes (in both time and space) than needed in theory to enable stable calculations. Moreover, to prevent numerical dispersion, one always needs to set grid sizes even smaller, which contributes extra computational cost to the migration process.

1.3.3 Frequency-wavenumber-domain migrations

Seismic data sometimes become easier to process in the frequency-wavenumber domain. Shortly after the occurrence of Kirchhoff and finite difference migrations, Stolt (1978) proposed a migration method using the Fourier transform. The Stolt (1978) migration uses the Fourier transform to calculate the spatial and temporal derivatives in the wave equation, which is very fast, accurate and less memory demanding during the migration. However, this method only works for homogeneous media. At the same time, Gazdag (1978) developed a similar method called phase-shift migration, also implemented in the frequency-wavenumber domain, but this method is capable of migrating seismic data related to layered earth models (velocity varying vertically but not laterally).

To overcome the difficulty from the laterally varying velocity structure, Gazdag and Sguazzero (1984) extended the phase-shift migration to accommodate the laterally varying velocity, known as phase-shift-plus-interpolation (PSPI) method. In this method, a set of constant velocities are used to carry out the constant-velocity phase shift or wavefield extrapolation in depth, then the wavefield at an output position with certain velocity is obtained using the interpolation between those wavefields calculated with the reference

velocities.

Following Stolt (1978) and Gazdag (1978), various phase-shift migration methods were given in the frequency-wavenumber domain (e.g., Stoffa et al., 1990; Wu and Huang, 1992; Roberts et al., 1997; Margrave and Ferguson, 1999; Rousseau and de Hoop, 2001; Jin et al., 2002; Grossman et al., 2002a; Ma and Margrave, 2005). These developments give a better handling of the lateral velocity variation. Two main steps in the phase-shift method are: phase shift using a constant velocity field and choosing a set of appropriate reference velocities. The phase-shift part is very straightforward to apply in the frequency-wavenumber domain with a constant velocity using the Gazdag (1978) method. However, dealing with the laterally varying velocity structure is not as easy as dealing with the phase shift and actually it is almost always cumbersome. This is where the effort has been made in those phase-shift migration methods.

Since we assume a constant velocity, the scalar wave equation can be solved in the frequency-wavenumber (FK) domain (see section 2.2.3 in Chapter 2). In the FK domain, direct calculation of temporal and spatial derivatives contained in the original wave equation is avoided. Computational cost in this migration method is proportional to the number of Fourier transforms. However, helped by the fast Fourier transform (FFT), we can make the frequency-wavenumber migration method competitively fast.

1.3.4 Recent developments and trends of seismic migrations

Seismic migration has a history of nearly a century: from basic geometry mapping by hand to sophisticated migration algorithms implemented on modern computer systems. The development of new migration algorithms never stops. Moreover, some ‘old’ algorithms may be found more attractive than they used to be thanks to increasing computing power.

Such an example can be seen from the 'reverse time' migration (e.g., Baysal et al., 1983; Chang and McMechan, 1987; Dong and McMechan, 1993; Chang and McMechan, 1994), which used to be prohibitive due to limited and expensive computing resources. Since 2004 or so, this two-way wave-equation migration method has become a hot topic in the industry because it can image the most complex depth velocity models (e.g., steep salt dome flanks) for which other methods have had difficulty obtaining images.

Migration methods are bound to be pushed to 3D, anisotropic, elastic and prestack depth migrations. Applications of migrations are not only restricted to imaging areas, but are also used for the amplitude analysis and the residual velocity analysis (e.g., Schleicher et al., 2008).

Seismic migration has been a very active research area in seismic data processing, and it will be so in the future, not only seismic migration itself but also other areas with seismic migration incorporated.

1.4 Motivations of the research in this thesis

Depth migrations with the Gabor depth imaging theories are in the category of frequency-wavenumber-domain migration methods. The main reasons that the Gabor depth imaging theories were chosen as the theoretical basis for an investigation into depth migration methods are as follows:

First of all, the theoretical research on deconvolution and depth imaging using the Gabor transform was carried out by Margrave and Lamoureux (2001) and Grossman et al. (2002a), and the theories have been shown a solid background for the depth imaging. This gave a good start for the research project because the application of the Gabor depth imag-

ing theories would be in a reasonable time frame. The work done by Grossman et al. (2002b) had shown that the Gabor imaging method would be very promising. This created a vision that the Gabor depth imaging theories would introduce an elegant frequency-wavenumber-domain depth migration algorithm with very good applications in the practical depth migration.

Another one is that phase-shift migrations are very good and attractive alternatives among wave-equation migration methods, with many potential areas that are worthy of research effort.

These reasons motivated the research project in this thesis.

1.5 Objective of the thesis

I set up the objective for the thesis as: exploring the Gabor imaging theories and extending the theories with new adaptive partitioning algorithms and other possible algorithms to obtain better depth migration schemes in applications. In order to reach the objective, I had planed the following developments and experiments:

- To explore Gabor imaging theories;
- To find possible optimal ways of implementation for Gabor depth imaging theories;
- To develop 1D adaptive partitioning algorithms for 2D Gabor depth migrations;
- To test and verify 1D adaptive partitioning algorithms;
- To control imaging accuracy and efficiency with 1D adaptive partitioning algorithms;

- To improve Gabor depth migration speed using the spatial resampling;
- To test Gabor depth migration algorithms with benchmark synthetic data sets;
- To verify effectiveness of the Gabor depth migration method;
- To develop 2D Gabor depth migration scheme for topographic seismic data;
- To test the Gabor depth migration method with 2D topographic data set;
- To develop 3D Gabor depth migration schemes;
- To develop a 2D adaptive partitioning algorithm for 3D Gabor depth migration;
- To test the 2D adaptive partitioning method;
- To test 3D Gabor wavefield extrapolator.

1.6 Claim of originality

Original contributions from this thesis research are listed in the following:

- 1D adaptive partitioning algorithm with phase errors;
- 1D adaptive partitioning algorithm with lateral position errors;
- Fast Gabor depth migration (2D) using 1D adaptive partitioning algorithms;
- Application of the spatial resampling in the Gabor depth migration method;
- Gabor depth migration schemes using intermediate p values;
- High-quality Marmousi depth images using the Gabor depth migration method;

- Gabor depth migration scheme for topographic seismic data;
- High-quality topographic depth images given by the Gabor depth migration method;
- 2D adaptive partitioning algorithm using lateral position errors;
- 3D Gabor depth imaging theory development;
- 3D Gabor wavefield extrapolation test.

1.7 Summary of the thesis

This thesis is focused on research and development of a phase-shift depth migration method using the Gabor imaging theories. Starting from the locally homogeneous approximation (LHA) (Fishman and McCoy, 1985), an approximation to the LHA formula is developed with the Gabor transform over the spatial coordinate.

Since the LHA formula uses exact velocities at output positions[2], it gives the most accurate imaging results over all phase-shift methods. However, the LHA method may be very inefficient when imaging with industry standard seismic data sets. The Gabor depth imaging method is developed to approximate the LHA, which can give a fast depth imaging method with an acceptable accuracy. In its simplest form, Gabor imaging method uses small, evenly distributed partitions (spatial windows) to localize the wavefield (e.g., seismic data recorded on the surface). Those partitions are set small enough to address the most rapidly varying lateral velocities, which does not make the Gabor depth imaging speed acceptable.

[2]Actually, it is a locally homogeneous approximation of the velocity about a neighbouring area of the output position.

Given an accuracy threshold, partitions may be consolidated when lateral velocity variations are small, resulting in fewer and wider partitions. Adaptive partitioning algorithms are used to create partitions adapted to the variation in lateral velocity. In this thesis three adaptive partitioning algorithms are described. These algorithms all address the problem very well and give efficient Gabor depth imaging.

Spatial resampling (Margrave et al., 2006) is another way to improve the Gabor depth imaging efficiency together with adaptive partitioning algorithms. The spatial resampling method improves imaging speed by reducing spatial samples at low frequencies, resulting in shorter Fourier (Gabor) transforms that can be processed fast.

With the efficiency-improved Gabor depth migration method, imaging tests on 2D synthetic and real data sets are carried out. The Gabor depth migration method is also developed for topographic seismic data. All depth migrations can give high-quality images.

In the final parts of this thesis, I extend Gabor depth imaging theories to 3D. A 2D adaptive partitioning algorithm is also developed and tested for the 3D Gabor wavefield extrapolation. The 3D Gabor wavefield extrapolator is tested.

The applications of the Gabor depth imaging theories in 2D seismic data sets are concluded in the end. Future research directions are suggested for possible research effort.

1.8 Summary of chapters

Chapter 1 (this chapter) begins with the necessity of seismic migrations. Reviews on seismic migration history and seismic migration methods give readers the whole picture of the seismic migration. Then the motivations of the research and the objective of the thesis are described. Finally, summaries of chapters give the contents of all chapters.

Chapter 2 starts with an introduction on explicit wavefield extrapolation (EWE). Then phase-shift wavefield extrapolation formulae are derived by solving a 2D wave equation using the Fourier transform. From the derived wavefield extrapolation formulae for a homogeneous medium, the formulae are generalized as one that holds in the locally homogenous media. Introduction of the Gabor wavefield extrapolation is actually an approximation to the generalized formula (known as the generalized phase shift plus interpolation, see Chapter 2). Then the Gabor wavefield extrapolation theories are explored, giving formulae simulating a variety of extrapolation schemes. After the introduction of the Gabor imaging theories, I describe how to implement the Gabor wavefield extrapolation simulating the PSPI method as an example. Impulse responses show that the 2D Gabor wavefield extrapolator works as expected.

In Chapter 3, two methods are introduced to improve the efficiency of the Gabor wavefield extrapolation. The first one uses adaptive partitioning algorithms to consolidate small partitions where the lateral velocity variation is small given an accuracy criterion. Application of the adaptive partitioning algorithms intends to reduce redundant partitions, hence the computational cost will be minimized. The second method is the spatial resampling, which uses fewer spatial samples at low (temporal) frequencies. This application results in shorter Fourier transforms and then fast imaging process. Examples are demonstrated to compare the efficiency improvement both in partitioning and the imaging speed.

Based on the fast Gabor depth imaging scheme developed in Chapter 3, Chapter 4 shows 2D depth imaging examples from a benchmark seismic data. These examples show that the Gabor depth imaging method gives accurate depth images from complex velocity structures, such as the Marmousi model (Bourgeois et al., 1991). The imaging examples from the Marmousi data also show that we can trade off between the imaging efficiency

and accuracy using the Gabor depth migration method.

Chapter 5 is devoted to depth imaging with topographic seismic data. The Gabor depth migration method is modified to address such an issue. Imaging results from both a synthetic data set and a real data set are shown. In both applications, Gabor depth imaging method gives very good imaging results.

Chapter 6 describes Gabor depth imaging theories in 3D. To improve the imaging efficiency, a 2D adaptive partitioning algorithm is developed, using the lateral position error as the accuracy criterion. The 2D adaptive partitioning algorithm is tested with various velocity models. Also 3D Gabor extrapolator is tested.

Chapter 7 gives general conclusions on the Gabor depth migration method and the depth imaging results given with 2D seismic data. Some suggestions on future research directions are suggested.

Chapter 2: Explicit Wavefield Extrapolation and the Gabor Transform

As reviewed in Chapter 1, there are many ways to carry out seismic migrations. In this chapter, I will introduce a wave-equation based depth migration method which uses the phase shift method and falls into a category called *explicit wavefield extrapolation*. Explicit wavefield extrapolation is usually implemented in the frequency-wavenumber domain.The migration method in question is a frequency-wavenumber-domain depth migration method.

2.1 Explicit wavefield extrapolation

Explicit wavefield extrapolation (EWE) is a tool that can be used to extrapolate seismic wavefields recorded on the surface (seismic data) back into depths. The EWE becomes an attractive depth migration method because the method is easy to extend for 3D wavefield extrapolation (Berkhout, 1981; Hale, 1991). Explicit wavefield extrapolation is most easily implemented in a fashion known as recursive depth marching.

A recursive depth marching scheme consists of a number of steps. Each step provides part of the final picture but also provides input for wavefield extrapolation to the next step. In this way, extrapolation can be applied repeatedly to go deeper and deeper. The recursive process continues until depth reaches its maximum. Figure 2.1 shows the recursive depth marching scheme. From surface z_0, seismic data extrapolation begins. With the explicit wavefield extrapolation, seismic data can be extrapolated to depths $z_1, z_2, ..., z_n$. The cartoon shown in Figure 2.1 demonstrates straight ray paths, indicating that the velocity has

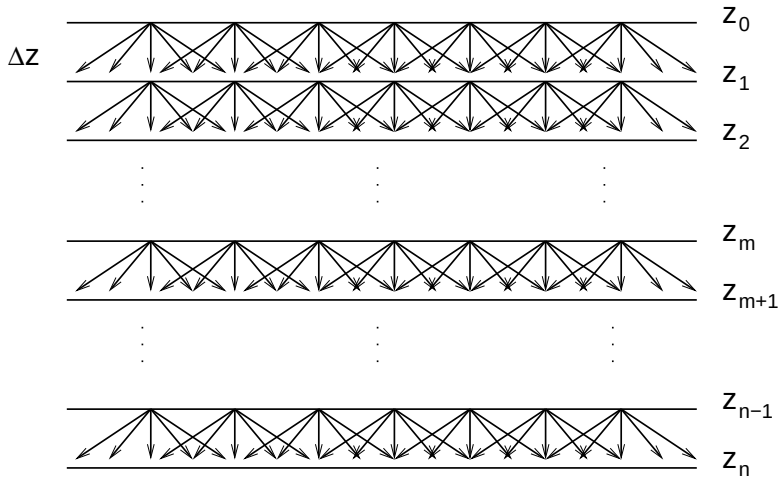


Figure 2.1: Explicit wavefield extrapolation using recursive depth marching scheme. Δz often denotes a small step (e.g., 10 m).

no lateral variations. In depth migration, laterally varying velocity structures are common. So these rays should be curved according to heterogeneity in the layers. In an explicit wavefield extrapolation, the lateral velocity variation is less cumbersome to deal with.

One difficulty from the explicit wavefield extrapolation is the stability of the extrapolation operator (Hale, 1991), also called the wavefield extrapolator. The magnitude of the wavefield extrapolator (see definition in equation (2.19)) is usually a unit (one) and ideally keeps unit during extrapolations. However, it can get larger than one due to numerical errors caused by the discrete Fourier transform during wavefield extrapolation. As a result, the amplitude grows when marching down to depths. Not surprisingly, the process breaks down very quickly during the extrapolation. The instability can be controlled with careful design of the wavefield extrapolator. Margrave et al. (2004, 2006) developed techniques

to overcome this difficulty and applied their explicit extrapolation operators in the space-frequency domain for prestack depth migrations. The method, known as *forward operator and conjugate inverse* (FOCI), uses the inverse operator to compensate for the growth of amplitudes in applying the forward extrapolation operator. In this way, the operator can be made fairly stable across several hundreds of extrapolation steps, and makes a very fast depth imaging method.

In the next section, I will introduce explicit wavefield extrapolation using phase-shift methods.

2.2 Explicit wavefield extrapolation with phase shift

2.2.1 Review of phase-shift methods

Wavefield extrapolation with phase shift, or also called seismic migration by phase shift (Gazdag, 1978), was proposed as an accurate and efficient method that is unconditionally stable provided that there are no lateral velocity variations. An approximate extension to lateral velocity variations, called PSPI (phase-shift-plus-interpolation), was presented by Gazdag and Sguazzero (1984), who proposed spatial interpolation among a set of extrapolated wavefields corresponding to reference velocities. This method has proven popular and, although it is no longer considered to be unconditionally stable (e.g., Etgen, 1994), it is more stable than a typical explicit space-frequency method (Margrave et al., 2006). Other ways of extending the concept of a phase shift include the split-step Fourier method (Stoffa et al., 1990), the various phase-screen methods (Wu and Huang, 1992; Roberts et al., 1997; Rousseau and de Hoop, 2001; Jin et al., 2002), the generalized phase-shift-plus-interpolation (GPSPI) and nonstationary phase-shift methods (NSPS) (Margrave and

Ferguson, 1999), and the Gabor method (Grossman et al., 2002a; Ma and Margrave, 2005). Margrave and Ferguson (1999) showed that the GPSPI Fourier integral is the limit of PSPI in the extreme case of using a distinct reference velocity for each output location. Writing outside the typical seismic literature, Fishman and McCoy (1985) derived the GPSPI formula as a high frequency approximation to the exact wavefield extrapolator for a laterally variable medium. They refer to the GPSPI formula as the locally homogeneous approximation (LHA) and I adopt that nomenclature. While the LHA is called a high frequency approximation it is still much more accurate than raytracing because the approximation is done at a different place in the theoretical development (see Fishman and McCoy for more discussion). In fact, it can be shown that virtually all explicit depth-stepping methods in practice today, including those mentioned above, are approximations to the LHA formula.

Since the Fourier transform is commonly used in the wavefield extrapolation with phase shift, I will introduce the Fourier transform in the next section.

2.2.2 Fourier transforms

Usually the Fourier transform is defined in the time and frequency domain. However, from a mathematical point of view, it can also be described in the space and wavenumber (spatial-frequency) domain. Fourier transforms in time with the following conventions (e.g., Sneddon, 1951; Papoulis, 1962) are used in this thesis. The forward Fourier transform in time is defined as

$$F(\omega) = \int_{\mathbb{R}} f(t) \exp\left(-i\omega t\right) \mathrm{d}t, \tag{2.1}$$

and the inverse Fourier transform is given by

$$f(t) = \frac{1}{2\pi} \int_{\mathbb{R}} F(\omega) \exp(i\omega t) \mathrm{d}\omega, \tag{2.2}$$

where t is the time coordinate, ω is the frequency coordinate, F is the Fourier transform of f, $\mathbb{R}$ is the real line for integration. The exponential parts, $\exp(-i\omega t)$ and $\exp(i\omega t)$, in equations (2.1) and (2.2) are known as Fourier kernels.

Equations (2.1) and (2.2) appear as a temporal Fourier transform pair. When the pair is used for spatial Fourier transforms, the signs in Fourier kernels should be changed into opposite ones. That is, in the forward spatial Fourier transform, Fourier kernel uses positive sign; and in the inverse spatial Fourier transform, Fourier kernel uses negative sign.

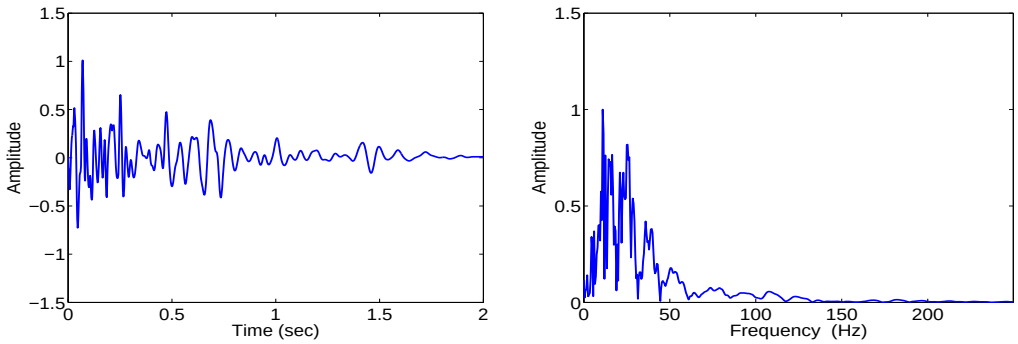
The discrete versions of temporal Fourier transforms are represented as

$$F_m = \sum_{n\in\mathbb{Z}} f_n \exp(-2\pi inm/N) \tag{2.3}$$

and

$$f_n = \frac{1}{N}\sum_{m\in\mathbb{Z}} F_m \exp(2\pi inm/N). \tag{2.4}$$

Figure 2.2 (a) shows a time series f_n, simulating a seismic trace; Figure 2.2 (b) shows



(a) (b)

Figure 2.2: A Fourier transform of a seismic trace. (a) Input signals. (b) Amplitude spectrum.

absolute values of F_m, computed from the discrete Fourier transform (DFT) shown in equation (2.3).

Following the introduction of the Fourier transform, I will show how to develop wavefield extrapolation formulae in a homogeneous medium. From there, I will introduce wavefield extrapolation in a medium with vertically varying velocity structure, and finally get to the method that performs wavefield extrapolation in a medium with both vertical and lateral velocity variations.

2.2.3 Wavefield extrapolation with the phase-shift method

In this section, I derive wavefield extrapolation in 2D. 3D wavefield extrapolation theory and the corresponding Gabor imaging method will be described in Chapter 6.

As we will see in this chapter and Chapter 6, from 2D to 3D, the wave equation is simply added into one extra dimension (y in the spatial domain and k_y in the wavenumber domain). Without a rigorous proof in math, the simple extending from 2D to 3D may be problematic. It is well known in mathematical physics that the *Huygen's* principle holds strictly in odd dimensions (e.g., 3 dimensions), it does not in even dimensions (Courant and Hilbert, 1962, p. 690).

In 3D, we have *spherical* waves, which propagate from a singular (source) point and build up spherical wavefronts (appearing as sharp impulses) according to the Huygen's principle. We observe distinctly sharp impulses on various concentric shells about the singular point. A 2D wave equation describes *cylindrical* waves radiating from an infinitely long line (in y direction in this thesis). Impulses can be observed on cylindrical surfaces, but they are not distinctly sharp in that they have tails, which are contribution from point sources along the axis of cylindrical shells. Hence, Huygen's principle does not hold strictly in 2D. However, the approximation is still very good in seismic applications. As a result, we can always extend the wave equation directly from 2D to 3D for the

wavefield extrapolation. Seismic wavefield extrapolation formulae in turn can be extended straightforward to 3D from its 2D versions.

Starting with a scalar (acoustic) wave equation (e.g. Berkhout, 1981), we have

$$\frac{\partial^2 \psi}{\partial z^2} + \frac{\partial^2 \psi}{\partial x^2} - \frac{1}{v^2}\frac{\partial^2 \psi}{\partial t^2} = 0, \tag{2.5}$$

where x and z are the spatial coordinates with z increasing downwards, x increasing to the right, t is the time coordinate, $\psi(x, z, t)$ is the pressure in the spatial-time domain, and v is the seismic velocity (wave propagation speed).

The problem we try to solve for the wavefield extrapolation here is to deduce the wavefield at depth z ($\psi(x, z, t)$) given the wavefield at $z = 0$, i.e., $\psi(x, 0, t)$. Such a wavefield extrapolation problem is equivalent to solving the second order partial differential equation (PDE) (2.5) given boundary values regarding ψ (Dirichlet) and the first order derivatives (Neumann) of ψ (e.g., Arfken, 1970), which can be written as

$$\psi(x, z, t)|_S = f_0(x, z, t) \tag{2.6}$$

and

$$\frac{\partial \psi}{\partial n}\Big|_S = f_1(x, z, t), \tag{2.7}$$

where S is a surface, n denotes the normal to the surface S, and f_0 and f_1 are functions specifying boundary values of function ψ and the first order derivative of ψ on S. These two kinds of boundary values can be used to determine the solution to equation (2.5). We may find many ways to solve equation (2.5) in a typical math textbook on PDE's. I choose the method using the Fourier transform in this thesis. The main idea behind the chosen method is that a PDE can be transformed into an ordinary differential equation (ODE) by the Fourier transform, and we can use techniques for solving ODE's to obtain the solution to a PDE.

Taking temporal Fourier transforms (see equations (2.1) and (2.2)) on both sides of equation (2.5) gives

$$\frac{\partial^2 \Psi}{\partial z^2} + \frac{\partial^2 \Psi}{\partial x^2} + k^2 \Psi = 0, \tag{2.8}$$

where Ψ is the temporal Fourier transform of the pressure field (or wavefield) ψ using the convention shown in equation (2.1) and k is the spatial frequency, also called total (scalar) wavenumber defined as

$$k = \frac{\omega}{v}. \tag{2.9}$$

From equation (2.5) to equation (2.8), a property of the Fourier transform of the time derivative has been used. That is, Fourier transform of the second order derivative in time is equivalent to the squared product of the imaginary (i) unit and the frequency coordinate (ω) in the Fourier domain, which can be written as

$$\mathscr{F}\left(\frac{\partial^2 \psi}{\partial t^2}\right) = (i\omega)^2 \Psi, \tag{2.10}$$

where $\mathscr{F}$ stands for the abstract symbol of the (forward) Fourier transform. Similar rules will apply in the forward Fourier transform on spatial derivatives in the following derivations.

Performing forward spatial Fourier transforms with respect to x on both sides of equation (2.8) and rearranging yields

$$\frac{\partial^2 \hat{\Psi}}{\partial z^2} + (k^2 - k_x^2)\hat{\Psi} = 0, \tag{2.11}$$

where $\hat{\Psi}$ is the wavefield in the frequency-wavenumber domain. Written in an explicit formula using the conventions of Fourier transforms defined in the previous section, the wavefield $\psi(x, z, t)$ in the frequency-wavenumber domain is

$$\hat{\Psi}(k_x, z, \omega) = \int_{\mathbb{R}^2} \psi(x, z, t) \exp(ik_x x - i\omega t) \mathrm{d}t \mathrm{d}x, \tag{2.12}$$

where $\mathbb{R}^2$ denotes the real plane (x, t) and k_x is the wavenumber coordinate related to x in the spatial domain.

Defined in terms of the total wavenumber k and k_x, the wavenumber component corresponding to the spatial coordinate z is

$$k_z = \begin{cases} \sqrt{k^2 - k_x^2}, & k^2 \geqslant k_x^2 \\ i\sqrt{k_x^2 - k^2}, & k^2 < k_x^2, \end{cases} \tag{2.13}$$

where the square-root symbol denotes explicitly the positive square root, the first part specifies the propagating wavefield, while the second part denotes the evanescent region; and the sign choice for the evanescent region assures exponential decay not exponential growth in the z direction. Using (2.9) in (2.13) immediately yields

$$k_z = \begin{cases} \sqrt{\frac{\omega^2}{v^2} - k_x^2}, & \frac{\omega^2}{v^2} \geqslant k_x^2 \\ i\sqrt{k_x^2 - \frac{\omega^2}{v^2}}, & \frac{\omega^2}{v^2} < k_x^2. \end{cases} \tag{2.14}$$

Using equation (2.13) in equation (2.11) gives

$$\frac{\partial^2 \hat{\Psi}}{\partial z^2} + k_z^2 \hat{\Psi} = 0. \tag{2.15}$$

Equation (2.15) is an ODE, which means that we have only one variable (z) when solving the differential equation. Solving (2.15) is straightforward using ODE solution techniques. The general solution to the ODE is

$$\hat{\Psi}(k_x, z, \omega) = A \exp(ik_z z) + B \exp(-ik_z z), \tag{2.16}$$

where A and B are arbitrary functions of k_x and ω. Given our definitions of the Fourier transforms, A and B have the physical interpretations as the strengths of upgoing and downgoing waves, respectively; they can be determined by Dirichlet and Neumann boundary conditions. However, usually we don't have the second boundary condition (the first order derivative of the wavefield) in the wavefield extrapolation.

In equation (2.16), since A represents upward traveling waves and B is downward traveling waves, the second term is usually discarded due to the assumption that our data consists only of upward traveling waves. We know this is not true but it is forced on us because we don't have the second boundary condition. As a result of this choice, equation (2.16) is reduced to

$$\hat{\Psi}(k_x, z, \omega) = A \exp(ik_z z). \tag{2.17}$$

Solution (2.17) requires only one boundary condition, the Dirichlet boundary condition. In our seismic applications, we set Dirichlet boundary values as seismic data recorded on the surface ($\psi(x, z = 0, t)$). Setting $z = 0$ in equation (2.17), we obtain $A = \hat{\Psi}(k_x, z = 0, \omega)$, which can be defined by equation (2.12) using $z = 0$. Putting constant A back in (2.17) yields

$$\hat{\Psi}(k_x, z, \omega) = \hat{\Psi}(k_x, 0, \omega) \exp(ik_z z). \tag{2.18}$$

Given boundary values, equation (2.18) can be used to calculate the wavefield at depth z in an analytic way. Usually, we write

$$\hat{W}\left(k_x, z, \omega\right) = \begin{cases} \exp\left(iz\sqrt{\frac{\omega^2}{v^2} - k_x^2}\right), & \frac{\omega^2}{v^2} \geqslant k_x^2 \\ \exp\left(-z\sqrt{k_x^2 - \frac{\omega^2}{v^2}}\right), & \frac{\omega^2}{v^2} < k_x^2. \end{cases} \tag{2.19}$$

$\hat{W}$ is called the wavefield extrapolator, a spatial phase-shift term.

To get back to the spatial domain, we can use the inverse spatial Fourier transform on both sides of equation (2.18). We have

$$\Psi(x, z, \omega) = \frac{1}{2\pi} \int_{\mathbb{R}} \hat{\Psi}(k_x, 0, \omega) \exp(ik_z z) \exp(-ik_x x) \mathrm{d}k_x. \tag{2.20}$$

Or we can write (2.20) as

$$\Psi(x, z, \omega) = \frac{1}{2\pi} \int_{\mathbb{R}} \hat{\Psi}(k_x, 0, \omega) \hat{W}\left(k_x, z, \omega\right) \exp(-ik_x x) \mathrm{d}k_x. \tag{2.21}$$

However, this analytic solution is only valid in a homogeneous medium, meaning v is constant. Otherwise, the procedures described above are not strictly applicable to solve the PDE (2.5).

In a homogeneous medium, equation (2.21) can be used to calculate a wavefield to any depth z given boundary values on the surface $z = 0$. If we set z as small as Δz, usually a step size in the wavefield extrapolation, we have a thin slab model, which means Δz is small compared to the lateral coordinate x. Inside such a slab, equation (2.5) still holds; an analytical solution to such a PDE can be found in a similar way as described above, which is

$$\Psi(x, \Delta z, \omega) = \frac{1}{2\pi} \int_{\mathbb{R}} \hat{\Psi}(k_x, 0, \omega) \hat{W}\left(k_x, \Delta z, \omega\right) \exp(-ik_x x) \mathrm{d}k_x. \tag{2.22}$$

Actually, equation (2.21) and equation (2.22) are exactly the same as each other from a mathematical point of view.

In a realistic model, we have not only velocity variations from layer to layer in the z direction, but also the velocity variations in the x direction inside each layer. This gives difficulty to the method when derived for solving PDE's in a homogenous medium or in a homogenous thin layer. However, an approximate solution can still be found using a method called generalized phase shift plus interpolation (GPSPI) (Margrave and Ferguson, 1999), which approximates the velocity as constant in the neighbourhood of a distinct output position (x, z). In such a case, PDE's still hold inside neighbourhoods of output positions. Detailed discussion of the GPSPI extrapolation formula will be presented in section 2.4.

The imaging theory to be described in this thesis is a phase-shift method as one of those approximations to the GPSPI formula. The Gabor transform, also called the win-

dowed Fourier transform (Gröchenig, 2001), is employed to carry out such a phase-shift method. Before we get into the wavefield extrapolation theory using the Gabor transform, I will give an introduction to the Gabor transform, which is closely related to the Fourier transform. In the next section, I will show how we get to the Gabor transform from the Fourier transform.

2.3 Gabor transforms (windowed Fourier transforms)

2.3.1 Continuous Gabor transforms

If time shifting windowing functions are inserted into the Fourier transform shown in equation (2.1), they give

$$F(\omega, t') = \int_{\mathbb{R}} f(t) g(t - t') \exp(-i\omega t)\, \mathrm{d}t, \tag{2.23}$$

where $g(t - t')$ is the shifted window and can be any kind of a continual math function that can localize the input signal $f(t)$. The localizing function is also characterized by a variable, t', which is recognized as the center of the shifting window. Equation (2.1) gives a spectrum of the input function $f(t)$ in the frequency domain. However, by using the shifting window in equation (2.23), we can have a modified version of the input signal (e.g., a seismic trace) in the time domain and then use a Fourier transform to obtain the spectrum of it. If we change the center into various positions along the time axis, we will have a variety of spectra for the localized versions of the input signal, which is $g(t-t')f(t)$. Usually, the shifting functions are usually set as functions with localizing properties. Such functions as Gaussian and Hanning windows can be good candidates for localizing. In this thesis, Gaussian windows are selected as the base windows (called *atomic* windows

in Chapter 3) and they are proven good windows for the Gabor transform, which will be introduced shortly. If $g(t-t')$ in equation (2.23) is such a localizing window with a short time extension, then $g(t-t')f(t)$ represents small pieces of the input signal $f(t)$ related to local times. $F(\omega, t')$ will be Fourier spectra of those pieces of the input signal $f(t)$. As a result, equation (2.23) is called the short-time (forward) Fourier transform (Gröchenig, 2001) or the windowed (forward) Fourier transform. The windowed (forward) Fourier transform is also known as the (forward) Gabor transform.

To recover the original signal from its localized spectra, we use the inverse Gabor transform. In our case, Gabor transforms are in the spatial-wavenumber domain. Put in general terms for our use, the Gabor transform pair can be written as (following Margrave and Lamoureux, 2001)

$$V_g\psi(x', k_x) = \int_{\mathbb{R}} \psi(x) g(x-x') \exp(ik_x x) \mathrm{d}x \tag{2.24}$$

and

$$\psi(x) = \int_{\mathbb{R}^2} V_g\psi(x', k_x) \gamma(x-x') \exp(-ik_x x) \mathrm{d}k_x \mathrm{d}x', \tag{2.25}$$

where x denotes the lateral coordinate, $\psi(x)$ is the input signal or, in this thesis, the input wavefield, $V_g\psi(x', k_x)$ is called the Gabor spectrum (a set of Fourier spectra) of $\psi(x)$, and k_x is the coordinate in the wavenumber domain related to x in the spatial domain.

To give a brief and clear development of the Gabor imaging theories in the following sections, I will use $\mathscr{F}$ and $\mathscr{F}^{-1}$ to denote the abstract operator symbols for the forward and inverse Fourier transforms, respectively.

Equation (2.24) is used to calculate the Gabor spectrum of $\psi(x)$. In order to recover the original signal $\psi(x)$ from its Gabor spectrum $V_g\psi(x', k_x)$, analysis and synthesis windows

must satisfy

$$\int_{\mathbb{R}} g(x)\gamma(x)\mathrm{d}x = 1 \tag{2.26}$$

(Margrave and Lamoureux, 2001), which is called *partition of unity* (POU). The analysis window could be quite general mathematical functions. In our wavefield extrapolation applications, we choose functions with a localization property. In this way, we may represent our wavefield extrapolator depending on local velocities. Gaussian windows are good candidates, and we have chosen them for this thesis.

2.3.2 Discrete Gabor transforms

The discrete versions of Gabor transforms (DGT) and POU (see equations (2.24), (2.25) and (2.26)) are expressed as (after Margrave and Lamoureux, 2001)

$$\hat{\psi}_{l,m} = \sum_{n\in\mathbb{Z}} \psi_n g_{l,n} \exp(2\pi i n m/N), \tag{2.27}$$

$$\psi_n = \sum_{l\in\mathbb{Z}} \sum_{m\in\mathbb{Z}} \hat{\psi}_{l,m} \gamma_{l,n} \exp(-2\pi i n m/N) \tag{2.28}$$

and

$$\sum_{l\in\mathbb{Z}} g_l \gamma_l = 1, \tag{2.29}$$

where l is associated with centres of windows, m is the index corresponding to the Fourier domain coordinate k_x, n is the index of the x coordinate, and N is the total number of discrete samples of ψ_n.

Before exploring the Gabor wavefield extrapolation theory, I give numerical examples from the discrete Gabor transform presented by equations (2.27) and (2.28). Figure 2.3 (a) shows a simulated seismic trace that is the same one as in Figure 2.2 (a). Figure 2.3 (b) shows the Gabor (amplitude) spectrum from the input signal, with amplitude values

increasing from colour blue to colour red. In Figure 2.3 (b), the horizontal axis denotes coordinates related to the centres of shifted windows used in the forward Gabor transform; and the vertical axis is the frequency. Figure 2.3 (c) is the signal recovered from the Gabor spectrum by the inverse Gabor transform. Comparing this with the input signal shown in Figure 2.3 (a), we can hardly tell the difference between the two using visual observation. A difference between the input and recovered signals is shown in Figure 2.3 (d), from which we can tell that the difference is negligible compared to amplitudes from both the original and recovered signals shown in Figure 2.3 (a) and (c). However, this difference is due to the particular implementation in MATLAB®, which uses an approximate POU. The difference can be made zero with another implementation, which is critical to the wavefield extrapolation when using the Gabor transform. That is, the input signal (wavefield) will not be changed when only forward and inverse Gabor transforms are applied to it in a sequence.

2.4 Explicit wavefield extrapolation with the Gabor transform

In the previous section, I showed how we get to the Gabor transform from the Fourier transform. Before that, I introduced wavefield extrapolation with phase-shift methods. In this section, I will continue with an explicit wavefield extrapolation in 2D using the Gabor transform.

As stated in the previous sections, GPSPI (or LHA) is the theoretical limit of all the phase-shift methods. The phase-shift method with the Gabor transform method is one of them. The Gabor depth imaging method approximates the LHA formula and is described in Grossman et al. (2002a) and Ma and Margrave (2005). This depth imaging method is

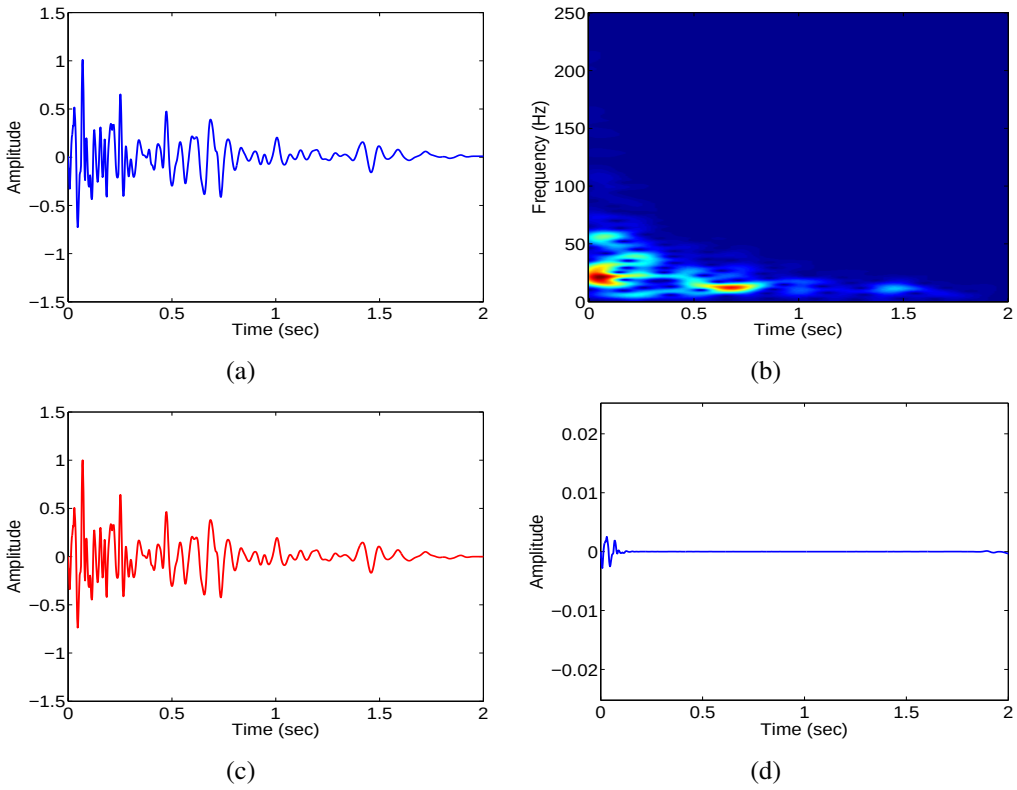


Figure 2.3: Numerical tests with Gabor transforms. (a) Input signal (the same one as shown in Figure 2.2). (b) Gabor spectrum of the input signal. (c) Recovered signal from the Gabor spectrum shown in (b) using the inverse Gabor transform. (d) Difference between the original (a) and the recovered (c) signals.

also closely related to the windowed screen method described in Jin and Wu (1998). The method to be presented is derived in a new way by seeking a Gabor, or windowed Fourier approximation, to the LHA formula. I will show in this chapter that this formulation as a Gabor method suggests a great variety of possible algorithms. Imaging examples from these algorithms will be shown in Chapter 4 with a synthetic seismic data set.

Equation (2.22) is obtained as the solution to the PDE (2.5) in a homogeneous thin layer with a thickness of Δz. If set in an inhomogeneous medium, with only the velocity variation but no (or very small) variations in density in x, wave equation (2.5) still holds (Červený, 2001, equations (2.1.24) and (2.1.25), p. 15) in such a medium. In this thesis, seismic waves in question are acoustic waves with an assumption of a constant density. Therefore, acoustic wave equation regarding a velocity-varying medium with a constant density still has a similar solution as shown in equation (2.22). Such an solution leads to a wavefield extrapolation formula given as the LHA (or GPSPI) (Margrave and Ferguson, 1999; Margrave et al., 2006), which reads

$$\psi(x, \Delta z, \omega) = \frac{1}{2\pi} \int_{\mathbb{R}} \hat{\Psi}(k_x, 0, \omega) \hat{W}(x, k_x, \Delta z, \omega) \exp(-ik_x x) \mathrm{d}k_x, \tag{2.30}$$

where x denotes the horizontal coordinate, k_x is the wavenumber coordinate, ω is the temporal frequency, and Δz is the step size of extrapolation in z (vertical) direction. $\hat{\Psi}(k_x, 0, \omega)$ is the temporal-spatial Fourier transform of $\psi(x, 0, t)$ (seismic data recorded on the surface) defined similarly in equation (2.12) but using a locally homogeneous velocity at x. Since velocity is x dependent, the total wavenumber (scalar) is also related to the locally homogeneous velocity $v(x)$, which is written as

$$k(x, \omega) = \frac{\omega}{v(x)}. \tag{2.31}$$

Usually, $v(x)$ is referred to as an instantaneous velocity at x, and it is constant in the slab at x given that the thickness (Δz) of the slab is small enough. If velocity also varies in the z direction at x, we may either set Δz smaller to make the local velocity approach a constant velocity or use an average velocity from 0 to Δz.

In equation (2.30), the Fourier integral is only an inverse Fourier transform when $v(x)$ is constant. Otherwise, equation (2.30) is a standard-form Kohn-Nirenberg pseudodifferential operator (Kohn and Nirenberg, 1965) and it is common to call the function which multiplies the field ($\hat{\Psi}$), the pseudodifferential operator *symbol*. In the special case of constant velocity the symbol becomes an ordinary Fourier multiplier, or filter transfer function, and corresponds to a stationary convolution filter in the x domain. $\hat{W}(x, k_x, \Delta z, \omega)$ is the wavefield extrapolation *symbol*, which is an key component in the LHA wavefield extrapolation, and it is defined as a spatial phase-shift operator.

$$\hat{W}(x, k_x, \Delta z, \omega) = \exp[ik_z(x, k_x, \omega)\Delta z]. \tag{2.32}$$

$$k_z(x, k_x, \omega) = \begin{cases} \sqrt{k^2(x,\omega) - k_x^2}, & k^2(x,\omega) \geqslant k_x^2 \\ i\sqrt{k_x^2 - k^2(x,\omega)}, & k^2(x,\omega) < k_x^2. \end{cases} \tag{2.33}$$

The first formula in (2.33) describes wave propagation in the wave-like region, and the second one is related to evanescent components in wavefields. Wavefield extrapolation with equation (2.30) is very accurate; however, it is generally not numerically efficient unless the velocity is constant, in which case the integral can be accomplished as a fast Fourier transform. Direct implementation for general $v(x)$ is a general matrix-vector multiplication and is usually too slow for large industrial data sets. Thus the next section develops an approximation with controlled accuracy and cost.

A first step is to express the LHA extrapolation symbol as the superposition of a set of

constant velocity symbols (i.e. ordinary Fourier multipliers) in the form

$$\hat{W}\left(x, k_x, \Delta z, \omega\right) \approx \sum_{j \in \mathbb{Z}} \Omega_j\left(x\right) S_j\left(x\right) \hat{W}_j\left(k_x, \Delta z, \omega\right). \tag{2.34}$$

A set of windows, $\Omega_j,\ (j \in \mathbb{Z})$, form a partition of unity, which means

$$\sum_{j \in \mathbb{Z}} \Omega_j\left(x\right) = 1, \tag{2.35}$$

where $\mathbb{Z}$ is the integer set. Reference (average) velocities in these partitions are defined as

$$v_j = \frac{\int_{\mathbb{R}} \Omega_j\left(x\right) v\left(x\right) \mathrm{d}x}{\int_{\mathbb{R}} \Omega_j\left(x\right) \mathrm{d}x},\ \ j \in \mathbb{Z}. \tag{2.36}$$

The total wavenumber corresponding to the reference velocity v_j is

$$k_j = \frac{\omega}{v_j}. \tag{2.37}$$

Locally constant wavefield extrapolator $\hat{W}_j\left(k_x, \Delta z, \omega\right)$, defined in window $\Omega_j\left(x\right)$, can be written in an explicit form, which is

$$\hat{W}_j(k_x, \Delta z, \omega) = \exp[i k_z(k_j, k_x, \omega)\Delta z], \tag{2.38}$$

and

$$k_z\left(k_j, k_x, \omega\right) = \begin{cases} \sqrt{k_j^2 - k_x^2}, & k_j^2 \geqslant k_x^2 \\ i\sqrt{k_x^2 - k_j^2}, & k_j^2 < k_x^2. \end{cases} \tag{2.39}$$

The split-step Fourier operator (Stoffa et al., 1990) in equation (2.34) is defined as

$$S_j(x) = \exp\left[i\omega\Delta z\left(\frac{1}{v\left(x\right)} - \frac{1}{v_j}\right)\right]. \tag{2.40}$$

The major utility of the approximation of equation (2.34) is that the spatial variation and wavenumber variation appear separated into independent multiplicative terms. The justification of this approximation is essentially similar to that of the split-step Fourier

process itself. Equation (2.34) is simply a separate split-step Fourier process in each Ω_j window. This has very attractive properties when coupled with our method for developing the POU. As will be seen in Chapter 3, Ω_j windows can be quite complicated functions with multiple local maxima and complex overlaps.

Using equation (2.34) in equation (2.30) yields the approximation of LHA wavefield extrapolation as

$$\psi_{\mathrm{P}}(x, \Delta z, \omega) = \sum_{j \in \mathbb{Z}} \Omega_j(x) S_j(x) \int_{\mathbb{R}} \hat{W}_j(k_x, \Delta z, \omega) \hat{\Psi}(k_x, 0, \omega) \exp(-ik_x x) \mathrm{d}k_x, \quad (2.41)$$

where $\Omega_j\,(x)$ and $S_j\,(x)$ are k_x independent and taken out of the integrand. Noting that the integral in equation (2.41) is now an inverse spatial Fourier transform. Using the (spatial) Fourier transform abstract operators (see section 2.3 in this chapter) gives

$$\psi_{\mathrm{P}}(x, \Delta z, \omega) = \sum_{j \in \mathbb{Z}} \Omega_j(x) S_j(x) \mathscr{F}^{-1}[\hat{W}_j(k_x, \Delta z, \omega) \mathscr{F} \Psi(x, 0, \omega)]. \quad (2.42)$$

Equation (2.41) or (2.42) is called the Gabor wavefield extrapolation in this thesis. $\hat{W}_j$ in these formulas is called the Gabor wavefield extrapolator.

If the window choice is $\Omega_j(x) = \delta(x)$ (i.e., the Dirac Delta function), then there will be one window (a spike) at each output position x, which makes the number of windows infinite. Summation over an infinite number of windows (spikes) approaches an integral in the real line $\mathbb{R}$. In such a case, split-step Fourier operators at x become 1 everywhere because $v_j = v(x)$ (see equation (2.40)). As a result, we obtain equation (2.30) from equation (2.42), which also proves that the LHA wavefield extrapolation is the limit of the Gabor method described in equation (2.42).

2.4.1 Gabor extrapolation schemes simulating the PSPI and the NSPS

Equation (2.42) can be considered as an operator in the spatial Gabor domain. In doing so, I will derive a general class of similar expressions for which equation (2.42) is a particular case.

Starting with the partition of unity (see equation (2.35)) and imposing the additional requirement that the $\Omega_j(x)$ be non-negative, $\Omega_j(x)$ can be split as (Grossman et al., 2002a)

$$g_j(x) = \Omega_j^p(x),\ \gamma_j(x) = \Omega_j^{1-p}(x),\ p \in [0,1], \tag{2.43}$$

where $g_j(x)$ and $\gamma_j(x)$ are the analysis and synthesis windows, respectively. The forward Gabor transform is written as

$$V_g\psi_j(x) = F\left[g_j(x)\,\psi(x)\right],\ j \in \mathbb{Z}, \tag{2.44}$$

where $V_g\psi_j(x)$ is called the Gabor spectrum of $\psi(x)$. $V_g\psi_j(x)$ is actually a set of windowed Fourier transforms of the input signal $\psi(x)$ localized by the analysis window $g_j(x)$. $\psi(x)$ can be recovered by the inverse Gabor transform

$$\psi(x) = V_\gamma^{-1}\left[V_g\psi_j(x)\right] = \sum_{j\in\mathbb{Z}} \gamma_j(x)\,\mathscr{F}^{-1}\left[V_g\psi_j(x)\right] = \sum_{j\in\mathbb{Z}} \gamma_j(x)\mathscr{F}^{-1}\mathscr{F}\left[g_j(x)\,\psi(x)\right], \tag{2.45}$$

where V_γ^{-1} stands for the inverse Gabor transform, which consists of the inverse Fourier transform of the Gabor spectrum, multiplication by the synthesis window, and a summation over the window index j. With $\mathscr{F}^{-1}\mathscr{F} = 1$, proof of equation (2.45) is immediate using equations (2.35) and (2.43) in equation (2.45).

Since $p \in [0,1]$, there are infinitely many ways to construct the Gabor transform pair. If $p = 0$ in equation (2.43), then

$$g_j(x) = 1,\ \gamma_j(x) = \Omega_j(x), \tag{2.46}$$

which concentrates the localization after the inverse Fourier transform. When $p = 1$, the opposite happens with all of the localization happening before the forward Fourier transform. Intermediate values of p give intermediate options.

Using equation (2.46) in equation (2.45) gives

$$\psi(x) = V_\gamma^{-1}\left[V_g\psi_j(x)\right] = \sum_{j\in\mathbb{Z}} \Omega_j(x)\,\mathscr{F}^{-1}\mathscr{F}\left[\psi(x)\right]. \tag{2.47}$$

Equation (2.47) does nothing but transform $\psi(x)$ to the Gabor domain and back. However, if $\hat{W}_j(k_x, \Delta z, \omega)$ and $S_j(x)$ are inserted into equation (2.47) in front of $\mathscr{F}$ and $\mathscr{F}^{-1}$, respectively, equation (2.47) becomes equation (2.42). So equation (2.42) uses the particular limiting case $p = 0$, where the forward Gabor transform degenerates to the Fourier transform and all of the windowing happens in the inverse transform.

To be clear about this operation, I give a brief description on implement of equation (2.42). First of all, wavefield extrapolations using reference velocities related to partitions $\Omega_j\,(j \in \mathbb{Z})$ are performed, but no windowing happens before extrapolations. In the inverse transform, wavefields are windowed by $\Omega_j\,(j \in \mathbb{Z})$, and summed over window indices j in the end. The windowing process is equivalent to the interpolation between wavefields related to various reference velocities. This results in an operation similar to the familiar PSPI process of Gazdag and Sguazzero (1984).

Using a similar derivation and the choice of $p = 1$ gives

$$\begin{aligned}\psi_{\mathrm{N}}(x, \Delta z, \omega) \;=\; & \int_{\mathbb{R}} \sum_{j\in\mathbb{Z}} S_j(x)\hat{W}_j(k_x, \Delta z, \omega) \int_{\mathbb{R}} \Omega_j(x)\Psi(x, 0, \omega)\exp\left(ik_x x\right)\mathrm{d}x \\ & \cdot \exp\left(-ik_x x\right)\mathrm{d}k_x,\end{aligned} \tag{2.48}$$

which is a nonstationary phase shift (NSPS) operator that is closely related to that of

Margrave and Ferguson (1999). Written in the abstract Fourier symbols, it is

$$\psi_{\mathrm{N}}(x,\Delta z,\omega)=\sum_{j\in\mathbb{Z}}S_j(x)\mathscr{F}^{-1}[\hat{W}_j(k_x,\Delta z,\omega)\mathscr{F}\Omega_j(x)\Psi(x,0,\omega)]. \tag{2.49}$$

In this process, the wavefield is windowed into many small pieces, Fourier transformed and then extrapolated. The phase-shifted pieces of the wavefield are summed and finally an inverse Fourier transform is performed to get the extrapolated version of the wavefield. There is no windowing in the inverse transform.

A combination of approximate PSPI and NSPS (both in half step size) within a single extrapolation step Δz can be written as

$$\begin{aligned}\psi_{\mathrm{PN}}(x,\Delta z,\omega) &= \sum_{j\in\mathbb{Z}}\Omega_j(x)S_j(x)\int_{\mathbb{R}}\Bigg[\hat{W}_j(k_x,\frac{\Delta z}{2},\omega)\sum_{m\in\mathbb{Z}}S'_m(x)\hat{W}_m(k_x,\frac{\Delta z}{2},\omega)\\ &\quad\cdot\int_{\mathbb{R}}\Omega'_m(x)\Psi(x,0,\omega)\exp(ik_xx)\mathrm{d}x\Bigg]\exp(-ik_xx)\mathrm{d}k_x,\end{aligned} \tag{2.50}$$

where a slab with a thickness of Δz is split into halves in the middle. An NSPS is carried out in the first half step $\Delta z/2$ and a PSPI follows in the second half step. In the NSPS process, the wavefield is windowed before extrapolation. $\Omega'_m(x)$ satisfies POU. The output from the NSPS process (still in the frequency-wavenumber domain) is the input for the PSPI process in the second half step. $\Omega_j(x)$ suffices POU, too. However, these two sets of POU may not be exactly the same if the velocity has variations in z direction inside the slab. Otherwise, these POU's are the same; so are the split-step Fourier corrections, $S_j(x)$ and $S'_j(x)$.

Whether this combination scheme is more efficient and accurate than Gabor approximations with single PSPI or NSPS method remains as a open research topic. In terms of

abstract symbols, we have equation (2.50) as

$$\begin{aligned}\Psi_{\mathrm{PN}}(x,\Delta z,\omega) &= \sum_{j\in\mathbb{Z}}\Omega_j(x)S_j(x)\mathscr{F}^{-1}\Bigg[\hat{W}_j\left(k_x,\frac{\Delta z}{2},\omega\right)\\ &\quad\cdot\sum_{m\in\mathbb{Z}}S'_m(x)\hat{W}_m\left(k_x,\frac{\Delta z}{2},\omega\right)\mathscr{F}\Omega'_m(x)\Psi(x,0,\omega)\Bigg]. \quad (2.51)\end{aligned}$$

While the algorithm we implement here can be viewed as a type of screen operator, the fresh perspective of a Gabor operator suggests new operator varieties, corresponding to different choices of p, that will be the subject of future research.

2.4.2 Gabor extrapolation schemes using intermediate p values

The Gabor depth imaging method is an effective extension of the phase-shift method to heterogeneous environments. The formulation as a Gabor algorithm allows further extensions and variations of the method. As a general form of the Gabor wavefield extrapolation, it reads

$$\begin{aligned}\psi_{\mathrm{G}}(x,\Delta z,\omega) &= \sum_{j\in\mathbb{Z}}S_j(x)\Omega_j^{1-p}(x)\int_{\mathbb{R}}\hat{W}_j(k_j,k_x,\Delta z)\int_{\mathbb{R}}\Omega_j^{p}(x)\\ &\quad\cdot\psi(x,0,\omega)\exp\left(ik_xx\right)\mathrm{d}x\exp\left(-ik_xx\right)\mathrm{d}k_x, \quad (2.52)\end{aligned}$$

which is also written in a form of abstract symbols as

$$\psi_{\mathrm{G}}(x,\Delta z,\omega)=\sum_{j\in\mathbb{Z}}S_j(x)\Omega_j^{1-p}(x)\mathscr{F}^{-1}[\hat{W}_j(k_j,k_x,\Delta z)\mathscr{F}\Omega_j^{p}(x)\psi(x,0,\omega)]. \quad (2.53)$$

The choice of $p=0$ in equation (2.52) is only one extreme of the continuum of possibilities for this key parameter. In this case, the Gabor algorithm proceeds with a conventional forward Fourier transform, a laterally variable phase shift, and an inverse Gabor transform. If a similar development is followed for the case $p=1$, the result is an algorithm that sandwiches the phase shift between a forward Gabor transform and

an inverse Fourier transform. When p is some other value between 0 and 1, then both transforms become Gabor. It is not clear at this stage what are the relative merits of these alternative algorithms, but they all have different localization properties and this may prove useful. What is not difficult to show is that the $p = 0$ and $p = 1$ cases correspond closely to standard and anti-standard pseudodifferential operators while the other p values give intermediate forms. Further investigation is more a problem in mathematics than geophysics and will be the subject of future research. In Chapter 4, I will show some imaging examples from selection of some of these intermediate parameters.

2.5 Implementation of the Gabor wavefield extrapolation

2.5.1 Gabor wavefield extrapolation

Gabor depth imaging theories have been developed in the previous sections. In this section, I will describe how to implement the Gabor wavefield extrapolation and impulse responses are shown as examples.

I use a shot record as an example. The Gabor wavefield extrapolation scheme is selected as the PSPI (see equation (2.41)) approximation to the LHA (see equation (2.30)). In the Gabor wavefield extrapolation, a shot record is usually referred to as seismic data in the time-space (x, t) domain, denoted by $\psi(x, 0, t)$. Each shot record contains a set of seismic traces, corresponding to various spatial positions of geophones or receivers. Each of these traces stores signal samples in a time series. We can see from equation (2.41) that the input signal is actually written as $\hat{\Psi}(k_x, 0, \omega)$, expressed in the frequency-wavenumber (f, k) domain, which means the shot record has had a temporal and then a spatial Fourier transform applied before the wavefield extrapolation is performed.

In the (f, k) domain, a phase shift has been performed by the Gabor wavefield extrapolator $\hat{W}_j(k_x, \Delta z, \omega)$; the final step of the extrapolation is to use an inverse Fourier transform and bring the wavefield back to the spatial-frequency (ω, x) domain. The process is performed within each spatial window $\Omega_j(x)(j \in \mathbb{Z})$; or in other words, the extrapolated wavefields are localized by those spatial windows and then Fourier transformed back to the spatial domain. A residual phase-shift correction is also done by the split-step Fourier operator $S_j(x)(j \in \mathbb{Z})$ inside each spatial window. Summation over the indices of spatial windows corresponding to pieces of the input wavefield with phase shift applied gives the extrapolated wavefield at a new depth Δz from depth 0. These are procedures of the Gabor prestack depth migration on a shot record. Figure 2.4 shows the flow chart regarding the process, which is presented by equation (2.41).

Similarly, the flowcharts for the NSPS and the General Gabor extrapolations are shown in Figure 2.5 and Figure 2.6 (see also equations (2.48) and (2.52), respectively).

To show how well the Gabor wavefield extrapolation works, I will give examples of impulse responses in the next section.

2.5.2 Impulse responses with the Gabor wavefield extrapolator

Impulse response in a homogeneous half space

The first step in systematic imaging tests of this research is to show the impulse response given by the Gabor wavefield extrapolator in a homogeneous medium. The homogeneous model with a constant velocity of 2000 m/s is used as a model without lateral velocity variations, the simplest case. Figure 2.7 (a) shows a set of impulses at various positions, set as seismic traces with wavelets embedded. Figure 2.7 (b) shows those impulse responses when extrapolated to depth 200 m downwards from their original positions. Figure 2.7

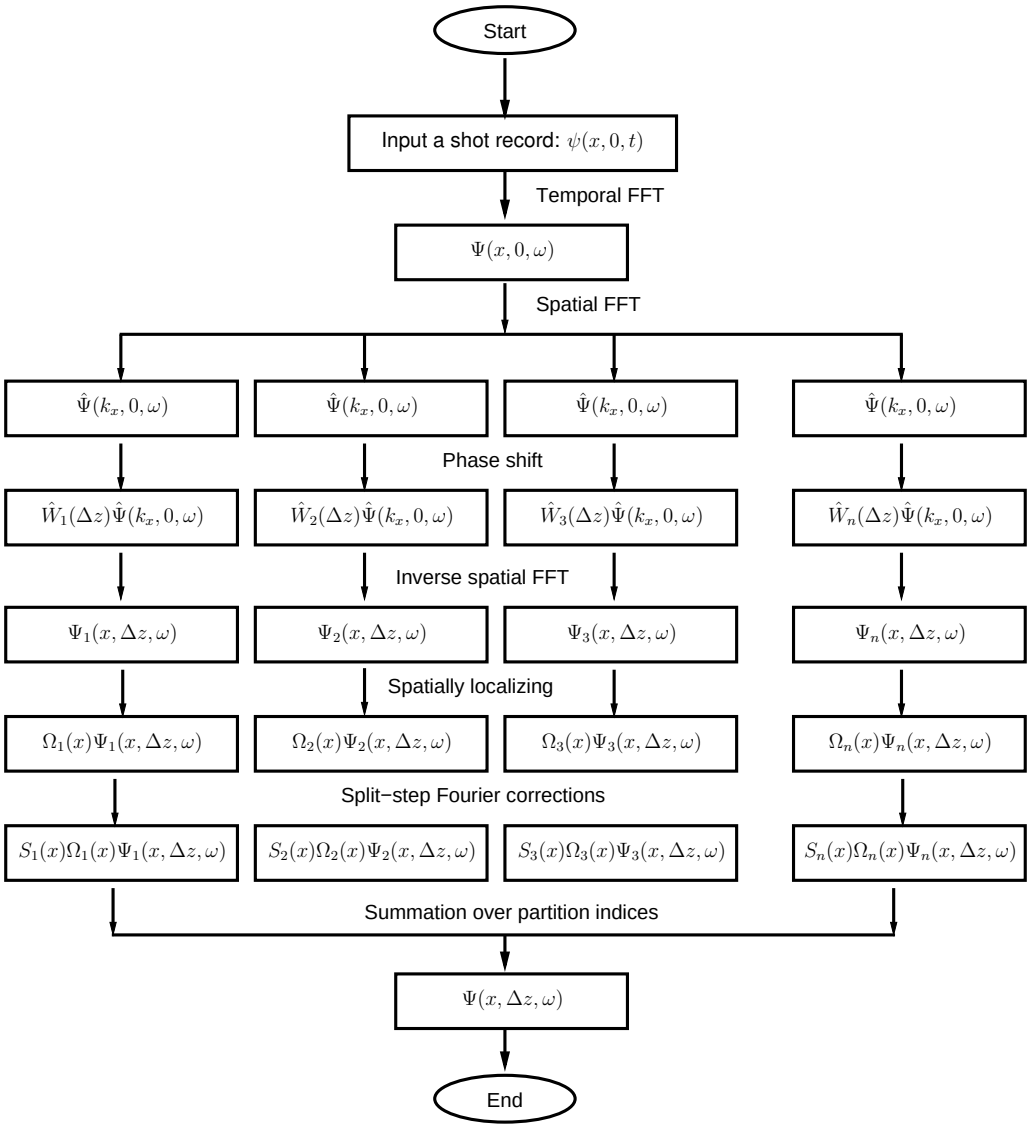


Figure 2.4: A flowchart showing the Gabor wavefield extrapolation (simulating the PSPI) on a shot record.

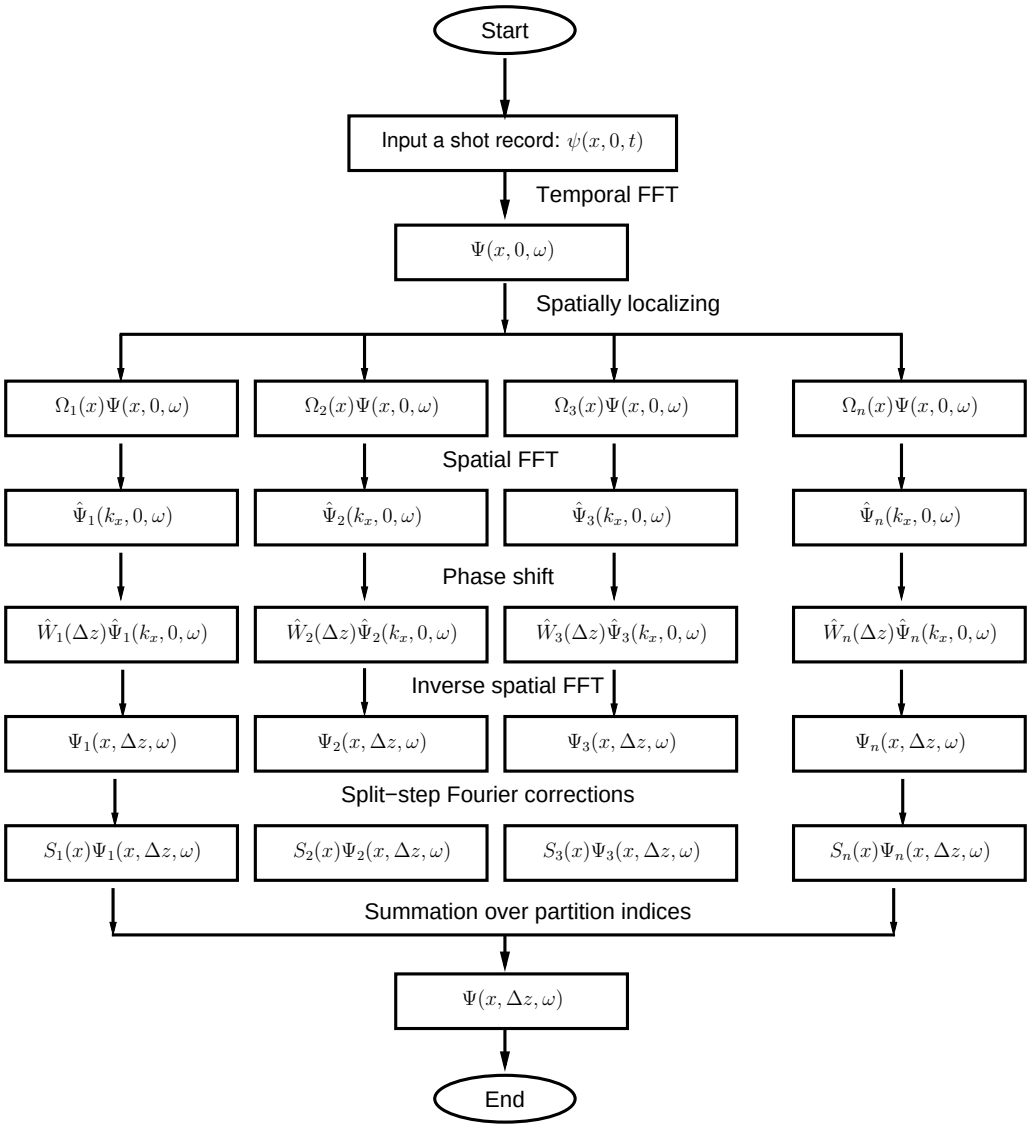


Figure 2.5: A flowchart showing the Gabor wavefield extrapolation (simulating the NSPS) on a shot record.

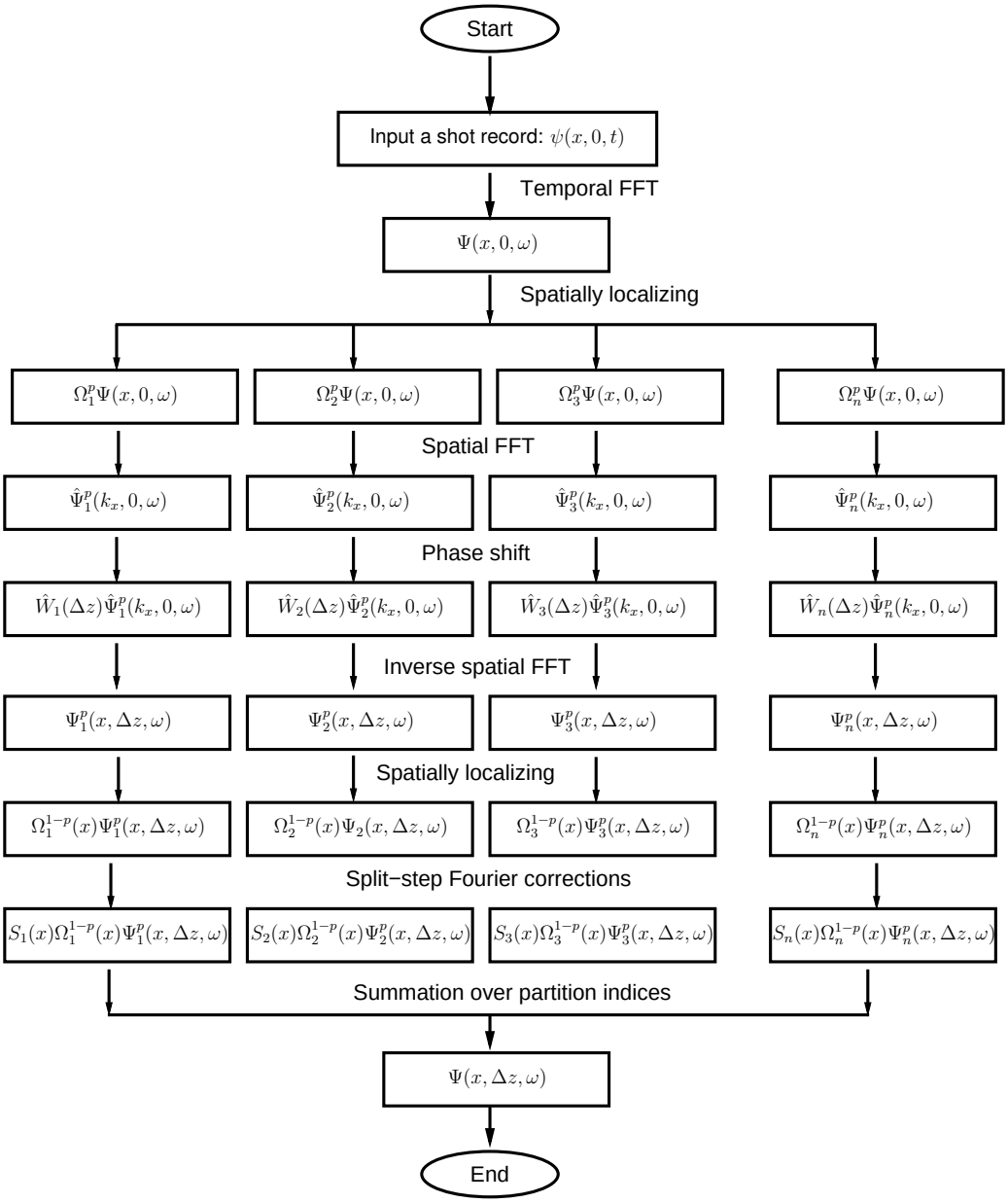


Figure 2.6: A flowchart showing the Gabor wavefield extrapolation (using the intermediate p) on a shot record.

(c) shows the response from those impulses when extrapolated upwards 100 m from depth 200 m. Figure 2.7 (d) shows the impulse responses when they are extrapolated back to their original positions. We can see that they match the original positions exactly, which is required of the theory since Gabor is equal to Gazdag (1978) in this case and the latter is exact.

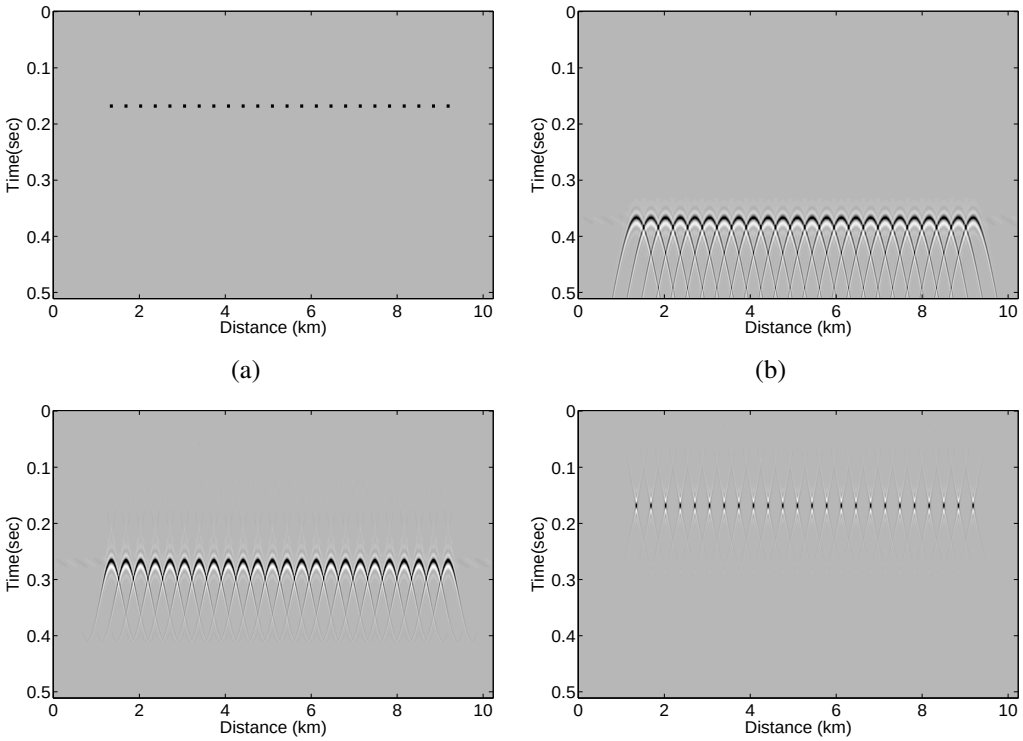


Figure 2.7: Impulse responses in a homogeneous half space with a velocity of 2000 m/s. (a) Impulses. (b) Impulse responses when extrapolated 200 m downward from their original positions. (c) Impulse responses when extrapolated 100 m upwards from 200 m. (d) Impulses extrapolated back to their original positions.

It can be noticed that though the impulses are extrapolated back to their original positions, but their shapes are not as exact as the original impulses. This is caused by the

evanescent filtering during the wavefield extrapolation. Since the impulses lose their (spatial) frequency contents, their shapes have been altered after they are extrapolated.

Impulse response in an inhomogeneous half space

Another test for the impulse response is done in an inhomogeneous medium. In the second model, I use a velocity model with two distinct velocity pieces neighbouring each other right in the middle. To the left side of the border, the velocity is 2000 m/s, and to the right the velocity is 4000 m/s. I use this model to represent a velocity model with lateral variations. We expect the impulse responses will be in two groups due to different propagation velocities. Figure 2.8 (a) gives extrapolated version of those impulses shown in Figure 2.7 (a), 100 m downward from their original positions. We can see in Figure 2.8 (a) that those impulses propagate at different velocities throughout the model and separate themselves into two groups. Figure 2.8 (b) gives a similar conclusion about the difference, when extrapolation is brought down to 200 m from the original position. Figure 2.8 (c) shows the response of the impulses when extrapolated 100 m upward from the bottom (200 m). The Gabor extrapolator propagates the impulses back to their original positions, just as it did to the impulses in the homogeneous medium (see Figure 2.8 (d)).

In Figure 2.8 (d), we can see that the 12^{th} impulse is distorted. A careful inspection tells us that this impulse is right on the edge between the high and low velocities, which gives an edge effect to this one and makes a distorted impulse after it is extrapolated. Another one (the 13^{th}) close to the edge is also slightly distorted. These could be cases in seismic data, which causes artifacts during migrations.

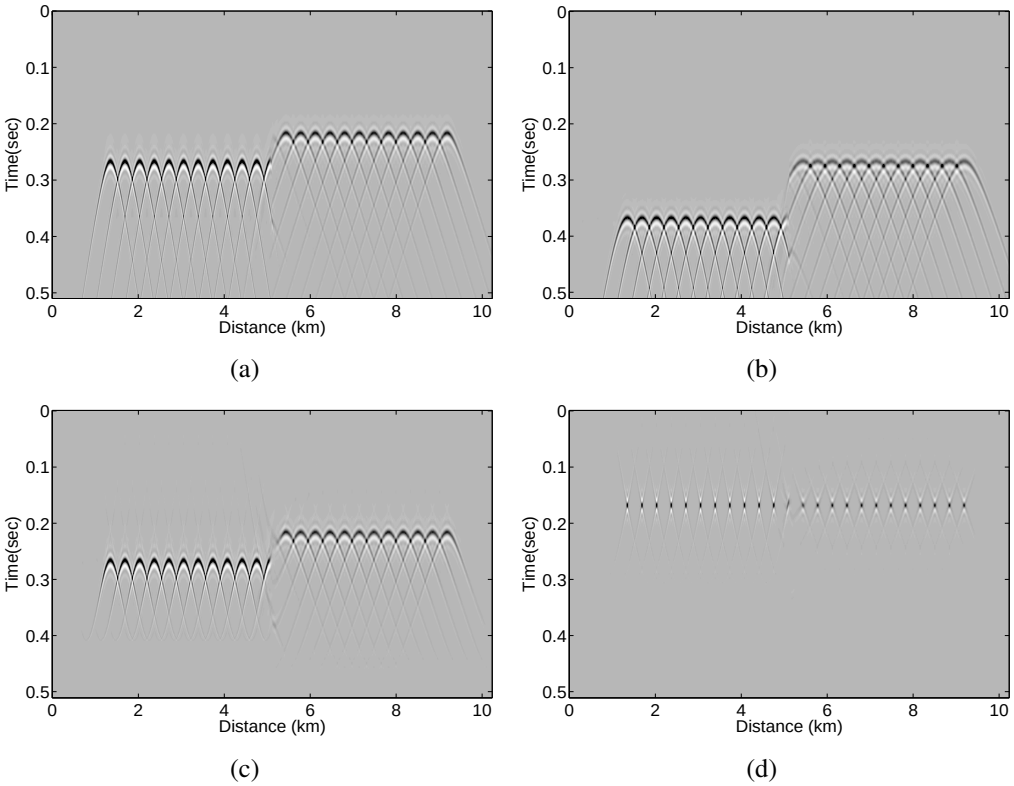


Figure 2.8: Impulse responses in an inhomogeneous half space. (a) Impulses extrapolated down to depth of 100 m. (b) Impulses extrapolated down to depth of 200 m. (c) Impulses extrapolated 100 m upward from depth 200 m. (d) Impulses extrapolated back to their original positions.

2.6 Chapter summary

In this chapter, I started with an introduction of explicit wavefield extrapolation and phase-shift migration methods. Then starting from a 2D scalar wave equation, the wavefield extrapolation formulas in various media, from homogeneous to heterogeneous, were derived. As the theoretical limit of these wavefield extrapolations, the GPSPI extrapolation was introduced.

Gabor wavefield extrapolation theory was introduced and developed as one of those approximations of the GPSPI using the Gabor transform. The approximation was specified by the choice of the p value, which gives infinitely many ways of using the Gabor transform. Two extreme cases ($p = 0$, $p = 1$) are known as the simulation of the PSPI and NSPS phase-shift methods. These two schemes may be used in wavefield extrapolation in one depth step, each of which extrapolates half step size.

Impulse response tests both in homogeneous and inhomogeneous half spaces have shown that the Gabor wavefield extrapolator, in general, works properly as required in theory and ready for the depth imaging with seismic data sets except for some extreme cases (see the 12^{th} impulse response). However, this method is still not efficient enough for depth imaging even of 2D data sets because the Gabor extrapolator uses narrow windows, each of which can accommodate the most rapidly varying velocity in lateral direction. However, this may not be necessary where velocity variations are not very rapid. In the next chapter, I will present the techniques that will help to develop a faster Gabor depth imaging.

Chapter 3: Fast Gabor Depth Migration

This chapter is expanded from a published paper in *Geophysics* by Ma and Margrave (2008). In the previous chapter, I described the depth imaging theory using the Gabor transform. Impulse response tests have shown that the Gabor wavefield extrapolator is able to extrapolate wavefields in a correct way. However, so far the Gabor imaging method has been implemented in a way that uses evenly distributed small windows across the lateral coordinate. These small windows (called atomic windows) are usually chosen to address the most rapidly varying velocity in the lateral direction at a depth step, which is obviously not necessary everywhere across the lateral coordinate. In places with slowly varying velocity, these narrow windows can be consolidated into fewer, wider ones, while maintaining acceptable accuracy in the depth imaging process. Since each of these windows (also called partitions) requires a Fourier transform, they are the main contributors to the computation cost in the Gabor wavefield extrapolation. It is attractive to consolidate those small partitions into fewer, large ones to improve the Gabor imaging speed.

Next, I will analyze the computational cost in the Gabor wavefield extrapolation. Based on the analysis, methods for efficient Gabor wavefield extrapolation will be suggested.

3.1 Computational cost of the Gabor depth migration method

Implementation of the Gabor wavefield extrapolation (see section 2.5 in Chapter 2) indicates that computational cost is proportional to the number of spatial Fourier transforms. In each partition, the Gabor method needs a forward and an inverse spatial Fourier trans-

form to extrapolate the wavefield to a new depth; and the process is particular to a (temporal) frequency. To generate the image, both the source field and the recorded data are extrapolated, which makes the number of Fourier transforms double at each depth step and frequency. i.e., four spatial Fourier transforms are required to extrapolate the source field and seismic data through a partition and image at each depth and for each frequency.

The number of depth steps in a typical depth-marching migration can be as few as several hundred or more than a thousand. The number of frequencies can easily be more than a thousand. The product of the number of Fourier transforms related to a single partition and those of depths, frequencies and partitions at depths gives a very large number of Fourier transforms during the extrapolation process. Any changes in the numbers of depths, frequencies and partitions to smaller ones will reduce Fourier transforms in a linear way, resulting in a much more efficient Gabor wavefield extrapolation.

Usually, in seismic depth migrations, numbers of depths and frequencies are fixed. However, the number of partitions can be minimized by consolidating some of them given an accuracy criterion (see section 3.3 in this chapter). It would also be very effective to improve imaging efficiency by altering spatial sample rates at low frequencies for wavefields, resulting in smaller sized Fourier transforms, which can be processed much faster (see section 3.4 in this chapter).

Reducing partitions, hence the number of Fourier transforms, comes as the first method to improve the Gabor depth imaging speed. As will be shown in the following sections, in many cases we can eliminate many redundant (small) partitions by consolidating them as a large (wide) one using an adaptive partitioning algorithm. Adaptive partitioning means that partitions created by a partitioning algorithm are not evenly distributed atomic windows but partitions with complicated shapes adapted to the lateral velocity variation. The

partition of unity (POU) criterion still holds for these varying partitions (examples will be shown in the following sections). Given an accuracy criterion, an adaptive partitioning algorithm can make a very efficient Gabor depth imaging method but it sacrifices imaging accuracy in some degree. Three adaptive algorithms will be introduced in section 3.3. The Gabor wavefield extrapolation with an adaptive partitioning algorithm is named 'adaptive Gabor wavefield extrapolation' in this thesis.

Altering sizes of Fourier transforms is the second method, called spatial resampling, which promotes efficient Gabor extrapolation while keeping imaging accuracy. Details of this method will be described in section 3.4.

In summary: two methods are suggested for efficient Gabor extrapolation based on computational cost analysis. The first one optimizes the number of partitions (related to Fourier transforms) at a depth step according to the lateral velocity variation; the second one uses the spatial resampling technique to minimize the number of samples in Fourier transforms when working with wavefields at low frequencies.

3.2 Adaptive Gabor wavefield extrapolation

The simplest implementation of equations (2.41), (2.48) or (2.52) would use a set of identical constant-width partitions $\Omega\left(x - x_j\right)$ $\left(j \in \mathbb{Z}\right)$, any one of which we call the atomic window, uniformly distributed along the lateral coordinate. However, this is generally inefficient as the atomic window width must be chosen to accommodate the most rapid velocity change and this may be completely unnecessary elsewhere. For efficiency, the number of partitions, hence the reference velocities, should be reduced to as few as possible while maintaining acceptable accuracy. This can be done by consolidating adjacent

atomic windows to form ‘molecules’ in regions where there is little velocity variation.

Application of adaptive partitioning algorithms is one of the most novel ideas in the Gabor imaging method. That is, we use an optimal set of spatial partitions that localize the wavefield more strongly (i.e. narrow partitions) as the velocity variation increases, while minimizing an estimate of phase errors from the wavefield extrapolator at each extrapolation step. Conversely, when the lateral velocity variation is weak, very little localization is required and much wider partitions can be used. Since computation costs are directly proportional to the number of partitions, this then controls cost; however, with some sacrifice of accuracy. These ideas will be demonstrated with imaging examples in the next chapter.

To implement an adaptive partitioning algorithm, a criterion should be set up to guide the algorithm to select partitions according to the lateral velocity variation. In the following sections, three partitioning algorithms will be described and demonstrated using various partitioning criteria.

3.3 Adaptive partitioning algorithms

The most intuitive way of implementing adaptive partitioning algorithm is to look at velocity gradients in the lateral direction such that we can see how rapidly the lateral velocities at a given depth are changing; based on these variations, we may set up an algorithm to create partitions adapted to these velocity variations. Another way is more physical, meaning the measurement of phase difference between the Gabor extrapolator and the LHA extrapolator, which will be called ‘phase error’ in the following sections. The advantage of the second partitioning method is that partitions are determined by the phase error, which is closely related to the physics behind the wavefield extrapolation.

From the next section, I will start with introduction to an adaptive partitioning algorithm using the lateral velocity gradient as criterion given by Grossman et al. (2002b). Then I will develop two other partitioning algorithms using criteria defined directly and indirectly in terms of phase errors.

3.3.1 Adaptive partitioning with velocity gradients

The velocity-gradient adaptive partitioning (VGAP) algorithm is the first partitioning algorithm implemented in the adaptive Gabor wavefield extrapolation (Grossman et al., 2002b; Ma and Margrave, 2005, 2006c). In this method, lateral velocity gradients determine localizing partitions in the Gabor wavefield extrapolation. For mathematical details of the algorithm, see Grossman et al. (2002b). I will describe the algorithm in an interpretational way. Next, some examples given by the partitioning algorithm will be shown.

Since each atomic window (atom) spans over a few output positions in the lateral coordinate, a reference velocity can be defined in an atom by averaging velocities at output positions involved in the atom. Starting with atoms evenly distributed in the lateral coordinate, the algorithm tries to eliminate redundant atoms by consolidating them given a partitioning criterion. The criterion can be defined as the product of a partitioning parameter set in percentage (e.g., 10%) and a reference velocity related to an atom or a molecule.

The algorithm begins with the first atom, called the current atom, from one side of a velocity profile and looks into the atom adjacent to it; the difference between reference velocities of the two is calculated. When the difference does not exceed the partitioning criterion, the current atom is summed to the second one, resulting in a wider partition, a molecule. Otherwise, the current atom will be left untouched and the second atom becomes the current atom.

The partitioning process is carried on with the current molecule or atom. If the partitioning criterion is satisfied, more atoms will be absorbed into current molecule (atom). The partitioning proceeds toward the other side of the velocity profile and stops at the far boundary. Usually, most of the atoms are consolidated into molecules, adapted to the lateral velocity variation given a partitioning criterion. Partitions created by the VGAP

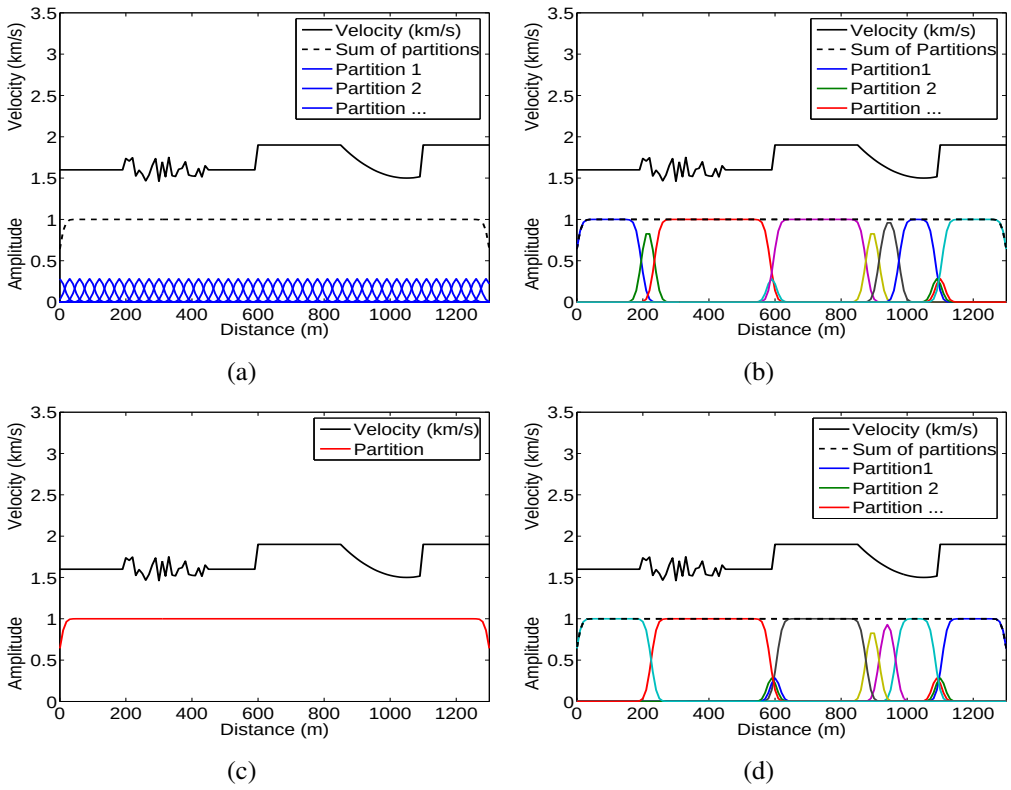


Figure 3.1: Partitions for the Gabor wavefield extrapolation. (a) Atomic windows. Partitions given by the VGAP algorithm using a partitioning parameter of (b) 5% (from left to right), (c) 50% (from left to right) and (d) 5% (from right to left).

algorithm with various partitioning criteria are shown in Figure 3.1. The velocity profile is designed after Grossman et al. (2002b). Figure 3.1 (a) shows a set of atomic windows used in the Gabor extrapolation by default no matter what kind of velocity profiles it deals with,

even constant ones. Without adaptive partitioning, Gabor extrapolation uses 131 windows (partitions) in this case, shown in blue lines at the bottom. Every third window has been plotted to make them easy to see.

Figure 3.1 (b) shows the result from the adaptive partitioning using a partitioning parameter of 5%. As predicted from the analysis, the adaptive algorithm consolidates most atoms, giving wide partitions (molecules) instead of narrow ones (atoms). The number of partitions drops to 11 from 131. We can also see that a few atoms (e.g., those atoms with their centres close to 600 m and 1100 m) still remain, adapted to the most rapid velocity variation, while this is not necessary elsewhere. A Gabor wavefield extrapolation employing the (adaptive) partitions created in this case would be predicted (only from the number of partitions) to run approximately 12 (131/11) times faster compared to the one using the atomic windows. Considering other aspects of efficiency associated with the reduction of partitions, the adaptive Gabor depth imaging could be even faster than this. I will show this by comparing run times to produce imaging results in Chapter 4.

Figure 3.1 (c) shows adaptive partitioning from the same velocity model, but with a partitioning parameter of 50%. We can see that this time the algorithm gives only one partition across the whole lateral coordinate, meaning all the atoms are consolidated into one partition. In this case, the adaptive Gabor wavefield extrapolation is the fastest, but it may not produce an imaging result with acceptable accuracy using only one reference velocity in the extrapolation with such a complex velocity model.

Figure 3.1 (d) shows the adaptive partitioning result also using a partitioning parameter of 5% as the one used in (b), but the partitioning starts from the right-hand side of the same velocity model as shown in Figure 3.1 (b). Compared to the partitioning result shown in (b), we can see the two sets of partitions are different, which indicates that the partitioning

algorithm is direction dependent.

The VGAP algorithm is a good partitioning algorithm, which allows us to control computational cost and accuracy. If a smaller partitioning parameter is used, the algorithm gives a large number of narrower partitions for the Gabor extrapolation, hence more accurate and slower wavefield extrapolation. Otherwise, we can use large partitioning parameters to get faster Gabor extrapolations but less accurate imaging results.

We can see that the partition of unity (POU) property is also satisfied in all examples shown in Figure 3.1 (see black-dashed lines on top of partitions in (a), (b) and (d) and the solid red line in (c)).

Although the VGAP algorithm gives a good partitioning method, we have difficulty when getting to 2D adaptive partitioning needed in the 3D Gabor extrapolation. First of all, calculating velocity gradient (or difference) in the 2D lateral dimensions is cumbersome. Another problem comes from the choice of partitioning parameters (i.e. the allowable velocity difference). Since it is not directly related to any physics behind the wavefield extrapolation, it is not immediately obvious how to set this parameter. Directional dependence also puts the algorithm in a dilemma when deciding on which side of the velocity profile to start the adaptive partitioning process.

With these considerations, I will develop another adaptive partitioning algorithm, which uses the phase error between the Gabor extrapolator and the LHA extrapolator as a partitioning criterion.

3.3.2 Adaptive partitioning with phase errors

Since the Gabor extrapolator is an approximation of the LHA (or GPSPI) extrapolator, we may use the phase difference between them as a partitioning criterion. The LHA wavefield

extrapolator (see definitions in section 2.4 of Chapter 2) can be written as

$$\hat{W}(x, k_x, \Delta z, \omega) = \exp\left(i\Delta z\sqrt{k^2(x,\omega) - k_x^2}\right), \tag{3.1}$$

where $k(x,\omega) = \omega/v(x)$ is the magnitude of the total wavenumber vector, Δz is the extrapolation step size, k_z and k_x are components of $k(x,\omega)$ in vertical and radial direction, respectively, and ω is the temporal frequency. This extrapolator works as a phase-shift operator on the input wavefield. The phase term in this operator is

$$\phi(x) = \Delta z\sqrt{k^2(x,\omega) - k_x^2},\ k^2(x,\omega) \geqslant k_x^2. \tag{3.2}$$

In the LHA extrapolator, the exact velocity at each output position x is used, giving the most accurate phase-shift operator among all phase-shift methods. As an approximation to the LHA extrapolator, the Gabor wavefield extrapolator is

$$\hat{W}_j(k_x, \Delta z, \omega) = \exp\left(i\Delta z\sqrt{k_j^2 - k_x^2}\right),\ j \in \mathbb{Z}, \tag{3.3}$$

and its phase term is

$$\phi_j = \Delta z\sqrt{k_j^2 - k_x^2},\ k_j^2 \geqslant k_x^2,\ j \in \mathbb{Z}, \tag{3.4}$$

where $k_j = \omega/v_j$. The Gabor extrapolator uses reference (average) velocities v_j in partitions instead of exact velocities at x. As a result, there is a difference in terms of phase between the LHA and the Gabor extrapolators. That is, there are phase errors when the Gabor extrapolation is used to approximate the LHA extrapolation. Therefore, the phase error can tell how accurate the approximation is, and it can be used as a criterion for a partitioning algorithm.

The total phase error between the LHA wavefield extrapolator and its Gabor approximation across the lateral (x) coordinate can be written as

$$E' = \left\| \arg\left[\hat{W}(x, k_x, \Delta z, \omega)\right] - \arg\left[\sum_{j\in\mathbb{Z}} \Omega_j(x)\hat{W}_j(k_x, \Delta z, \omega)\right] \right\|,\ j \in \mathbb{Z}, \tag{3.5}$$

where $\| \ \|$ denotes a certain norm, ϵ is the phase error in terms of arguments of the wave-field extrapolators, defined in the following:

$$\arg(a+ib) = \tan^{-1}\left(\frac{b}{a}\right), \tag{3.6}$$

where a, b are real used to define a complex variable, i is the imaginary unit.

In this thesis, we select L_1 norm for phase error estimation in equation (3.5), which is

$$E = \left\| \arg\left[\hat{W}(x,k_x,\Delta z,\omega)\right] - \arg\left[\sum_{j\in\mathbb{Z}} \Omega_j(x)\hat{W}_j(k_x,\Delta z,\omega)\right] \right\|_1, \ j\in\mathbb{Z}. \tag{3.7}$$

Definition of equation (3.7) in an explicit form gives:

$$E = \int_{\mathbb{R}} \left| \arg\left[\hat{W}(x,k_x,\Delta z,\omega)\right] - \arg\left[\sum_{j\in\mathbb{Z}} \Omega_j(x)\hat{W}_j(k_x,\Delta z,\omega)\right] \right| \mathrm{d}x, \ j\in\mathbb{Z}, \tag{3.8}$$

where $\mathbb{R}$ is the real line.

Considering one of those partitions (Ω_j), the phase term of the LHA extrapolator can be written as

$$E_j^{\mathrm{LHA}} = \arg\left[\Omega_j(x)\hat{W}(x,k_x,\Delta z,\omega)\right], \tag{3.9}$$

and the phase term corresponding to the Gabor extrapolator is

$$E_j^{\mathrm{G}} = \arg\left[\Omega_j(x)\hat{W}_j(k_x,\Delta z,\omega)\right]. \tag{3.10}$$

The phase error between the two extrapolators in partition $\Omega_j(x)$ is (compare to equation (3.7))

$$E_j^{\mathrm{diff}} = \left\| E_j^{\mathrm{LHA}} - E_j^{\mathrm{G}} \right\|_1, \ j\in\mathbb{Z}. \tag{3.11}$$

Equation (3.11) gives the phase error between the LHA and Gabor extrapolators in a partition. However, it is not obvious when using the absolute phase error as a criterion in adaptive partitioning. There are other methods for phase-error estimations (e.g., Ferguson

and Margrave, 2005) to address such a problem. A fractional (or relative) phase error estimation is selected in this thesis, which is

$$E_j^r = \frac{E_j^{\text{diff}}}{E_j^{\text{LHA}}}, \quad j \in \mathbb{Z}, \tag{3.12}$$

where E_j^r is defined as a relative phase error compared to the phase of the LHA extrapolator in the j^{th} partition, $\Omega_j(x)$. Equation (3.12) is used to calculate relative phase errors in a partition. Using equation (3.12) and summing over partition indices (subscript) gives the relative phase error between the total phases of the LHA and Gabor extrapolators across the lateral coordinate. Written explicitly, it is

$$E_r = \sum_{j \in \mathbb{Z}} E_j^r, \quad j \in \mathbb{Z}. \tag{3.13}$$

If a criterion (or partitioning parameter) is set in terms of relative phase error as a threshold, we can use it to create adaptive partitions. This adaptive partitioning method is called 'phase-error adaptive partitioning' (PEAP) algorithm in this thesis. It gives similar partitioning results as does the VGAP algorithm. However, these two methods are different because the partitioning criterion in the PEAP method is explicitly set as a phase error threshold between the LHA extrapolator and its approximation, which has more physical meaning. The VGAP algorithm does not deal with the phase error in a direct way. Therefore, the PEAP algorithm is more suitable for adaptive partitioning than the VGAP algorithm is at this point.

As will be shown, the smaller the phase-error criterion, the greater the number of created partitions. So the phase-error criterion can also be used to control accuracy and efficiency in the Gabor wavefield extrapolation.

A brief description of how to run the PEAP algorithm is provided here. The algorithm does adaptive partitioning in an opposite way compared to one done by the VGAP algo-

rithm. Given a velocity profile in 1D at a depth, the PEAP algorithm starts with a single wide partition across the lateral coordinate instead of many atomic windows in the VGAP method. Using the phase error given by equation (3.12) and a partitioning criterion (in relative phase error), the algorithm decides if the current partition needs to be divided into smaller ones. If the relative phase error in the partition is smaller than the criterion and it is the first and the only partition, no further partitioning is needed in this case; we know that this is a laterally homogeneous velocity profile or close to it, one partition is enough. The PEAP partitioning stops.

Otherwise, if the total relative phase error in the current partition (the first and only one) exceeds the partitioning criterion, the partition is split into two in the middle. The two new partitions are inspected and partitioned by the algorithm in the same way as is done to their parent partition. The process will be carried on in a recursive way until any one of existing partitions is divided into atomic windows or there is no possibility of further partitioning in it. This process is called ‘sweeping’. Following this, the algorithm proceeds into another process called ‘combining’.

Although each partition created by ‘sweeping’ process satisfies the partitioning criterion, there is the possibility that some neighbouring partitions may be combined as wider partitions and the partitioning criterion still holds in them. If this is true, these adjacent partitions should be consolidated, which helps to eliminates unnecessary small partitions.

From the description, we know that the first process in the PEAP breaks the lateral coordinate into as many partitions as possible given a phase-error criterion. The second process consolidates adjacent partitions into wider ones that still satisfy the phase-error criterion. This is the way that the PEAP algorithm optimizes localizing partitions in a velocity profile given a phase-error criterion.

Figure 3.2 shows partitioning examples given by the PEAP algorithm. The PEAP algorithm seems to have less difficulty when getting to 2D partitioning because calculation of phase errors for 2D extrapolators is straightforward. However, we can see from Figure 3.2 (c) and (d) that this partitioning algorithm is also direction dependent.

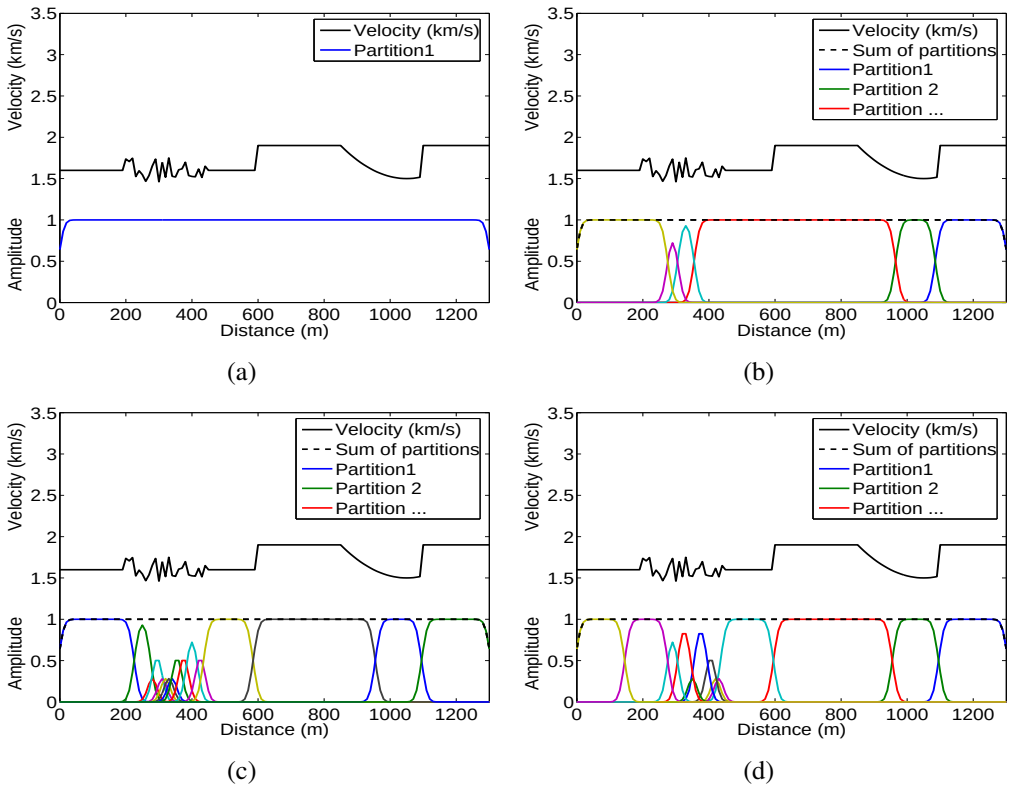


Figure 3.2: Partitions created by the PEAP algorithm for the Gabor wavefield extrapolation using various (relative) phase-error criteria. (a) 60% . (b) 40%. (c) 10% (from left to right). (d) 10% (from right to left).

To address the direction-dependent problem in both VGAP and PEAP algorithms, I will develop the third and the best adaptive partitioning algorithm for the Gabor wavefield extrapolation in this thesis. Additionally, this one is also easy to extend to higher dimensions.

3.3.3 Adaptive partitioning with lateral position errors

As introduced in the previous sections, the partitioning methods are either not physical or direction dependent. In this section, a new algorithm is developed for the adaptive partitioning using the 'lateral position error' as a criterion (Ma and Margrave, 2008), which will be defined shortly. Using this partitioning algorithm, those problems can be solved very well.

Considering a straight ray emanating at angle θ from a point $(x, 0)$ and traversing to $(x, \Delta z)$, we have the relationships in the spatial and wavenumber domains (see Figure 3.3 (a) and (b)); i.e.,

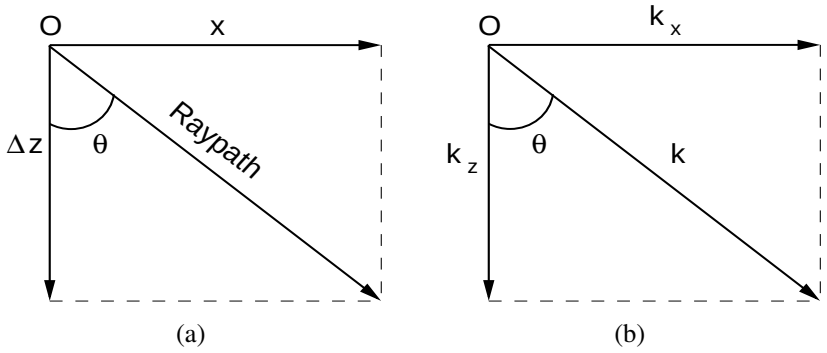


Figure 3.3: Geometry relationships of a local ray path between the vertical and lateral components in the (a) spatial domain and (b) wavenumber domain.

$$x = \Delta z \tan\theta \tag{3.14}$$

and

$$k = \frac{k_x}{\sin\theta} = \frac{\omega}{v}. \tag{3.15}$$

Let δx denote the perturbation in lateral position due to a variation in velocity δv. It is assumed that these quantities are sufficiently small and that we can relate them with the

differential calculus expression $\delta x = \frac{\partial x}{\partial v}\delta v$. Differentiating both sides of equations (3.14) and (3.15) with respect to v and rearranging gives

$$\frac{\partial x}{\partial v} = \Delta z \sec^2\theta \frac{\partial \theta}{\partial v} \tag{3.16}$$

and

$$\frac{\partial \theta}{\partial v} = \frac{k_x}{\omega}\sec\theta. \tag{3.17}$$

Substituting equation (3.17) into equation (3.16) and employing the differential calculus relation mentioned above yields

$$\delta v = \frac{\cos^3\theta}{\sin\theta}\frac{\delta x}{\Delta z}v, \quad \theta \in (0^\circ, 90^\circ]. \tag{3.18}$$

Equation (3.18) links the velocity variation δv with the position perturbation δx. Hereafter, we will refer to δx as the position error. The two quantities, δv and δx, relate in a way that depends upon the local ray angle θ, the local velocity v, and the extrapolation step size Δz. We use equation (3.18) to choose the reference velocities and these in turn define the adaptive partitions. The θ dependence of equation (3.18) means that we must choose a particular *design angle* at which to calculate equation (3.18). The trigonometric ratio in equation (3.18) decreases monotonically to zero as θ approaches 90 degrees as shown in Figure 3.4. This means that, in the limiting case, we would require infinitely many small partitions to bound the position error, which is clearly impossible. Thus, some moderate angle must be selected; and this designing angle has been chosen as 45 degrees (in all of the examples in this section). This does not necessarily mean that Gabor depth images are in error for greater angles as this ultimately depends on the lateral velocity gradient. In the homogeneous velocity case, this method is trivially correct for all angles regardless of the choice of design angle. However, in highly heterogeneous media, it can only be

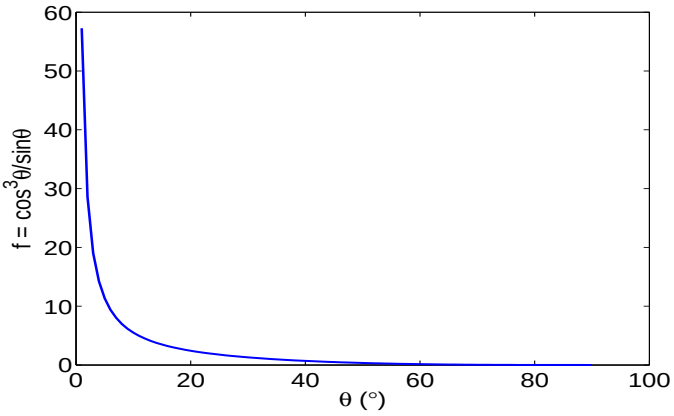


Figure 3.4: The trigonometric function of θ used in equation (3.18).

said that the position error is bounded for scattering angles below the chosen design angle. Also, the LHA extrapolator, which the Gabor extrapolator is approximating, is known to be inaccurate for large scattering angles in strong lateral gradients (Fishman and McCoy, 1985). To specify reference velocities, v_1 is chosen as the most frequently occurring velocity (easily done from a histogram of the discretely sampled $v(x)$) and then δv_1 is obtained from equation (3.18) as $\delta v_1 = a v_1$, where $a = \cos^3\theta \sin^{-1}\theta \delta x/\Delta z$. Letting v_2 denote the next higher reference velocity and v_3 the next lower, we can set the conditions

$$v_1 - \frac{1}{2}\delta v_1 = v_3 + \frac{1}{2}\delta v_3 = v_3\left(1 + \frac{1}{2}a\right) \tag{3.19}$$

and

$$v_1 + \frac{1}{2}\delta v_1 = v_2 - \frac{1}{2}\delta v_2 = v_2\left(1 - \frac{1}{2}a\right). \tag{3.20}$$

Using equations (3.19) and (3.20), calculation of v_2 and v_3 gives

$$v_2 = \frac{2v_1 + \delta v_1}{2 - a} = \frac{2 + a}{2 - a}v_1 \tag{3.21}$$

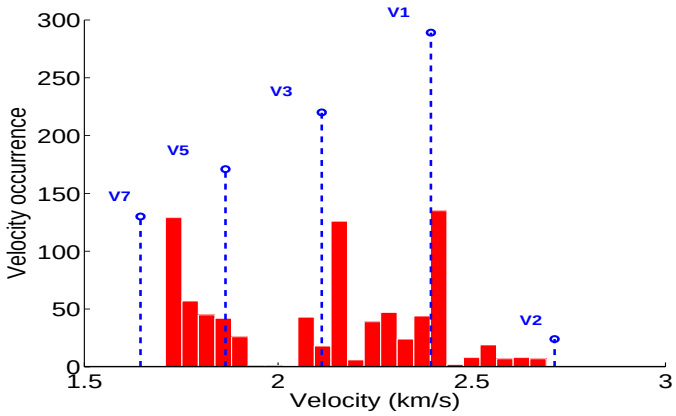


Figure 3.5: Reference velocities selected using equations (3.18), (3.23) and (3.24). Statistics of the velocities (histogram shown in red bars) from the Marmousi velocity model (Bourgeois et al., 1991) at depth of 750 m and the reference velocities (shown in blue–dashed vertical lines) selected using a lateral position error (δx) of 2.5 m.

and

$$v_3 = \frac{2v_1 - \delta v_1}{2 + a} = \frac{2 - a}{2 + a} v_1. \tag{3.22}$$

Proceeding in a similar way until the range of reference velocities exceeds the range of $v\,(x)$ defines a complete set of reference velocities (see Figure 3.5), which are

$$v_{2n} = \left(\frac{2 + a}{2 - a}\right)^n v_{1,} \quad n = 1, 2, \ldots \tag{3.23}$$

and

$$v_{2n+1} = \left(\frac{2 - a}{2 + a}\right)^n v_1, \quad n = 1, 2, \ldots. \tag{3.24}$$

The use of equation (3.18) in selecting the interval between reference velocities suggests that lateral position error will be bounded by δx. This bound may not be strictly held because equation (3.18) was derived with constant velocity theory; however, this is consistent with the spirit of the LHA approximation. As the design angle θ (contained in constant a from equations (3.19) to (3.24)) increases, more reference velocities are needed

in each extrapolation step, therefore, the extrapolation process is slower. As mentioned before, we have chosen 45 degrees as a design angle in all of the examples in this chapter.

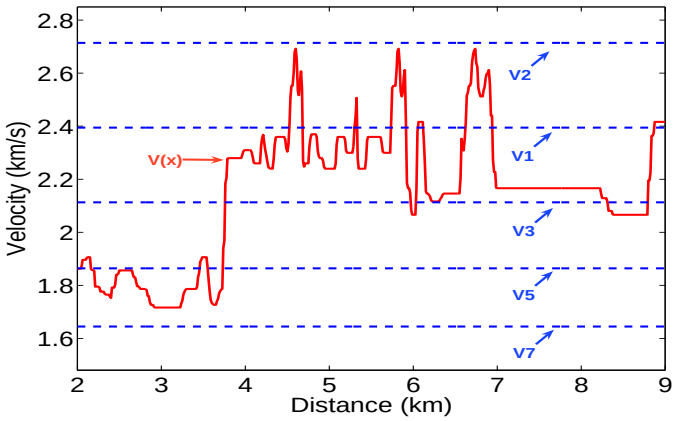


Figure 3.6: The reference velocities from Figure 3.5 (blue-dashed horizontal lines) superimposed on the exact velocity profile (solid red line) from the Marmousi velocity model at depth 750 m. These reference velocities will be used as measures to create indicator functions for adaptive partitions.

Once the reference velocities are selected, they can be used as measures (see Figure 3.6) to build indicator functions $I_j(x)$ defined by

$$I_j(x) = \begin{cases} 1, & |v(x) - v_j| = \text{minimum (over } j) \\ 0, & \text{otherwise,} \end{cases} \tag{3.25}$$

where $v(x)$ is the exact velocity at the lateral position x and v_j $(j = 1, 2, ..., n)$ are reference velocities chosen by equation (3.18). We require that the indicator functions form a POU and care must be taken to ensure that this is so. A problem arises if $v(x)$ is exactly midway between two reference velocities and it must be guaranteed that the point x is not assigned to both corresponding indicator functions. Any arbitrary but consistent rule will suffice, such as always assigning the point to the slower velocity.

A convolution of the indicator function I_j with the atomic window creates partitions (also forming a POU) corresponding to reference velocities used in Gabor wavefield extrapolation. This is done by

$$\Omega_j(x) = \frac{\int_{\mathbb{R}} I_j(x - x')\,\Theta(x')\,\mathrm{d}x'}{\int_{\mathbb{R}} \Theta(x')\,\mathrm{d}x'}, \tag{3.26}$$

where Θ stands for the atomic window. The atomic window can be any type of non-negative window, and here a Gaussian with half-width equal to the spatial grid spacing is chosen.

For 2D wavefield extrapolation, the partitioning scheme works in 1D; and this method can be easily extended to 2D partitioning required for 3D Gabor wavefield extrapolation (Ma and Margrave, 2006a) (see also details in Chapter 6). Here I limit the description to building indicator functions in 1D but this can clearly be extended to a second lateral dimension as needed for 3D imaging. The 2D indicator function for the j^{th} reference velocity is simply unity at those coordinates where the j^{th} reference velocity is closest to the local velocity, and zero otherwise. The position error criterion will not change nor will the selection of reference velocities. Once the indicator functions are built, then equation (3.26) simply becomes a 2D convolution, over both lateral coordinates, involving the 2D indicator functions and a 2D atomic window.

The partitioning algorithm just developed is called 'lateral-position-error adaptive partitioning' (LPEAP) algorithm.

Next, the partitioning method is illustrated with a few examples. The first example is a step velocity profile with a smooth transition zone in the middle. Figure 3.7 shows partitioning results using this partitioning method. The algorithm creates differing numbers of partitions given various lateral position error criteria.

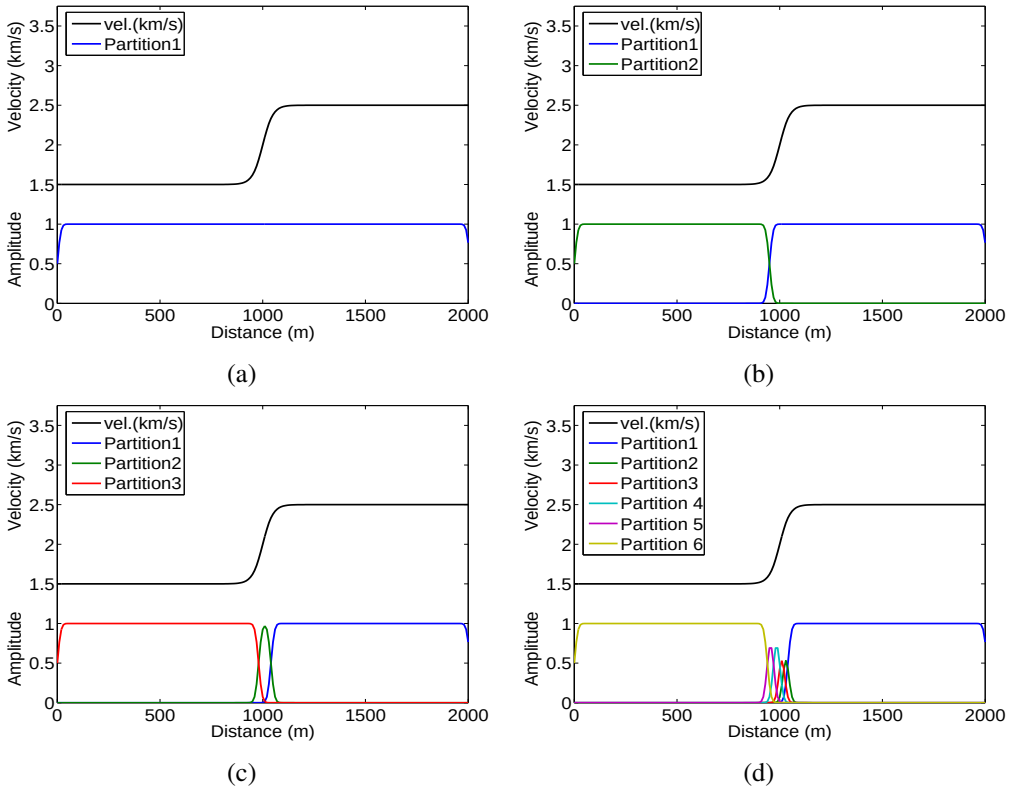


Figure 3.7: Partitions created by the LPEAP algorithm. Partitions created using a lateral position error of (a) 15 m, (b) 10 m, (c) 5 m and (d) 2 m.

These examples indicate that the lateral-position-error criterion can also be used to control the number of partitions, thereby affecting the computational cost and imaging efficiency.

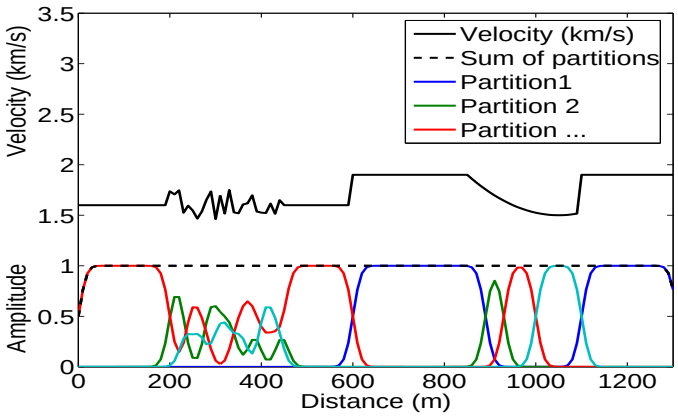


Figure 3.8: Partitions created by the LPEAP algorithm over the same velocity profile shown in Figures 3.1 and 3.2. The lateral position error used in this example is 2 m.

Shown in Figure 3.8 is a partitioning result from the same velocity profile as shown in Figures 3.1 and 3.2, used by the previous partitioning algorithms. I use this model again for the LPEAP algorithm, for the purpose of comparison to other two algorithms introduced in the previous sections. Also, the 1D velocity model is designed to be as complicated as those we would meet in the Marmousi velocity model, which I will use as an example in Chapter 4 to demonstrate prestack depth migration using the adaptive Gabor wavefield extrapolation. In this case, a specific position error criterion has resulted in partitions with many local maxima and complicated overlaps, which still sum precisely to unity (see the black-dashed line on top of partitions). In the pseudo-random segment (from 200 m to 500 m) of the velocity profile, those complex overlapping partitions indicate an implicit interpolation of wavefields during the Gabor wavefield extrapolation. Such complicated

overlaps were not observed in the partitions generated by other partitioning algorithms.

Another important point about partitions with local maxima is that this may also minimize the number of partitions. In the VGAP and PEAP algorithms, one partition only has one local maximum. In the LPEAP algorithm, one partition can have one to many local maxima. Actually, those atoms corresponding to the same reference velocities can be marked by the indicator function (see equation (3.25)). The convolution of the indicator function and those atoms (see equation (2.43)) gives such a partition with complex shape and many local maxima. This is one of the most beautiful properties of the LPEAP algorithm. And this works in 2D partitioning as well as it does in 1D partitioning (examples will be shown in Chapter 6).

Also, since the LPEAP algorithm works with the statistics from all velocities across the lateral coordinate, it does not need to start from one side of the velocity profile. Therefore, it is intrinsically direction independent. All these properties enable the LPEAP algorithm to be a more elegant and thus advanced partitioning algorithm over other methods.

As a summary for this section, I have introduced the velocity-gradient adaptive partitioning (VGAP) algorithm and developed two partitioning algorithms using phase errors and lateral position errors. All the three partitioning algorithms work quite well with 1D velocity profiles and will make highly efficient Gabor wavefield extrapolation possible by reducing spatial partitions across the lateral coordinate (Efficient depth imaging examples will be shown in Chapter 4).

In the next section, I will introduce a method that allows the Gabor wavefield extrapolation to be faster by reducing wavefield size, resulting in shorter Gabor (Fourier) transforms.

3.4 Spatial Resampling

The Gabor depth imaging method can be made much more efficient thanks to adaptive partitioning algorithms. Another way to improve the Gabor depth imaging efficiency is introduced in a space-frequency domain imaging method, known as the forward operator and conjugate inverse (FOCI) (Margrave et al., 2006). Wavefield spatial sampling at low frequencies can be decreased, resulting in a lower computational cost without loss of imaging accuracy. This spatial resampling technique can be implemented in the Gabor imaging method because this imaging method also works in the space-frequency domain.

Next, I will introduce the application of spatial resampling in the Gabor imaging method and demonstrate substantial reduction in computational effort.

3.4.1 Spatial resampling theory

Margrave et al. (2006) introduced the concept of spatial resampling in the FOCI depth imaging method, a space-frequency domain imaging method. Many wavefield extrapolation methods use operators of a fixed number of samples, which would run into problems at low frequencies. That is because if both the number of operator samples and the wavefield sample rate are constant, then as frequency decreases, the operator spans fewer wavelengths. In fact, at low frequencies, wavelengths are large and typical operators span only a small fraction of a wavelength. The result is: the fixed-length operator incurs poor phase and instability when working with low frequencies.

To address this problem, the spatial resampling technique was suggested by Margrave et al. (2006). In their paper, they mentioned the advantage of using this technique. That is, by resampling the lower frequencies to a larger sample size, an operator with a fixed

number of points actually spans a greater physical length. Resampling can improve the extrapolation speed because the lower frequencies of the wavefield have fewer samples and can therefore be processed more quickly. While the Gabor method uses a phase shift rather than a spatial convolution operator, we can still use spatial resampling to increase efficiency because the Fourier transforms require less effort as frequency decreases. The first step of the implementation is to break the frequency range into small bands as (Margrave et al., 2006)

$$[\omega_{\min}, \omega_{\max}] = [\omega_{\min}, \omega_1) \cup [\omega_1, \omega_2) \cup \ldots \cup [\omega_{n-2}, \omega_{n-1}) \cup [\omega_{n-1}, \omega_{\max}]\,,\ n \in \mathbb{Z}, \quad (3.27)$$

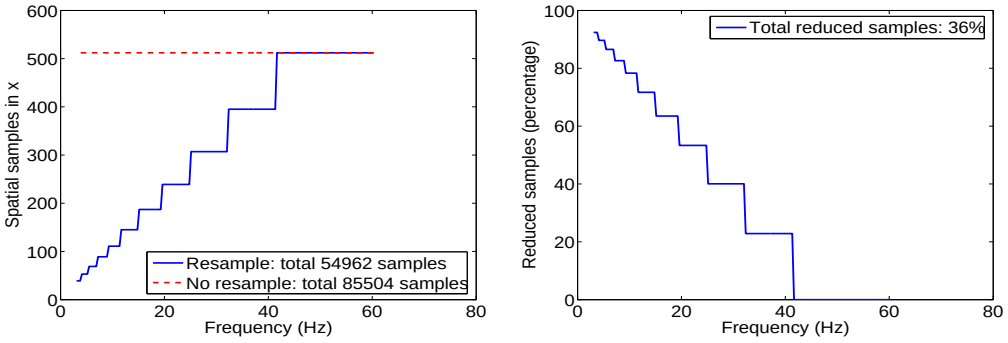
where $\omega_{\min}$ and $\omega_{\max}$ are the minimum and maximum frequencies of interest (i.e. signal frequencies), respectively, from band-limited seismic data. Suppose we have the original (fixed) spatial sampling interval as Δx; we resample it to new ones, Δx_j, according to various frequency bands, which are defined as

$$\alpha \left(\frac{\pi}{\Delta x_j} \right) \leqslant \frac{\omega}{v_{\text{crit}}} \leqslant \beta \left(\frac{\pi}{\Delta x_j} \right), \quad \alpha < \beta \in [0, 1],\ \omega \in (\omega_{j-1}, \omega_j),\ j \in \mathbb{Z}, \qquad (3.28)$$

where v_{crit} is the velocity chosen to define the highest evanescent boundary.

3.4.2 Spatial resampling examples

Using spatial resampling, we always have $\Delta x_j > \Delta x$, which means that the wavefield at low frequencies always has fewer spatial samples than the input wavefield. Fewer samples give faster Fourier transforms. Hence, we have a faster Gabor imaging than the one with fixed lengths of operators for all frequencies. Figure 3.9 shows the number of samples in using the spatial resampling method, from shot record 200 in the Marmousi data set (Bourgeois et al., 1991). Table 3.1 shows the same statistics in numbers.



(a) (b)

Figure 3.9: Comparison of samples with and without the spatial sampling. (a) Samples versus frequency: the lower the frequency, the fewer the samples in x; (b) change in samples (in percentage compared to no spatial resampling) versus frequency. Each stair-step (in blue) in (a) or (b) corresponds to a frequency band.

Table 3.1: Statistics of spatial samples in frequency bands.

Frequency band (Hz)	Frequency samples	Spatial samples
60.3 - 41.7	55	512
41.3 - 32.4	27	395
32.0 - 25.1	21	307
24.8 - 19.6	16	239
19.3 - 15.2	13	187
14.8 - 11.7	10	145
11.4 - 9.3	7	111
9.0 - 7.2	6	89
6.9 - 5.5	5	69
5.2 - 4.1	4	53
3.8 - 3.1	3	29

It is imperative that the spatial resampling be done as accurately as possible, and for this purpose, we use a Fourier method. Essentially, for each frequency, we apply a sharp low-pass filter to zero the wavefield at all wavenumbers greater (in absolute value) than ω/v_{crit}. We then shift the wavenumber Nyquist from $\pi/\Delta x$ to $\pi/\Delta x_j$, thereby truncating in the wavenumber domain most of the samples that were zeroed by our low-pass filter. Finally, we apply an inverse wavenumber Fourier transform and get back to the spatial domain with lower spatial sample rates in the wavefield.

3.5 Chapter summary

In this chapter I started with the analysis of computational cost in the Gabor wavefield extrapolation, and then presented two methods to improve efficiency of the Gabor wavefield extrapolation.

One of these methods is the Gabor depth imaging using partitions generated by adaptive partitioning algorithms given partitioning (or accuracy) criteria. I have described three adaptive partitioning algorithms in this chapter. The first is called the velocity-gradient adaptive partitioning algorithm, using the lateral velocity gradient as the partitioning criterion. This method, however, does not partition according to the physics behind the wavefield extrapolation, and it is also direction dependent and not easy to extend to 2D partitioning. The second is called the phase-error adaptive partitioning algorithm, which is also a good partitioning algorithm. The method is grounded in physics (phase errors) in the criterion for the adaptive partitioning. However, direction dependency is still a problem for this partitioning method.

The third partitioning method, called the lateral-position-error adaptive partitioning al-

gorithm, is the best and most preferred one in this thesis. The method gives physical meaning in the partitioning process and gets rid of the direction dependency. It also has other beautiful properties, such as easily extending to higher dimensions and multi-maxima in a single partition.

Following the adaptive partitioning, the spatial resampling method further increases the Gabor wavefield extrapolation efficiency by reducing the sizes of wavefields, resulting in shorter Gabor (Fourier) transforms at low frequencies.

Examples from adaptive partitioning algorithms and the spatial resampling method have demonstrated that those methods will be effective to improve depth imaging speed.

To be consistent with those demonstrations, I will use depth imaging examples to show improvement of imaging efficiency obtained by the application of adaptive partitioning and spatial resampling to Gabor depth imaging in the next chapter.

Chapter 4: Depth Migration with 2D Synthetic Data

In Chapter 2, I explored the theory of Gabor wavefield extrapolation. Impulse response examples demonstrate that the Gabor wavefield extrapolator works correctly. Continuing from those impulse response experiments in Chapter 2, in this chapter I will present prestack depth migration results using the adaptive Gabor depth imaging method. These depth images are obtained with a well-known benchmark synthetic data set, the Marmousi data set (Bourgeois et al., 1991), which was designed for testing depth migration algorithms.

In Chapter 3, I described methods for a fast Gabor wavefield extrapolation. In this chapter, improvement in imaging efficiency will be shown with imaging examples from the benchmark data set. The next section will start by introducing the benchmark data set.

4.1 Background of the Marmousi model

The Marmousi synthetic data set was initiated by the Institut Français du Pétrole (IFP) based on a true geological profile from the North Quenguela Trough in the Cuanza Basin in Angola, Africa (Versteeg, 1994). The original motivation behind this project was to create a complex synthetic data set to test advanced velocity estimation algorithms as well as depth migration algorithms.

Since the model was built according to a real-world geological model, it includes such complex structures as many dipping reflectors and other typical structural patterns: faults and folds. Figure 4.1 shows the Marmousi velocity model from distance $x = 2000\,\mathrm{m}$ to distance $x = 9000\,\mathrm{m}$, which is the main part of interest in the Gabor depth imaging.

The full dimension of the Marmousi model are: horizontal distance from $x = 0\,\mathrm{m}$ to $x = 9200\,\mathrm{m}$, and vertical depth from $z = 0\,\mathrm{m}$ to $z = 3000\,\mathrm{m}$. The original velocity grid was created as $4\,\mathrm{m} \times 4\,\mathrm{m}$, which was used to generate the synthetic data set. However, for depth imaging examples in this thesis, the velocity model has been down-sampled into a $12.5\mathrm{m} \times 12.5\,\mathrm{m}$ grid.

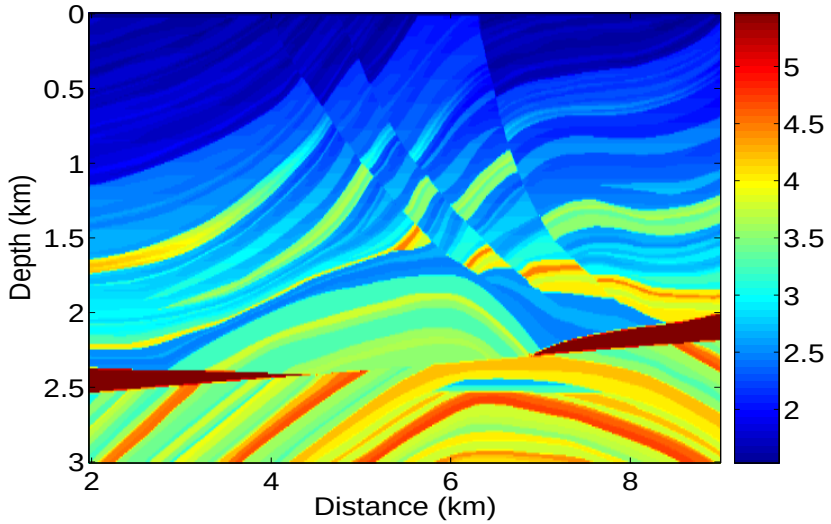


Figure 4.1: The Marmousi velocity model. The colour bar on the right codes velocities in km/s.

Not only has the Marmousi data set been used in velocity estimation tests, it has also been widely treated as a benchmark data set for depth imaging algorithm tests. The main purpose in a depth imaging test is to see if a depth imaging algorithm is able to reconstruct a reflectivity image from seismic data given an exact velocity model. Two items are always of concern in depth imaging tests. One is how accurately a depth imaging algorithm can reconstruct images from seismic data; the other is imaging speed.

Table 4.1: Acquisition parameters of the Marmousi data set.

Shot interval	25 m
Group interval	25 m
Line length	9.2 km
Channels/spread	96
Near trace/Far trace offset	200 m/2575 m
Time sample rate	4 ms
Energy source	Water gun (6-gun array)
First shot position	$x = 3000\,\mathrm{m}$
Last shot position	$x = 8975\,\mathrm{m}$

4.2 Marmousi data set

The Marmousi data set was 'acquired' (actually obtained from numerical modelling using a finite difference method) from west to east along the survey line, from small x to large x in the profile shown in Figure 4.1. The acquisition parameters are listed in Table 4.1 (after Versteeg, 1994). There are 240 shot records in the data set; one shot record includes 96 channels, each of which contains 726 time samples with a maximum recording time of about 2.9 seconds. Each channel has 5 hydrophones (measuring pressure) as a group. The source is a water-gun array, containing 6 water guns in a group.

4.3 Prestack Gabor depth migration with the Marmousi data

All depth migrations using the Gabor imaging method in this thesis have been done using prestack migration of individual shot records, each of which requires 241 extrapolation steps with a step size of $\Delta z = 12.5\,\mathrm{m}$. As introduced in section 2.5 of Chapter 2, the prestack Gabor depth migration starts with both seismic data (a shot record) and a simu-

lated point source. The data is extrapolated backwards down to deeper layers, while the source field is propagated (forward extrapolated) to the same depth, where the two extrapolated wavefields, downward (source) and upward (data), are compared to give the reflectivity image. This process is carried out recursively all the way down to the bottom of a model.

As for the reflectivity image, there are two methods to get this. The first is called cross-correlation imaging, which is written as (e.g., Claerbout, 1971; Lee et al., 1991; Biondi, 2002)

$$R_c(x,z) = \sum_{\omega} \psi_U(x,z,\omega)\psi_D^*(x,z,\omega), \tag{4.1}$$

and the second is called deconvolution imaging, which is defined as

$$R_d(x,z) = \sum_{\omega} \frac{\psi_U(x,z,\omega)\psi_D^*(x,z,\omega)}{\psi_D(x,z,\omega)\psi_D^*(x,z,\omega) + \epsilon \max\left[\psi_D(x,z,\omega)\psi_D^*(x,z,\omega)\right]}, \tag{4.2}$$

where x and z are lateral and vertical coordinates, respectively, ψ_U denotes the upgoing wavefield, ψ_D denotes the downgoing wavefield, both in the spatial frequency domain, $*$ in the superscript stands for the complex conjugate, ω is the temporal frequency, and ϵ is a small number (called the stability parameter) used in the denominator in equation (4.2) in case $\psi_D(x,z,\omega)\psi_D^*(x,z,\omega)$ becomes very small at some output positions (x,z), causing numerical instability during the imaging process.

Usually, deconvolution imaging gives images with higher resolution and better amplitudes (except for using stability parameter ϵ) in the final image. This is because the geometrical divergence effect is cancelled out (see equation (4.2)) by dividing the incident wavefield (ψ_D) with the reflected wavefield (ψ_U). As a result, it is very useful in amplitude analysis. Cross-correlation imaging provides us smoothed reflectivity images, but it does not preserve amplitudes in depth images. However, this imaging condition always enables

a robust imaging process and thus is a very popular imaging method (Valenciano, 2002). We may also use shot records (data) with balanced amplitudes by using automatic gain control (AGC) to generate images, which gives better amplitudes in final depth images but they may not be suitable for amplitude analysis. Unless otherwise explained, depth imaging examples shown in the following sections will use cross-correlation imaging condition.

The description above is the imaging process for one shot record. In each of these shot migrations, the Gabor wavefield extrapolator actually works as two parts, driving the data and the source field in opposite directions. Once this imaging process is done for all shot records, the final depth image can be obtained by stacking these shot migrations.

In the next section, I will give depth imaging examples for the Marmousi data set using the process just described. The imaging examples are also used to compare imaging efficiency of the Gabor prestack depth migration.

4.4 Efficient Gabor depth migrations

4.4.1 Fast Gabor depth migration using the adaptive partitioning

Although the Gabor depth imaging algorithm is sophisticated, the principle behind it is surprisingly simple. That is, source field and data are extrapolated at each depth, then a depth image is generated at a given depth by applying an imaging condition. The depth image throughout the model is constructed recursively from depth to depth.

The first depth image example from the Gabor depth imaging method is obtained using the same set of atomic windows at depths. This example is compared to images given by adaptive Gabor depth imaging using various partitioning algorithms introduced in Chapter 3. These imaging comparisons only involve depth imaging results from one shot record

because the Gabor depth imaging method using atomic windows (partitions) is very slow, as we shall see.

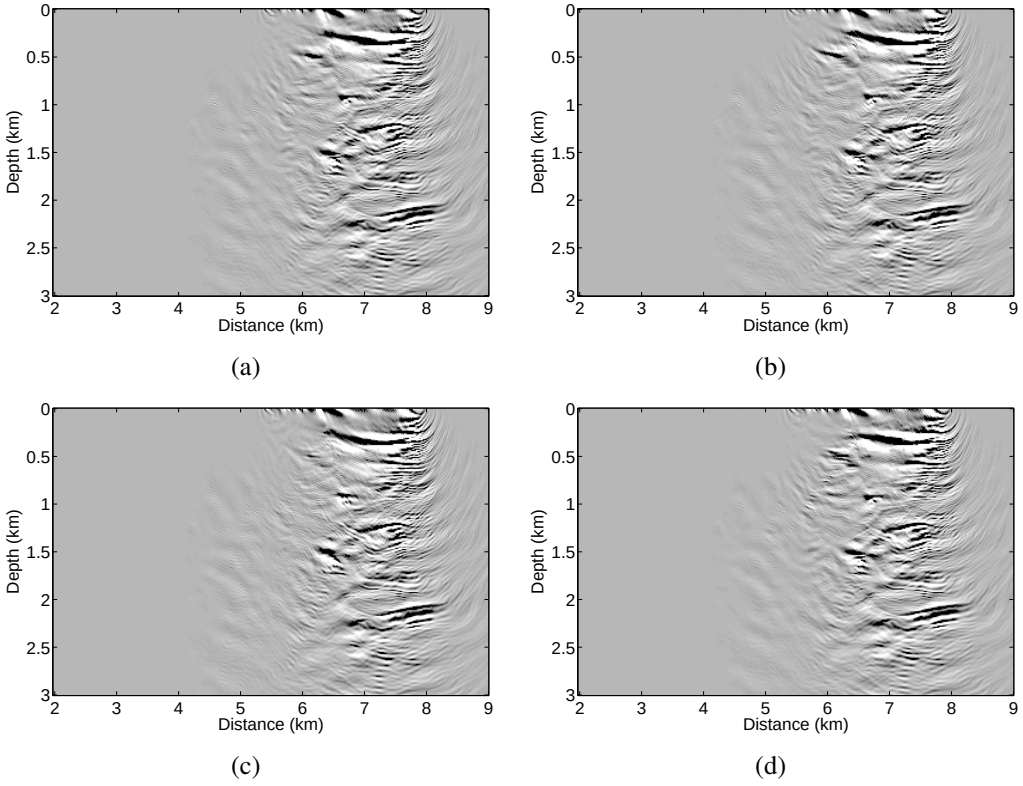


Figure 4.2: Prestack shot migrations from shot record 200 using various partitioning schemes. (a) Atomic windows. (b) Partitions from the VGAP method using a partition parameter of 5%. (c) Partitions from the PEAP method using a relative phase error of 5%. (d) Partitions from the LPEAP method using a lateral position error of 2.5 m.

Figure 4.2 shows shot migration results from shot record 200 using atomic windows and partitions given by the three adaptive partitioning algorithms in the Gabor depth imaging. Run times used by those shot migration schemes are shown in Figure 4.3 (a). In this chart, we see that Gabor depth migration using atomic windows required a huge amount of CPU time compared to run times from other methods. Although the image shown in

Figure 4.2 (a) is the most accurate one among those shot migrations, however, this one demands 10 hours on a single CPU (2.67 GHz). Other images from the shot migration using adaptive partitions are close to the one obtained using atomic windows, but run times all drop down quite dramatically. Shown in Figure 4.3 (b) are projected run times for all 240 shot migrations if those partitioning schemes were used in the Gabor depth migration. We can see from Figure 4.3 that using atomic windows in the Gabor depth imaging gives very low efficiency in the Gabor depth migration, though we can get accurate images. With adaptive partitioning algorithms applied in the Gabor depth imaging, we can have images with reasonable accuracy while keeping computation costs quite low compared to the inefficient one shown in Figure 4.2 (a).

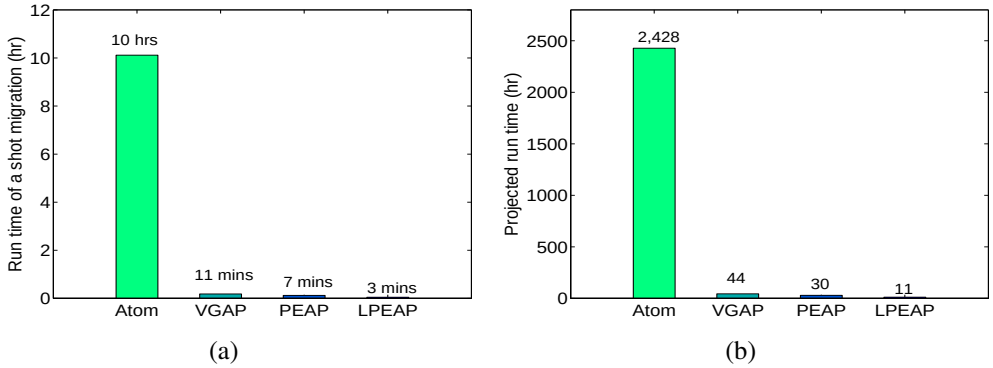


Figure 4.3: Statistics of run times. (a) Shot migration run times from shot record 200. (b) Projected run times for all 240 shot migrations in the Marmousi data set.

From run-time comparison, we understand that adaptive partitioning algorithms can make the Gabor depth imaging much faster while keeping acceptable accuracy in images.

For a quantitative comparison (not rigorous) between these images, let us take difference between the image (held in a matrix) shown in Figure 4.2 (a) and those shown in

Figure 4.2 (b), (c) and (d), respectively. We then take absolute values and sum them up as three numbers. These numbers are divided by the summation of magnitudes in the image shown in Figure 4.2 (a) and converted into a percentage. This gives us a comparison on imaging accuracy (assuming the one shown in Figure 4.2 (a) is an accurate one) for the adaptive Gabor depth imaging results. These percentage numbers are shown in Figure 4.4, from which we can tell that the three adaptive Gabor depth (shot) migrations give images close in accuracy to each other.

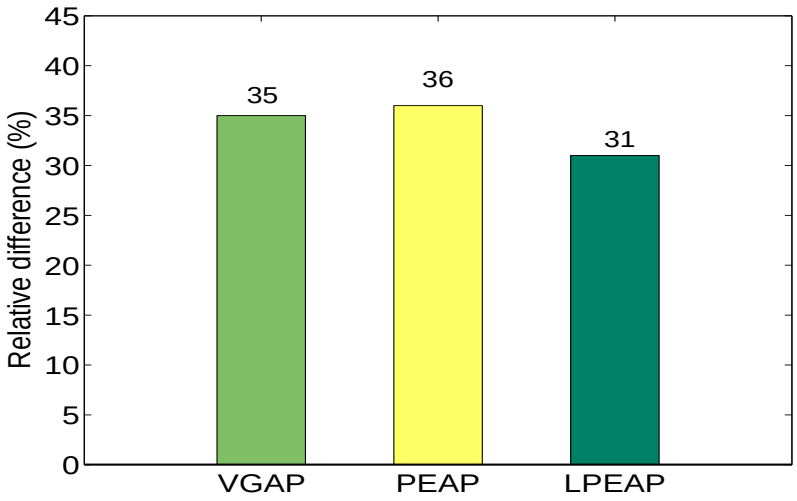
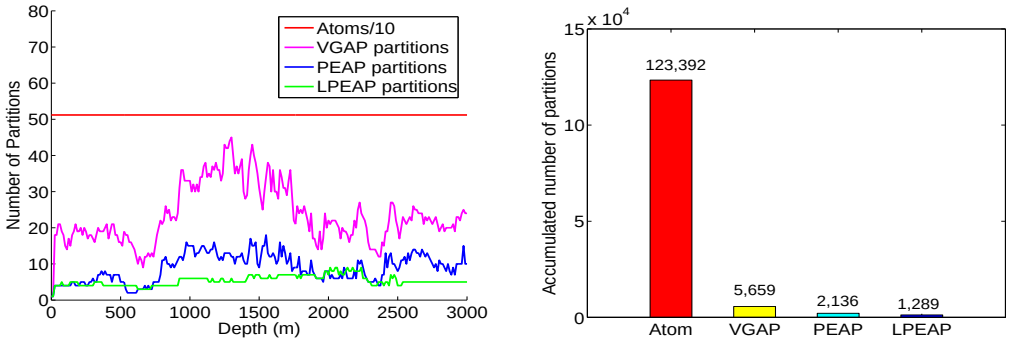


Figure 4.4: Relative difference between depth images (from shot record 200) given by the Gabor depth migration using atomic windows and using adaptive partitions.

As is known from computational cost analysis, the main contribution to such large improvement in imaging speed is from reducing the number of partitions. Figure 4.5 (a) shows statistics of partitions used at depths from these partitioning schemes when imaging with shot record 200. If we accumulate all those partitions (windows) at depths, we can see that the non-adaptive Gabor depth imaging uses a huge number of windows compared to

those used in the adaptive Gabor depth imaging. For example, compared to the number of windows used in 'Atoms' (scheme), the one used in the 'LPEAP' algorithm is negligible. The former one uses 120 times as many windows as the latter one does. We still have an acceptably accurate imaging result using the LPEAP (compare Figure 4.2 (a) and (d), and see Figure 4.4). However, the imaging efficiency has been improved enormously (see Figure 4.3).



(a) (b)

Figure 4.5: Comparison of the numbers of partitions (windows) used in various partitioning schemes in the Gabor depth migration.

The improvement in imaging efficiency from the adaptive partitioning side has proven to be a huge one. Starting from here, we still can make the Gabor imaging method faster using the spatial resampling method as described in the previous chapter. Section 4.4.2 will demonstrate the improvement in imaging speed using the spatial resampling method.

To be brief and clear, the adaptive Gabor depth imaging will only use the lateral-position-error adaptive partitioning (LPEAP) algorithm from now on. As claimed in Chapter 3, this partitioning algorithm turns out to be the most preferable and the best partitioning of all methods investigated in this thesis.

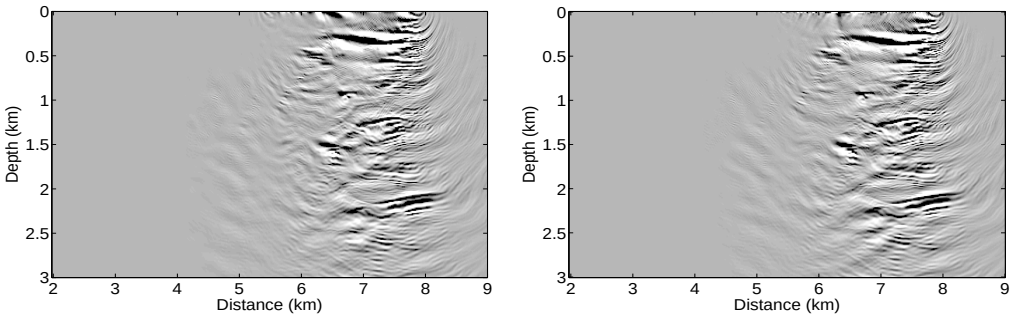
4.4.2 Fast Gabor depth migration using the spatial resampling

In a typical spatial-frequency domain wavefield extrapolation, fixed-length wavefields are used for all (temporal) frequencies. However, this is unnecessary for wavefields at low frequencies. The spatial resampling technique (Margrave et al., 2006) reduces sampling rates in wavefields to smaller ones at low frequencies. As a result, Fourier transforms are smaller and can be processed faster. Since the Gabor wavefield extrapolation also works in the (temporal) frequency domain, spatial resampling methods can also be used in the Gabor depth imaging method to improve imaging speed.

In the previous sections, fixed-length wavefield extrapolation was applied, and shot record 200 was used to compare the imaging speed. In this section, this shot record will be also used in depth imaging in order to make the imaging results from the spatial resampling method comparable to those shown in the previous section. Figure 4.6 (a) shows depth migration of shot record 200 using fixed-length (no spatial resampling) wavefields at all frequencies in the Gabor depth imaging. Figure 4.6 (b) shows the image from the same shot record but spatial resampling is applied. All other imaging parameters such as lateral-position-error criteria are the same. These two images look very similar to each other. With careful inspection, we can see subtle difference between them, which are due to accumulated numerical errors.

Run time of the shot migration shown in Figure 4.6 (a) is 170 seconds, and run time of the image shown in Figure 4.6 (b) is 62 seconds. Using spatial resampling, we make the depth imaging speed about 3 times faster for this shot record while preserving imaging accuracy.

Figure 4.7 shows depth images from all 240 shot migrations given by the adaptive



(a) (b)

Figure 4.6: Depth images from shot record 200: (a) Spatial resampling is not used. (b) Spatial resampling is used. The lateral position error used in the migrations is 2.5 m.

Gabor depth imaging using a lateral-position-error criterion of 2.5 m. The depth image shown in Figure 4.7 (a) is obtained without spatial resampling. For this image, 12 hours of CPU time is needed on the same PC mentioned in the previous sections. Shown in Figure 4.7 (b) is a depth image generated using the same depth imaging parameters except that spatial resampling is also applied during the depth imaging. Similar to the conclusions from shot migration 200, subtle difference between the two images can be seen with a careful inspection (e.g., in a small area around $(x, z) = (5.5, 1.6)$). Run time for the depth image using spatial resampling is reduced significantly, from 12 hours to 4 hours.

These examples have clearly demonstrated that spatial resampling reduces computation costs and makes the Gabor depth imaging method faster while keeping imaging accuracy. The spatial resampling method is another critical aspect of an efficient Gabor depth migration method.

As presented in Chapter 3, in depth migration from shot record 200, spatial resampling can reduce spatial samples to about 2/3 of the total samples used by fixed-length

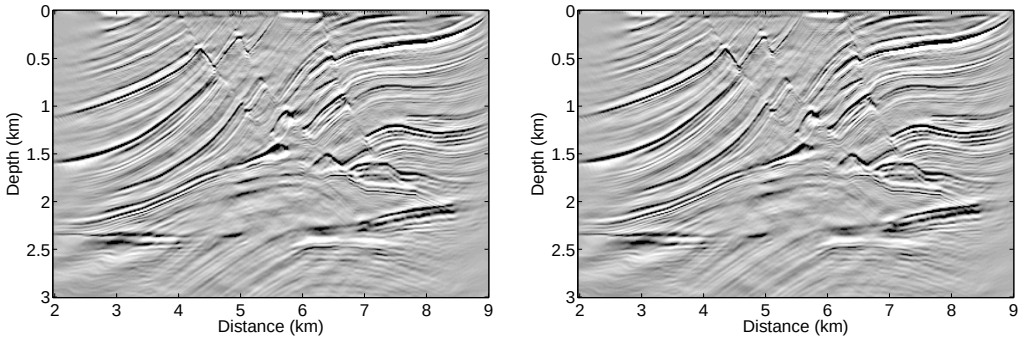


Figure 4.7: Prestack depth migrations with all 240 shots in the Marmousi data set. The lateral position error used in 2.5 m. (a) No spatial resampling. (b) Spatial resampling.

wavefields. In other words, 1/3 of the total samples are reduced (without losing imaging accuracy) after spatial resampling has been applied. According to the computational cost analysis in section 3.1, Chapter 3, the reduction of run time should be approximately 1/3. Why are we seeing double (2/3 instead of 1/3) the expected run time savings?

This question actually helps us to recognize an overlooked aspect in the Gabor depth imaging decribed in the previous sections. Temporal frequencies used in the Gabor wavefield extrapolation should only be those related to band-limited signals. In other words, the maximum temporal frequency used in the wavefield extrapolation should not be the Nyquist frequency (f_{Nyq}) but the maximum frequency ($f_{\max}$) from the input signal. So those temporal frequencies higher than $f_{\max}$ should be avoided. Since the number of frequencies is proportional to computational cost in the Gabor wavefield extrapolation (see computational cost analysis in section 3.1, Chapter 3), reduction of temporal frequencies results in a fast Gabor depth imaging.

In Marmousi imaging examples, we have f_{Nyq} = 125 Hz and $f_{\max}$ = 60 Hz. This

means that frequencies from 60 Hz to 125 Hz (about half of the total frequencies) do not contribute anything useful to final depth images. So extrapolation at those frequencies should be avoided, which would reduce the run times by another factor of 2 in the previous sections. When using spatial resampling, those frequencies higher than 60 Hz are not used in the wavefield extrapolation.

If those run times in the previous section are reduced by factor of 2, then run-time comparison between the Gabor imaging methods without and with spatial resampling would be 1.5 instead of 3 times. That is, 1/3 run time is actually saved with the application of spatial resampling in the Gabor depth imaging.

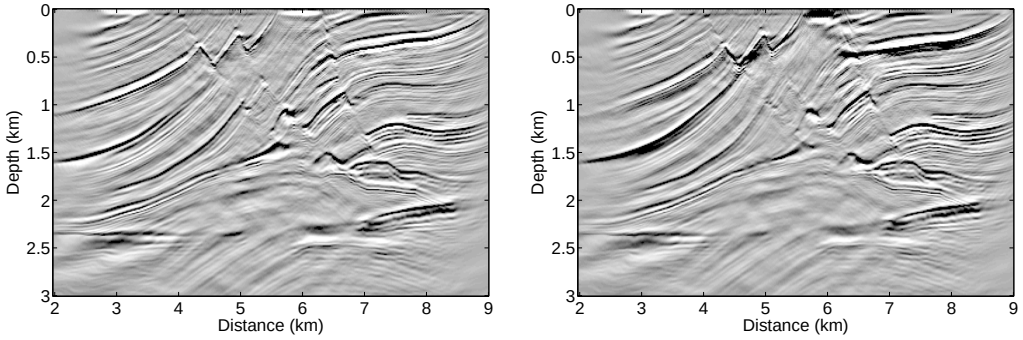
In the Gabor depth imaging method described in the following section and chapters, both the LPEAP algorithm and the spatial resampling method are used to create depth images.

Optimization of a given method can not improve both accuracy and efficiency at the same time, which motivates us to search for a balance between imaging speed and accuracy. In the next section, I will discuss the trade-off between imaging speed and accuracy using the Gabor depth imaging method.

4.5 Trade-off between imaging accuracy and imaging efficiency

We can use adaptive partitioning algorithms to choose between a fast but less accurate and a slow but accurate imaging process. From the single shot record imaging examples (see Figure 4.2 in section 4.4.1), we know that the Gabor depth imaging method using the partitions created by the LPEAP algorithm consumes only 1/200 of the run time needed by the Gabor depth imaging method using atomic windows (see Figure 4.3). The down side

of the adaptive Gabor depth imaging process is that it loses some accuracy in the depth imaging result (see Figure 4.4). Nevertheless, it has improved imaging speed quite a lot and imaging accuracy is still acceptable.



(a) (b)

Figure 4.8: Prestack depth migrations from the Marmousi data set using lateral position errors of (a) 5 m and (b) 10 m.

Figure 4.8 shows depth imaging results using lateral-position-error criteria of 5 m (a) and 10 m (b) in adaptive partitioning. In these images, lateral-position-error criteria (thresholds) are both larger than the one used in the previous examples from adaptive Gabor depth imaging. As a result, these imaging results should not be as accurate as the one using a smaller lateral-position-error criterion (2.5 m). When we look at these two depth images, we can see that imaging quality (accuracy) in (a) is still good. The image shown in (b) is an acceptable one though in some areas imaging quality becomes poor, e.g., in the shallow part, most notably near $(x, z) =$ (6.5,0.5), where steeply dipping faults are not clearly defined. Comparing (a) to (b) shows that there is obvious improvement in the upper part of the section when using a smaller δx, especially, the faulted areas near $(x, z) =$ (6.5,0.5), including slight improvement in the imaging target (a reservoir extend-

ing from distance 6.0 km to 7.5 km at depth of about 2.5 km) in terms of amplitude and phase coherence. Careful inspection also shows both amplitude and phase distortions in the lower resolution image.

Since a smaller lateral-position-error criterion (δx) introduces more windows, i.e., more reference velocities in Gabor imaging, more CPU time is needed in such a case. It is obvious that the imaging quality in the area, around $x = 5.5-6.5\,\mathrm{km}$ and $z = 0-0.5\,\mathrm{km}$ in Figure 4.8 (b), is quite low compared to that shown in Figure 4.8 (a). Similar conclusions can be made when inspecting the faulting area in the upper middle part ($x = 4.5 - 6.0\,\mathrm{km}$ and $z = 0.5 - 1.5\,\mathrm{km}$).

Figure 4.8 (a) and (b) both show images that are not as good as the one shown in Figure 4.7 (b). But run times in both cases go down (from 4 hours to 3 hours and 2 hours, respectively), meaning faster depth imaging compared to the one (see Figure 4.7 (b)) using $\delta x = 2.5$ m. In some cases, we care more about imaging speed, we may use large lateral-position-error partitioning criteria in the adaptive Gabor depth imaging. Otherwise, small lateral-position-error criteria should be used to obtain more accurate images.

We can always use the adaptive Gabor depth imaging method to trade off between imaging speed and accuracy. In the imaging examples shown in this section, using a lateral position error of $5\,\mathrm{m}$ in the Gabor imaging would be good trade-off between imaging accuracy and efficiency. The image shown in Figure 4.8 (a) appears as a good depth image for the Marmousi model and it uses a moderate amount of run time among the depth imaging results. Figure 4.9 shows a run-time comparison between four images shown in Figure 4.7 (a) and (b) and Figure 4.8 (a) and (b) in a sequence. They are labelled with A, B, C and D, corresponding to the order of the images.

Spatial resampling makes it possible to use fewer samples of wavefields at low fre-

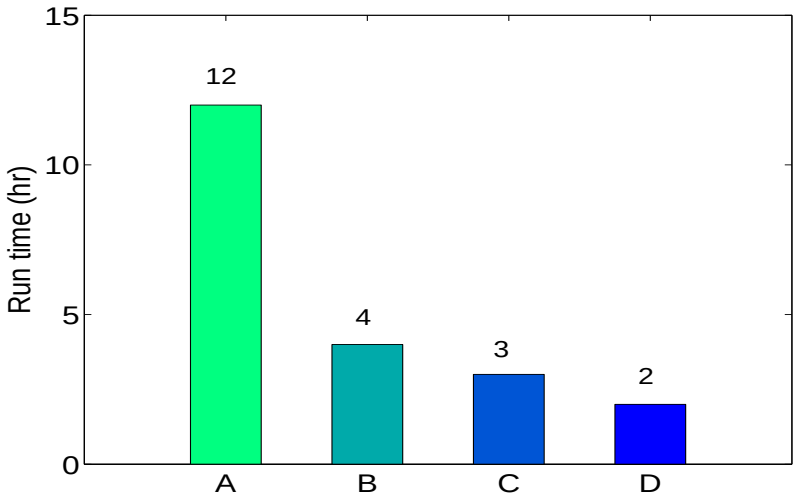


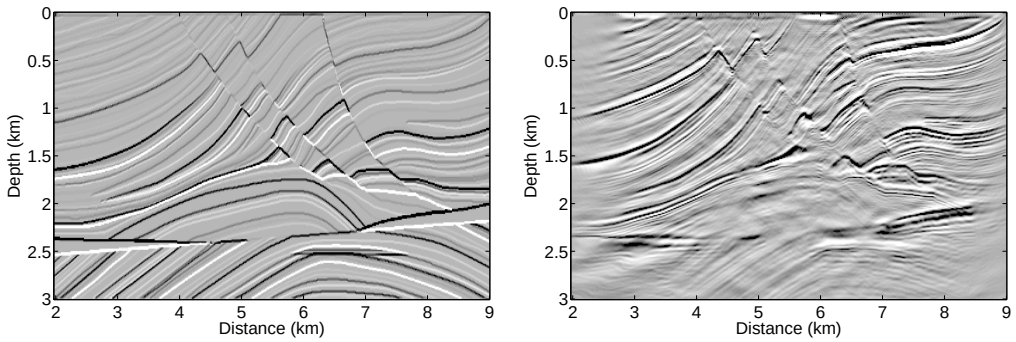
Figure 4.9: Run times used by Gabor depth migrations with various lateral position errors. A: 2.5 m (no spatial resampling). B: 2.5 m. C: 5 m. D: 10 m.

quencies while keeping accuracy in depth imaging. However, it seems not to allows us to trade off between imaging speed and accuracy as we can do with adaptive partitioning algorithms.

Next, I will use the Gabor imaging example using $\delta x = 2.5$ m (Figure 4.10 (b)), one of the best depth images given by the Gabor method, and compare it to the Marmousi velocity model, shown as a reflectivity image (Figure 4.10 (a)).

From this imaging result and other ones shown before, it can be concluded that the Gabor depth imaging method gives good depth images for the Marmousi model. A close examination of the image of the target reservoir (from distance 6.0 km to 7.5 km at depth of about 2.5 km) shows that it has been clearly imaged (compare this portion in the image to the corresponding part in the original velocity structures shown in Figure 4.10 (a)).

Finally, let us compare the Gabor imaging result ($\delta x = 2.5$ m) to that of a known



(a) (b)

Figure 4.10: The Marmousi velocity model and the constructed image given by the Gabor depth migration method. (a) Reflectivity image (from the velocity model). (b) Gabor depth image ($\delta x = 2.5\,\mathrm{m}$) of the Marmousi model.

space-frequency domain imaging method (Margrave et al., 2004, 2006) (see Figure 4.11 (a) and (b)). Figure 4.11 (c) and (d) shows two enlargements of the upper middle part from Figure 4.11 (a) and (b), respectively. From visual observations, we can conclude that, in this particular case, the Gabor imaging method ($\delta x = 2.5$ m) gives higher resolution than the space-frequency method of Margrave et al. (2006) (compare those dipping reflectors in Figure 4.11 (c) and (d)).

Compared to the FOCI algorithm, the Gabor algorithm requires roughly 50% more compute time using comparable MATLAB® codes on comparable machines. However, both methods have many parameters and it is likely that this comparison could vary dramatically using different parameter choices.

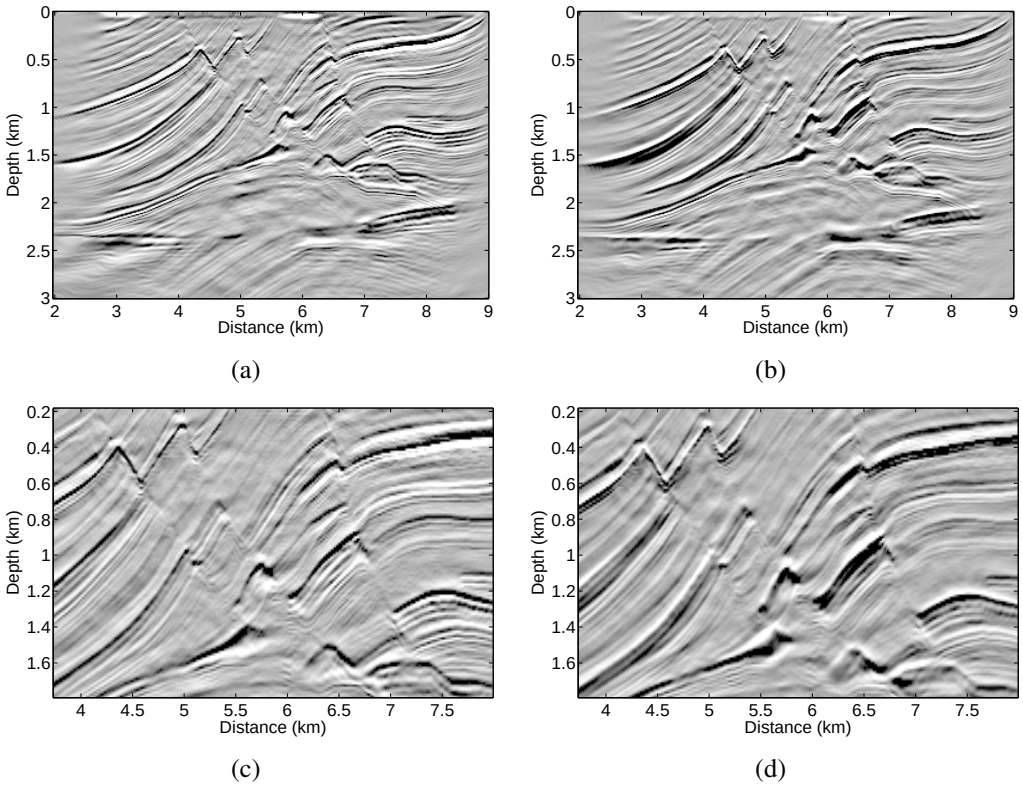


Figure 4.11: Depth migrations from the Marmousi data set. (a) Gabor depth imaging using δx =2.5 m. (b) FOCI depth imaging using a 51-point operator. (c) Zoomed-in part from (a). (d) Zoomed-in part from (b).

4.6 Gabor depth migrations with various p values

Equation (2.52) in Chapter 2 opens other avenues for the Gabor depth imaging. That is, by choosing various parameters of $p \in [0, 1]$, the Gabor imaging gives different schemes of depth imaging. In two special cases, extreme values, $p = 0$ and $p = 1$, are used and give simulations of the PSPI and NSPS depth imaging method, respectively (see equations (2.41) and (2.48) in Chapter 2).

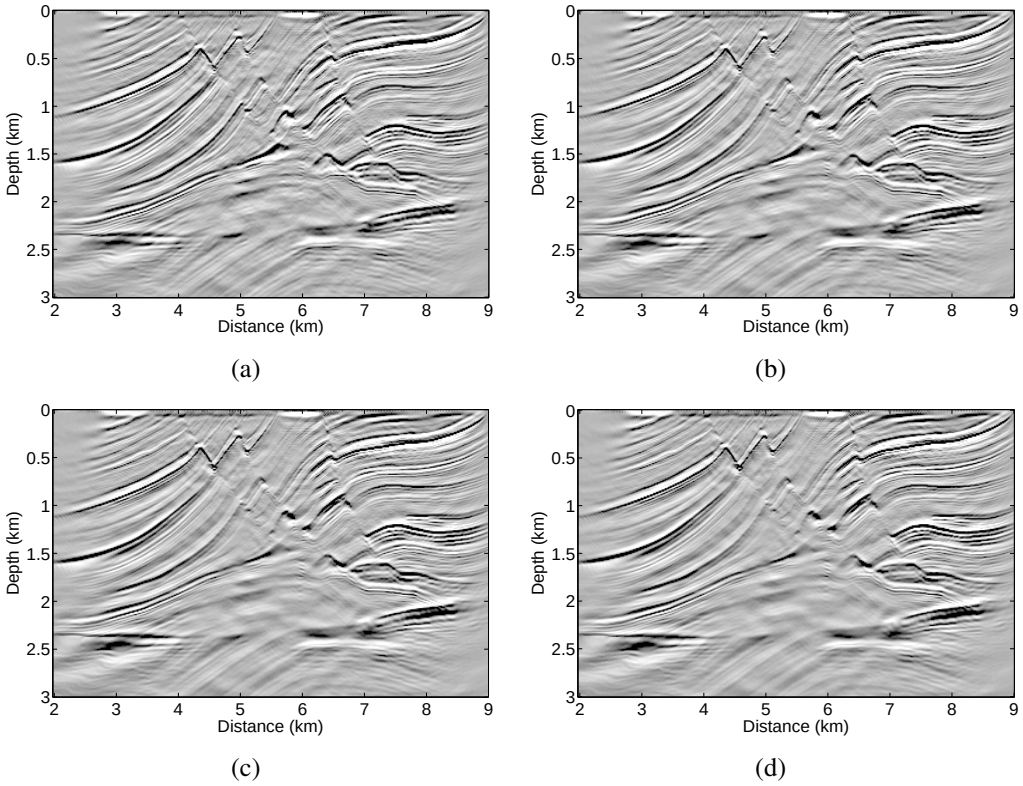


Figure 4.12: Depth migrations from the Marmousi data set using Gabor depth migration schemes with various p values (1). (a) $p = 0$ (PSPI). (b) $p = 0.1$. (c) $p = 0.25$. (d) $p = 0.5$.

Since $p \in [0, 1]$ and can be any value in the range, we have infinite ways of choosing

p. At this time, I have chosen $p = 0, 0.1, 0.25, 0.5, 0.75, 0.9, 1$ in equation (2.52). Imaging examples from the Marmousi data set with these selected p values are shown in Figure 4.12 and Figure 4.13. The PSPI and NSPS imaging results are also included for comparison.

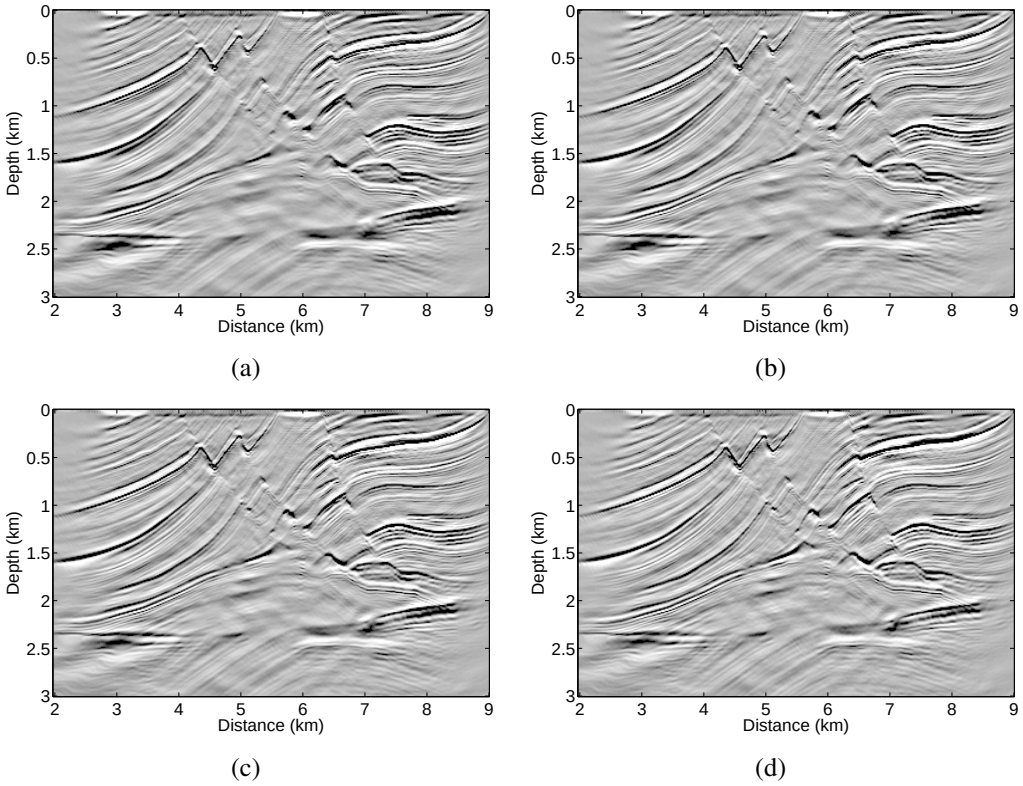


Figure 4.13: Depth migrations from the Marmousi data set using the Gabor depth migration schemes with various p values (2). (a) $p = 0.5$. (b) $p = 0.75$. (c) $p = 0.9$. (d) $p = 1$ (NSPS).

All these Gabor depth images with various parameters p are very similar at a quick look. Careful comparison shows very subtle difference. For example, in Figure 4.12, let us start from (a) ($p = 0$, PSPI) and compare this to (b) ($p = 0.1$). One of the most obvious changes is the small 'knob' located almost in the centre of the image, i.e., $(x, z) = (5.5, 1.5)$. The 'knob' starts becoming 'blurry' in (b) and gets worse in (c) ($p = 0.25$) and

(d) ($p = 0.5$). There is amplitude change in this order, too. Also, another careful inspection in four images shown in Figure 4.12 also gives us a similar conclusion, such as faulting areas in the upper-middle parts of the images. There are also slight variations in the target reservoir, getting blurry from (a) to (d).

Similar observations on the images in Figure 4.13 show that images of the 'knob' and faulting areas get better and better (clearer images) from (a) ($p = 0.5$) to (d) ($p = 1$, NSPS). And all the other imaging areas (e.g., faulting) mentioned in Figure 4.12 appear better and better from (a) to (d) in Figure 4.13.

A possible explanation of these phenomena is that when localization happens both before and after the wavefield is extrapolated at each depth step, there may be 'edge' effect or other unknown effects caused by 'windowing' in the wavefield. These effects then could be propagated or even amplified and cause 'blurring' in the final depth image. While in the PSPI ($p = 0$) or NSPS ($p = 1$) simulation, windowing in the wavefield only happens once at each depth step, either before or after it is extrapolated, which may account for better imaging results in these applications. However, this problem is open to more rigorous research.

4.7 Chapter summary

This chapter uses the Marmousi synthetic data set to test the Gabor depth imaging method. The adaptive partitioning algorithms developed for an efficient Gabor depth imaging method have been verified effective by prestack depth migration examples with the synthetic data set. Examples from shot migration using shot record 200 have been used for comparing the improvement in depth imaging speed, when applying adaptive partitioning algorithms in

the depth imaging process. As is demonstrated, Gabor depth imaging efficiency has been improved enormously thanks to the adaptive partitioning algorithms while the imaging quality has been preserved acceptably. Another merit of using adaptive partitioning algorithms is that they also help to control computational cost: we have high imaging quality when using higher computational effort, and lower imaging quality, otherwise. This gives us freedom when using the Gabor method for depth imaging.

One can make further improvement in imaging speed without losing any imaging accuracy by using spatial resampling in the Gabor depth imaging. This method is performed by means of altering spatial samples of wavefields at low frequencies, resulting in faster processing of those Fourier transforms corresponding to wavefields at low frequencies. It is not possible to trade-off between imaging accuracy and speed with this method.

Tests of the Gabor depth imaging method on Marmousi data set show high quality images, better than those of the FOCI space-frequency algorithm, though at greater computational expense.

Until this stage, the Gabor imaging method developed is for seismic data acquired from flat surfaces. To make this depth imaging method more flexible, I will develop a Gabor depth imaging method for seismic data acquired from irregular surfaces in the next chapter. And the modified Gabor depth method will be tested using both synthetic and real seismic data.

Chapter 5: Depth Migration with Topography

In Chapter 4, I showed depth imaging examples given by the Gabor depth imaging algorithms. The synthetic seismic data sets used there were created for testing depth imaging algorithms using seismic data acquired from flat surfaces. The purpose of those imaging tests is to see if a depth migration algorithm is able to reconstruct reflectivity images related to geological structures, given this synthetic seismic data and the accurate velocity models used to construct them. Usually, seismic data used in this type of imaging are acquired along a flat survey line (in 2D) or a flat grid (in 3D). The assumption is that seismic depth imaging algorithms are always used to migrate data from flat topography, or data that are adjusted as if acquired from flat topography. However, real seismic data are often recorded along highly irregular topography. If such imaging algorithms for flat-datum seismic data are applied to the migration of topographic seismic data with simple time shifting to a datum, the resulting images may not be well focused, or may focus to erroneous positions.

To address topographic problems in seismic depth migrations, we need seismic depth imaging algorithms able to deal with such seismic data directly where they were recorded. In this chapter, I will describe the method of using the Gabor wavefield extrapolation for seismic data from topography and show the imaging results from a 2D synthetic data set and a 2D real data set.

5.1 Topographic seismic migrations using regular imaging methods

Since seismic data are often recorded on irregular surfaces, all depth migration algorithms should be able to handle data from somewhat irregular acquisition settings. Generally, imaging methods for flat-datum seismic data (called 'regular imaging methods' in this thesis) can be used to image structures at depth using topographic seismic data, provided extra considerations are taken for them to work.

One of these methods given by Bevc (1997) uses a wave-equation datuming method, called 'flooding the topography'. In this method, the region between a flat datum (above the highest elevation on a topographic surface) and the topographic surface is flooded with a constant (not zero) velocity. Then topographic data are continued upwards using wave equation to the flat datum. Then the data can be migrated by a regular imaging method. Another method (Beasley and Lynn, 1992) is called the 'zero-velocity layer' method, which sets velocity as zero or a very small value between the flat datum and the topography. Then the algorithm migrates the data from the flat datum. Both of these methods transform recorded topographic data to a flat datum level above the highest elevation of the topographic surface. After the transform is applied to topographic seismic data, regular imaging algorithms are able to work and give correct topographic images.

In other methods, we can modify regular depth migration algorithms and use them to extrapolate wavefields directly from topographic surfaces, where receivers and sources sit, with careful arrangements of seismic data at depth steps (e.g., Reshef, 1991; Margrave and Yao, 2000). The advantage of the second method is very obvious. On one hand, the migration process or extrapolation can be started directly from the highest points on the topography instead of the top of the section, at a datum level. As the extrapolation steps

down into new depths, receivers at these depths on the surface are added in. Datuming with a wave equation method gives physical meaning to the process and an accurate image; however, it adds a fair amount of computation to the whole depth imaging process (Beasley and Lynn, 1992), which can always be avoided by using the method of direct extrapolation from topography, to be introduced in the next section. If we use static shifting, final imaging results are always inaccurate.

5.2 Topographic seismic migrations using the Gabor method

Gabor imaging theories for regular seismic depth migrations are described in Chapter 2. Velocity-adaptive data-partitioning algorithms (Grossman et al., 2002b; Ma and Margrave, 2007a) (see also in Chapter 3) have been developed to improve the imaging efficiency, which has been demonstrated in previous chapters. Run time reduction is substantial when using adaptive partitioning algorithms as well as spatial resampling techniques (Margrave et al., 2006; Ma and Margrave, 2007b) (see details also in Chapter 3).

In this section, I introduce the details of a method that enables the Gabor depth imaging method to handle wavefield extrapolation from an irregular recording surface. The application of the Gabor imaging algorithms in topographic migration is close to Reshef (1991) and Margrave and Yao (2000), which is called the laterally variable depth step scheme in this thesis. As I stated in the previous section, it will not need any datuming or data transform to a flat datum level.

For convenience, I include the formula of the Gabor wavefield extrapolation algorithm for regular seismic data (Ma and Margrave, 2006b) here (see also equation (2.41) in Chap-

ter 2), which is

$$\psi_{\mathrm{P}}(x,\Delta z,\omega)=\sum_{j\in\mathbb{Z}}\Omega_j(x)S_j(x)\int_{\mathbb{R}}\hat{W}_j(k_x,\Delta z,\omega)\hat{\Psi}(k_x,0,\omega)\exp(-ik_x x)\mathrm{d}k_x, \quad (5.1)$$

where ω is the temporal frequency, k_x is the wavenumber coordinate, $\hat{\Psi}(k_x,z,\omega)$ is the wavefield at depth z. Wavefield extrapolator $\hat{W}_j$ extrapolates $\hat{\Psi}(k_x,z,\omega)$ to a new depth $z+\Delta z$ as $\hat{\Psi}(k_x,z+\Delta z,\omega)$ (Δz is called the depth step) and is defined as

$$\hat{W}_j=\exp\left(i\Delta z k_j\right)=\exp\left[i\Delta z\sqrt{k_j^2-k_x^2}\right], \quad (5.2)$$

where $k_j=\omega/v_j$, v_j is the reference (constant) velocity for extrapolation inside partition Ω_j. The wavefield extrapolation using equation (5.1) is performed in various partitions Ω_j $(j\in\mathbb{Z})$; S_j is known as the split-step Fourier correction in Ω_j (Stoffa et al., 1990). Partitions Ω_j $(j\in\mathbb{Z})$ are actually very important in the Gabor depth imaging as I showed in Chapter 2 and they sum to unity along the lateral coordinate (Ma and Margrave, 2005) so that for all x

$$\sum_{j\in\mathbb{Z}}\Omega_j\left(x\right)=1, \quad (5.3)$$

where $\mathbb{Z}$ refers to the integer set. A Gabor prestack depth migration uses equation (5.1) recursively to extrapolate both source and receiver wavefields to any depth and then compares them using an imaging condition to estimate reflectivity.

In recursive depth marching schemes, the key role is played by the wavefield extrapolator $\hat{W}_j$, which extrapolates the wavefield from depth to depth. An inspection of equation (5.2) shows that $\hat{W}_j$ is closely related to the depth step Δz. If $\Delta z=0$, then $\hat{W}_j=\exp(0)=1$. This relation can be used to specify where the wavefield should be extrapolated in the Gabor imaging method from topography. We set the starting point as a flat datum above all sources and receivers. As a result, there is a region between the

flat datum and the topography, where no physical waves are expected to propagate. The seismic wavefields (shot records) should only be extrapolated when the marching depth reaches the topographic surface where data (shot records) are recorded. This is also true for point sources modelled at various source positions. To specify where to extrapolate, we allow the extrapolator depth step, Δz, to vary with lateral position. At a lateral coordinate where the depth level of the marching scheme is above the topography, then extrapolation is not needed, and we set $\Delta z = 0$, and $\hat{W}_j = 1$, meaning no phase-shift will be applied. Otherwise, $\Delta z \neq 0$, and a phase-shift will be performed accordingly using equation (5.1).

Let $z_t(x)$ be the depth to the topography at lateral coordinate x as measured from the highest point of the topography and Δz_{nom} be the nominal depth step. Assume the depth marching begins at the highest topographic point and after j steps is at depth $z_j = j\Delta z_{\text{nom}},\ (j \in \mathbb{Z})$. Then define

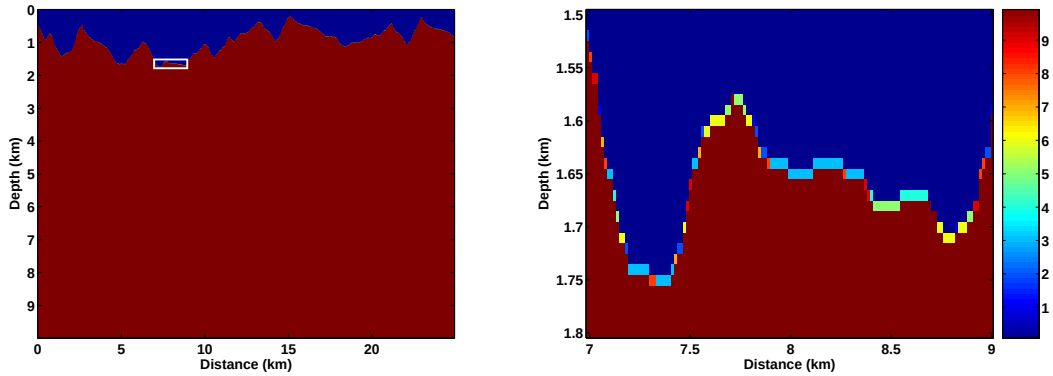
$$n_{z_j}(x) = \frac{1}{\Delta z_{\text{nom}}} \left[z_j - z_t(x) \right], \tag{5.4}$$

which is a count of the number of depth steps between the topography and the current depth. Negative $n_{z_j}(x) < -1$ corresponds to the case when the marching depth is above the topography. Then the actual depth step used in our algorithm for step $j+1$ is given by

$$\Delta z_{j+1}(x) = \begin{cases} 0 & n_{z_j}(x) < -1 \\ \left[1 + n_{z_j}(x)\right] \Delta z_{\text{nom}} & -1 \leq n_{z_j}(x) < 0 \\ \Delta z_{\text{nom}} & 0 \leq n_{z_j}(x) \end{cases} . \tag{5.5}$$

Equation (5.5) defines the extrapolation step sizes above, on and below the topographic surface where seismic data are acquired. Figure 5.1 shows the resulting depth steps on a topographic model that we will use to test the Gabor depth imaging method. With these step sizes well defined, the wavefield extrapolation with phase shift methods, including

the Gabor depth imaging method, will be straightforward whether the data are recorded on flat surfaces or topographic surfaces. That is, the extrapolation will be automatically controlled by the step sizes set up according to the topography.



(a) (b)

Figure 5.1: Step sizes created by equation (5.5). (a) Laterally variable step sizes for Model 94 topography. (b) Closeup of the region in white box shown in (a). Depth steps are color-coded in m.

5.3 Depth migration with 2D Synthetic Data from Topography

5.3.1 Velocity Model

The velocity model is a specific geological model, also known as Model 94 (Gray and Marfurt, 1995), which simulates complex geological structures from the Canadian Foothills. A 2D synthetic seismic data set was created with the velocity model and I use it to test the Gabor depth imaging algorithms designed to image topographic seismic data. Figure 5.2 shows the complex velocity structures in the model. The coordinates of the velocity model are from left to right 0 km to 25 km; and from top to bottom 0 km to 10 km.

Looking at velocity structures in Figure 5.2, we see that lateral velocity variations change very rapidly with depth, whether along or below the topography. Depths from 1 km to 6 km, for example, have the most rapid velocity variations in both lateral and vertical directions, which makes the wave propagation through them very complicated. We will see the corresponding seismic response in the shot records from topography above this model in the following sections. Also the relatively low velocities and irregular velocity variations just below the topography will cause a lot of trouble when imaging the structures in the shallow part below the topography. I will get to this point in section 5.3.3, where some initial imaging results will be shown and problems will be presented.

As we have seen from the imaging of the Marmousi data set, it is very challenging for depth imaging algorithms to reconstruct images from reflection seismic data recorded even on a flat surface with such complex velocity structures (see imaging examples in Chapter 4), let alone seismic data acquired from rapidly changing topography. Therefore, Model 94 will be a good model to test how well depth imaging algorithms work when imaging with the data set from topography.

To show lateral velocity variations, I pick velocity slices at two depths from Model 94 and show them in Figure 5.3 (a) and (b), respectively. These two slices are adapted at certain depths from Model 94, shown in two blue lines superimposed on the velocity model (see Figure 5.2). Also I demonstrate the adaptive partitions along the lateral coordinate, which are associated with a given lateral position error (5 m) criterion using the adaptive partitioning algorithm for the Gabor imaging.

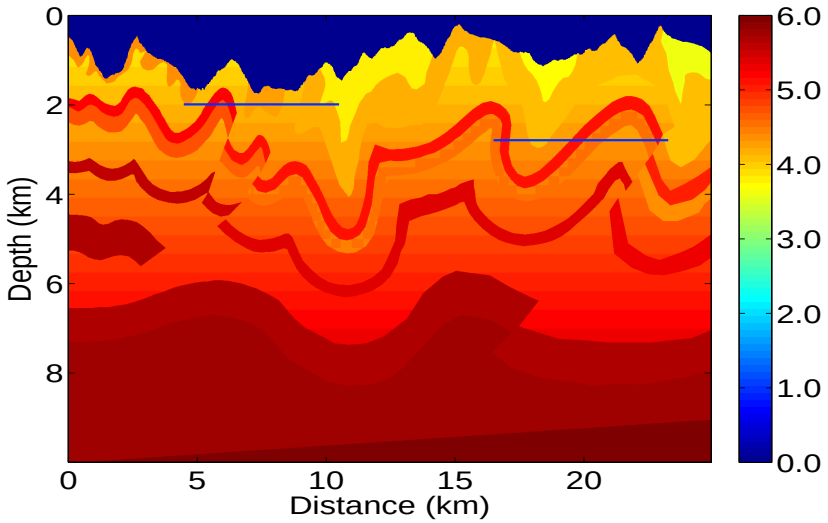
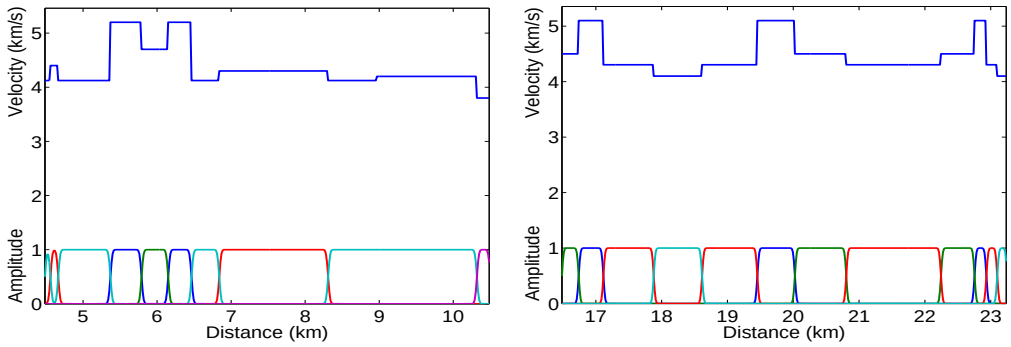


Figure 5.2: Model 94 velocity model. The colour bar codes velocities in km/s. The dark blue regions shown on top of the pictures are the air with a velocity of 0 km/s. Two blue horizontal lines at depths about 2 km and 3 km are where the example velocity slices are selected (see Figure 5.3).



(a) (b)

Figure 5.3: Velocity slices from Model 94. (a) A slice adapted at depth of 2 km, from x = 4.5 km to x = 10.5 km. (b) A slice adapted at depth of 2.8 km, from x = 16.5 km to x = 23.25 km. The blue lines on top of (a) and (b) show velocities in km/s. The partitions in the lower parts of (a) and (b) in colours are created by the adaptive partitioning algorithm using lateral position errors, corresponding to various velocity segments shown above. Those partitions in both (a) and (b) always sum to 1.

5.3.2 Model 94 synthetic data

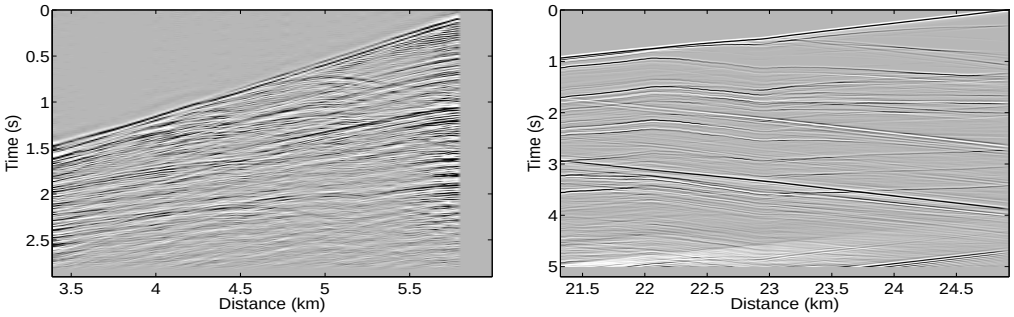
Acquisition parameters

Model 94 synthetic seismic data are created with 2D finite-difference modelling (Gray and Marfurt, 1995) using the velocity model introduced in the previous section. 278 shots are excited from the topographic surface, with a shot interval of 90 m. The first shot position is at 0 m with 240 channels, which represents the half spread on the right of the shot. The last shot spread has the half spread on the left containing exactly 240 channels, with shot position at 25 km. Spreads roll in and out from the left to the right and the full spread (both left half and right half) has a maximum of 480 channels in a shot record. Channel interval is 15 m. From a half spread to a full spread, 6 channels roll in or out each time when the shot position changes. Trace length is 2000 samples with time sampling rate of 4 ms. In section 5.3.3, the imaging step size in the vertical direction is set as 10 m.

Shot records from Model 94

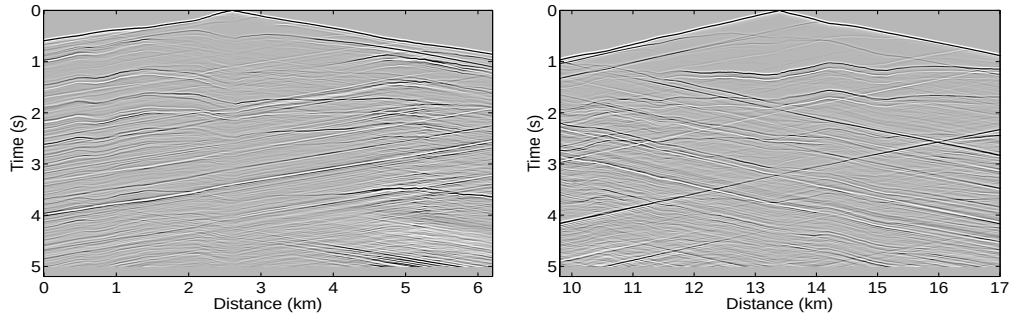
Seismic data with topography look different compared to those acquired from the flat surfaces. We can see this from comparing shot records from the Marmousi and the Model 94 data sets shown in Figure 5.4 (a) and (b), respectively.

From the shot records shown in Figure 5.4 (b), we can see that some events are 'crooked' or 'folded' (not as those in Figure 5.4 (a), close to hyperbolae) due to the time shift caused by the elevation difference between receivers on a topographic surface. These can be also seen in two other shot records from Model 94, shown in Figure 5.5. One of them is a full spread containing 480 channels, shown in Figure 5.5 (b). In the next section, I will set up parameters for the Gabor depth imaging and show some imaging examples and analysis on the imaging results.



(a) (b)

Figure 5.4: Seismic data acquired from flat and topographic surfaces. (a) A shot record from Marmousi data set (flat surface). (b) A shot record from Model 94 data set (topographic surface).



(a) (b)

Figure 5.5: Two shot records selected from Model 94 data set. (a) Shot record 30. (b) Shot record 150, a full spread containing 480 channels.

5.3.3 Initial imaging results and analysis

Imaging parameters

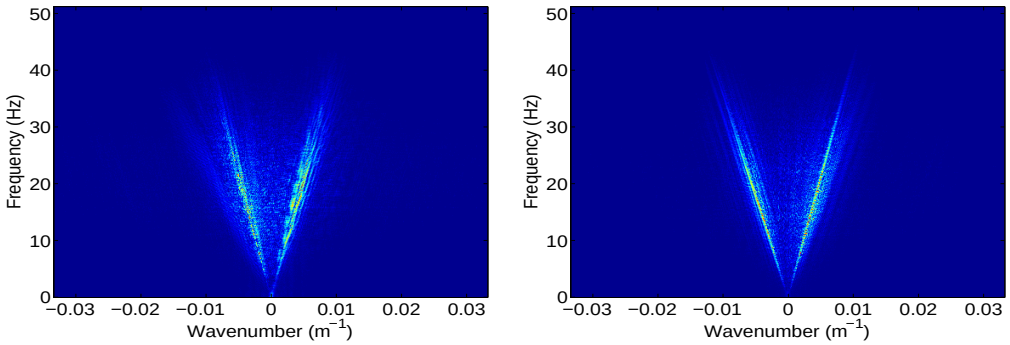


Figure 5.6: Selected FK spectra of the shot records from Model 94 data set. (a) FK spectrum of shot record 30. (b) FK spectrum of shot record 150.

Since the Gabor depth imaging is performed in the frequency-wavenumber (FK) domain, selection of parameters related to the frequency band and wavenumbers are very important to obtain images of good quality. The (temporal) frequency range of a band-limited wavelet is obtained by the spectral analysis of seismic traces from the Model 94 data set. As we can see from two selected FK spectra of the shot records in Figure 5.6, anything above 50 Hz is negligible. Similar conclusions can be drawn from FK spectral analysis on other shot records. So I conclude that the maximum frequency for signals in this data set is about 50 Hz. More details can be seen from seismic traces which are Fourier transformed. Some example traces are shown in Figure 5.7 (a) to (d), where we can reach the same conclusion about the maximum temporal frequency. Also one can see from Figures 5.7 that the dominant frequency is about 20 Hz. Based on these analysis, I designed an Ormsby wavelet with a frequency range of [3 10 30 50] Hz for seismic source

modelling when doing the Gabor depth imaging for the Model 94 data set.

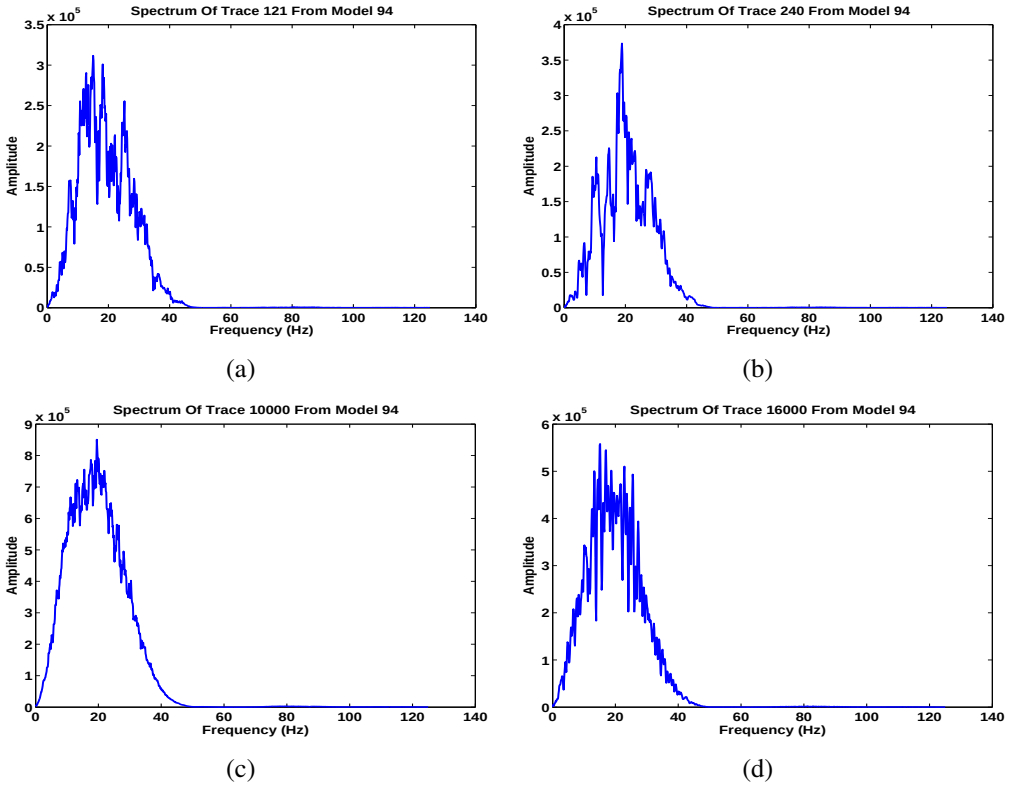


Figure 5.7: The temporal spectra of selected traces from Model 94 data set.

Shot migrations

First, let us see some single-shot migrations from Model 94. I use the shot migrations from the shot records mentioned in the previous section, i.e., shot 30 and shot 150. In creating these shot images, I used gained (amplitude-balanced) seismic data and the cross-correlation imaging condition. Figure 5.8 (a) shows the migration result from shot record 30. Put side by side with it is the reflectivity image for comparison, corresponding to the velocity piece used in the shot migration (see Figure 5.8 (b)). Figure 5.8 (c) shows the shot

migration of shot 150, and the related velocity piece is also shown as reflectivity image in Figure 5.8 (d).

From these shot migrations, we can see that one shot illuminates only partial structures because of the limited aperture from a point source on the surface. However, many such point sources will give sufficient illumination of structures beneath the surface. So stacking all of the shot migrations will help to obtain the whole picture of those structures. We will see this in the next section.

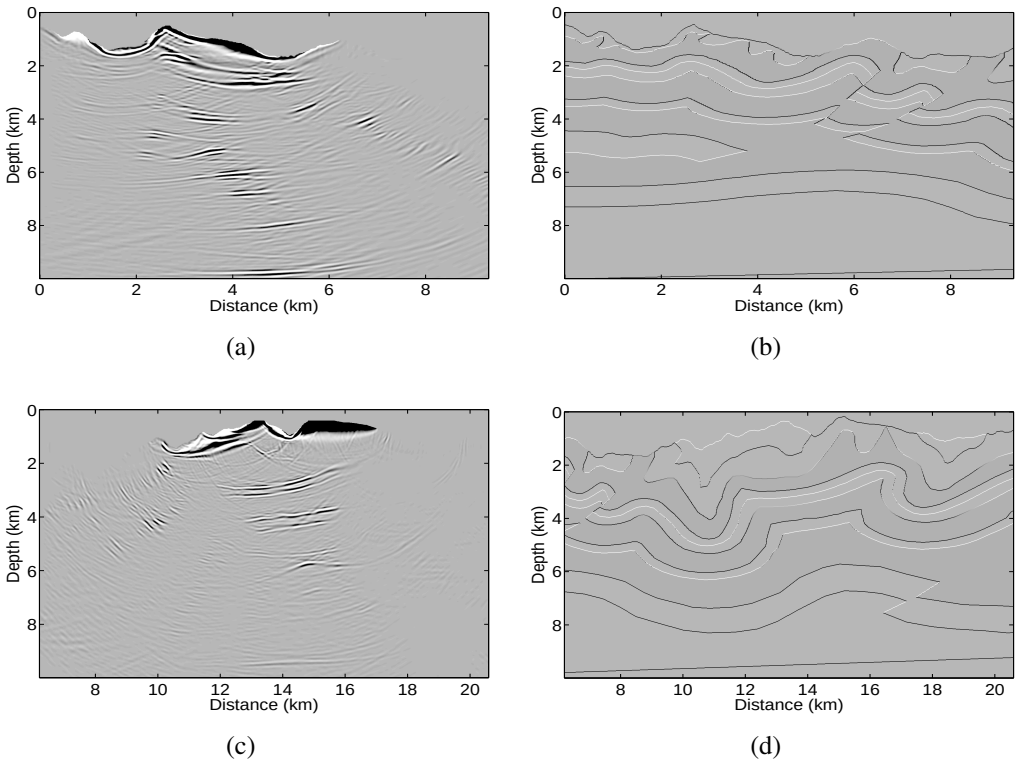


Figure 5.8: Shot migrations and reflectivity image pieces. (a) Image with shot record 30. (b) Reflectivity image piece related to shot 30 migration. (c) Image with shot record 150. (d) Reflectivity image piece related to shot 150 migration.

Stacking of all shot migrations

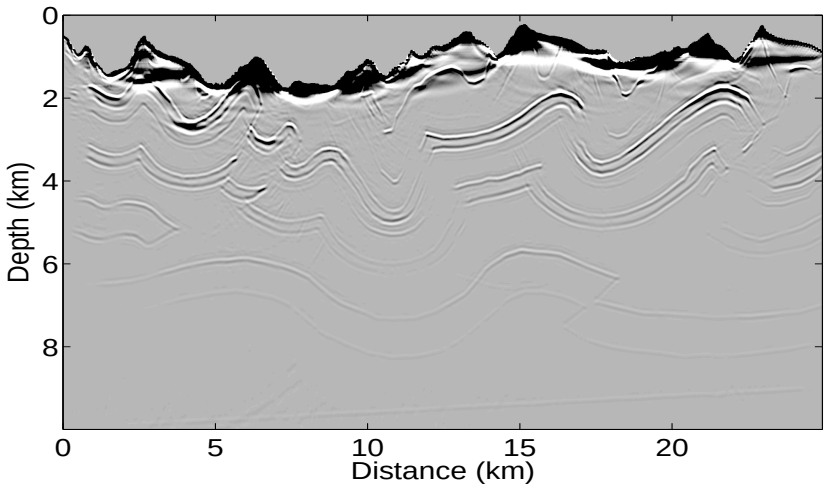
Putting all 278 shot migrations together gives a depth image shown in Figure 5.9 (a). A reflectivity image of the whole model has been shown in Figure 5.9 (b) for comparison. Close comparison of the images tells us that structures deeper than 1.5 km are mostly well imaged. However, the near surface part has been heavily biased by high amplitudes, appearing as black colour, which shades structures in the near surface (just below the topography).

In the next section, I will analyze reasons for the amplitude problems in the image and present methods for removing them.

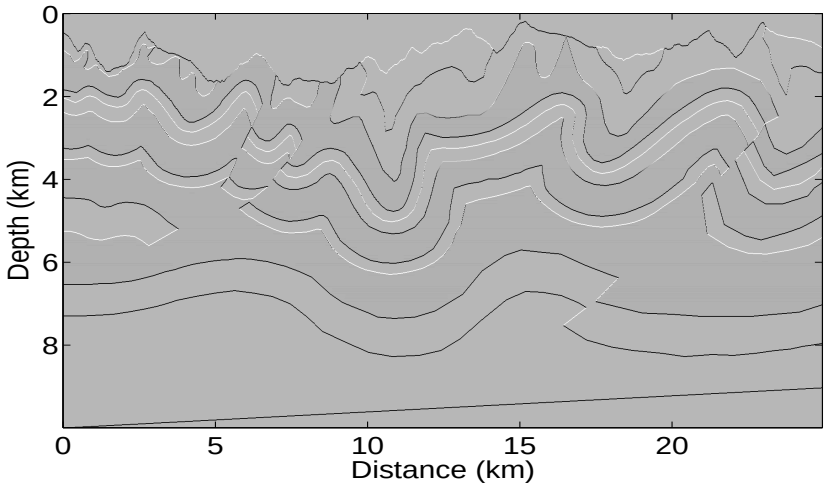
5.3.4 Problems in depth migrations and the solutions to them

There are several possible reasons for the amplitude problems. In the Gabor depth imaging algorithm, evanescent components in wavefields (see equations (2.38), (2.39) and (2.41) in Chapter 2) are suppressed using the second formula in equation (2.39). However, there is still a possibility that these evanescent components can leak into the wave-like region from areas close to the evanescent boundary during the wavefield extrapolation. Therefore, the first reason for the amplitude problem is that evanescent components may creep into extrapolated wavefields during phase-shift operations. The second reason is that if direct arrivals in shot records are included in the imaging, they may be the main contribution for high-amplitude contamination in the near surface image. Another reason is possibly low frequency components in wavefields, which may be amplified when extrapolated near the topographic surface. With these in consideration, I have set up a series of tests to assess these possible causes.

In the first test, I use dip filters (or wavenumber filters) in the wavenumber and fre-



(a)



(b)

Figure 5.9: Model 94 images. (a) Stacking of 278 shot migrations. (b) Reflectivity image.

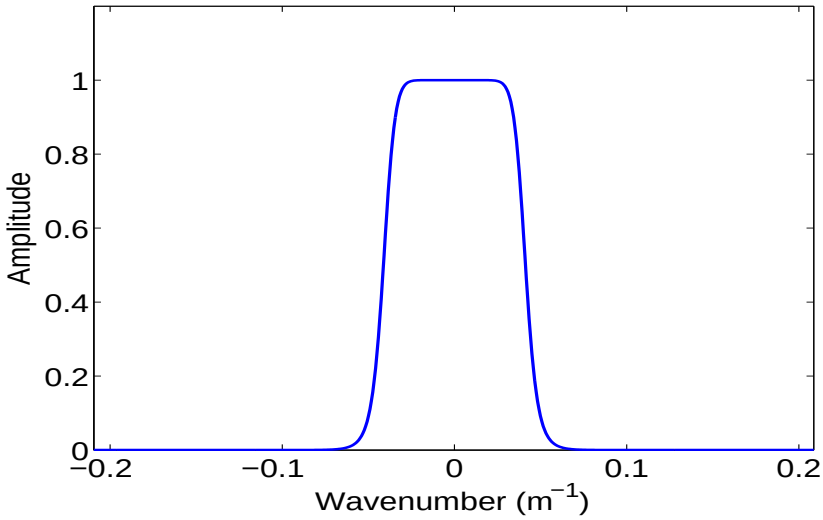


Figure 5.10: A wavenumber (dip) filter used in the Gabor imaging of Model 94.

quency domain when carrying out the wavefield extrapolation. Figure 5.10 shows the wavenumber spectrum of one of the filters designed for this imaging test. The boundary of the filter varies with frequencies and wavenumbers, and is defined as a 1D Butterworth filter (e.g., Hale and Claerbout, 1983),

$$filt\left(k_x, f, \theta_{\max}\right) = \frac{1}{1 + \left(\frac{k_x \sin\theta_{\max}}{fv}\right)^n} \ , \tag{5.6}$$

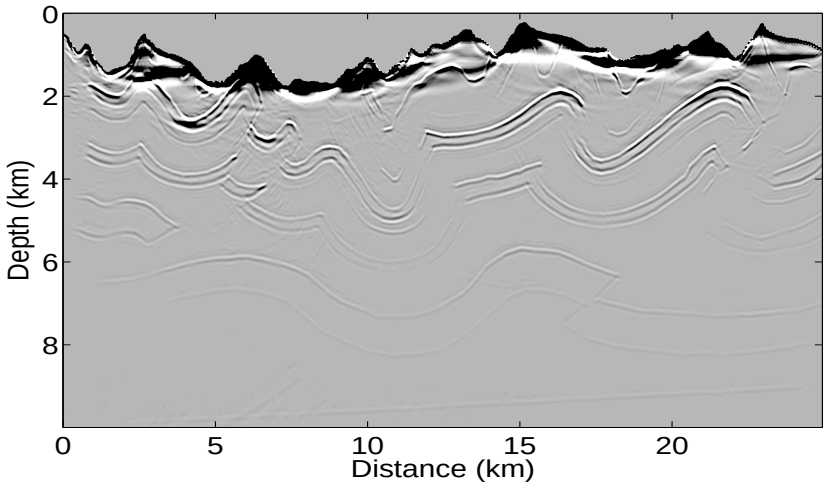
where k_x is the transverse wavenumber coordinate, $\theta_{\max}$ is the maximum dipping angle, v is velocity, f is frequency and n is the order of the Butterworth filter. Figure 5.10 shows the shape of the dip filter formulated in equation (5.6) with $\theta_{\max} = 80°$, $v = 4000$ m/s, $f = 30$ Hz and $n = 12$.

The dip filter is applied to wavefields in a recursive way at depths. Figure 5.11 (a) shows the depth image from Model 94 using dip filtering defined in equation (5.6). If one uses visual observation to compare Figure 5.9 (a) and Figure 5.11 (a), it would be

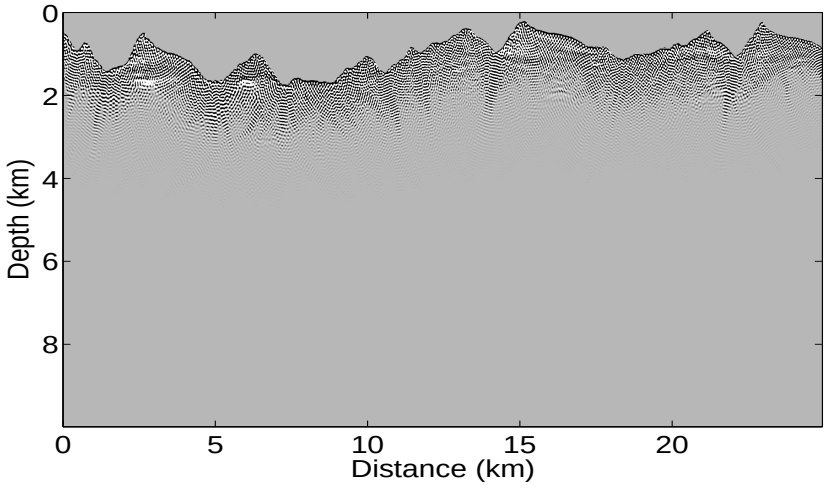
hard to tell the difference between the two images. However, by using the difference between the two images (with and without dip filtering), we are able to see something very clear in Figure 5.11 (b) . We can see that dip filters do remove something from high amplitudes near surface, i.e. something that is related to evanescent components during the wavefield extrapolation has been eliminated. However, this image is still not good enough. Therefore, something else may also help to remove high amplitudes in the near surface image.

The second test is designed to see if high amplitudes are related to low frequency wavefields. A low-cut frequency filtering process on the depth image is given in Figure 5.12 (a). Comparing with the image in Figure 5.9 (a), we know there is improvement in the near surface region after this application. We can see that some structures stand out as the high amplitudes near surface are filtered out. However, the resolution of the whole image has been lowered, and the structures near the surface still have distortions.

The third test is used to see if direct arrivals contribute to the high amplitude problem. To see this, we have to go back to raw shot records and mute direct arrivals before migration. Let us use shot record 150 as an example to demonstrate this. Figure 5.13 (a) shows the original shot record 150 for comparison. Figure 5.13 (c) shows shot record 150 with direct arrivals muted. Figure 5.13 (b) and (d) show the images corresponding to shot records shown in (a) and (c). From these two shot migrations, we see that when direct arrivals are muted in shot records, the resulting imaging result gets much better. Not only have high amplitudes been removed, showing a better balance in amplitude in the image, but also some near surface structures shadowed by high amplitudes in the previous shot migrations appear in the image. Comparison between Figure 5.13 (b) and (d) gives a clear presentation of this.

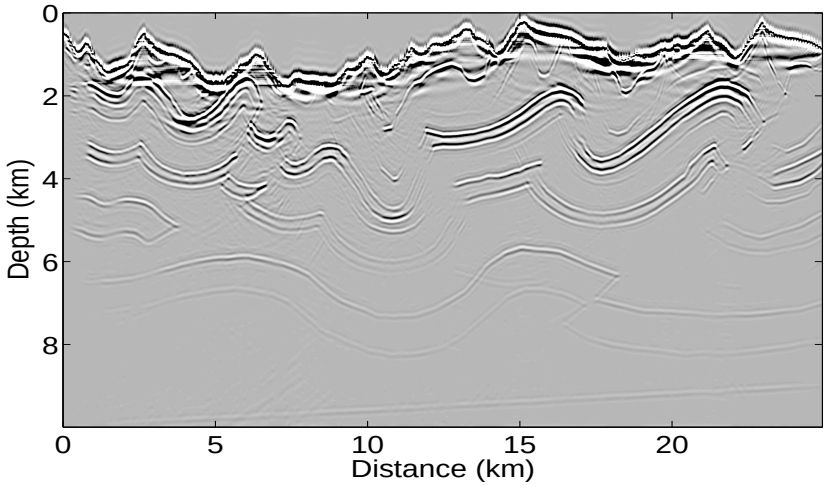


(a)

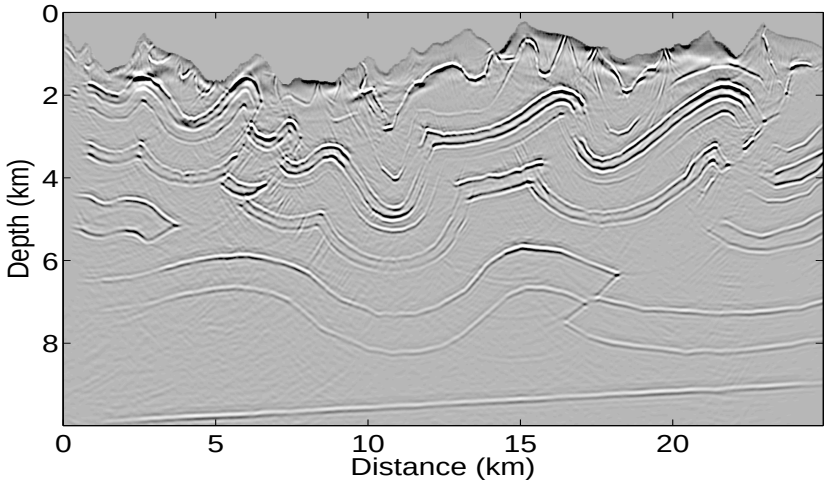


(b)

Figure 5.11: Dip filtering test in depth imaging of Model 94. (a) Depth image after dip filtering applied. (b) Difference taken between the image in (a) and the one in Figure 5.9 (a).



(a)



(b)

Figure 5.12: Depth imaging of Model 94. (a) Image with low-cut frequency filtering. (b) Image from data with direct arrivals muted.

Figure 5.12 (b) shows the result of this test on the whole image of Model 94. We can see that the depth image quality is much better compared to the previous ones (see Figure 5.9 (a) and Figure 5.11 (a)). Detailed structures near surface are clearly imaged though there still exist high amplitudes (black) in the image close to surface.

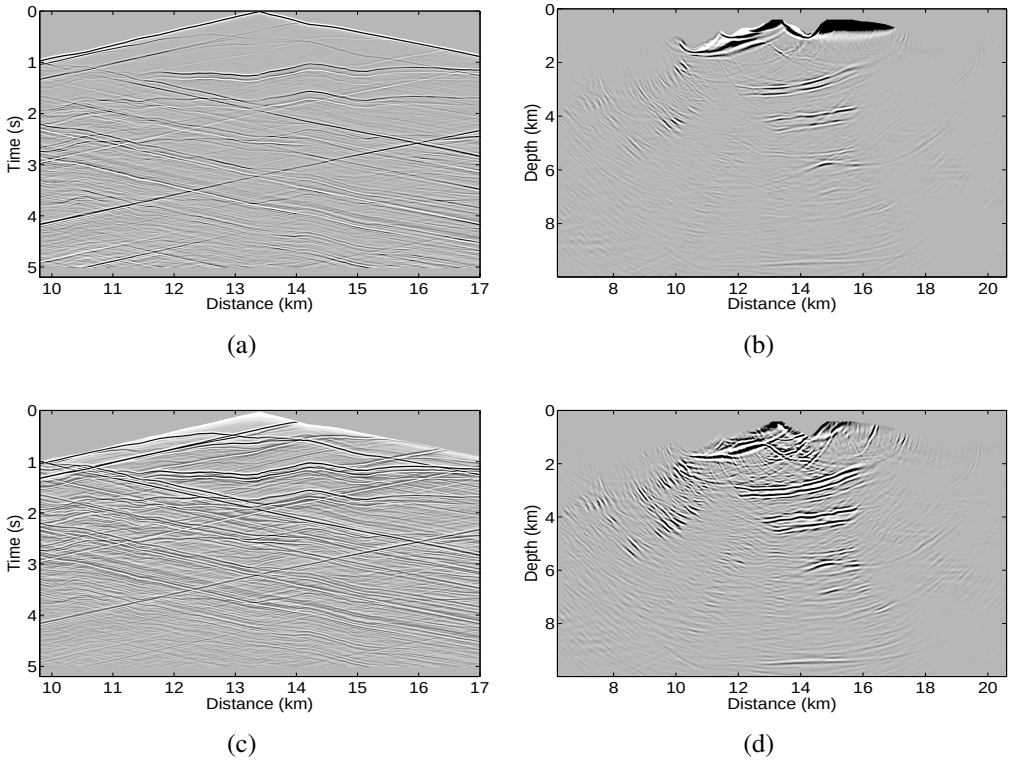


Figure 5.13: Test of direct arrival muting in depth migration of shot record 150. (a) Original shot record. (b) Image of shot record in (a). (c) Shot record 150 with direct arrivals muted. (d) Image of shot record in (c).

5.3.5 Final imaging results for the synthetic data set

Based on the analysis and the tests of depth imaging from Model 94, it is suggested that dip filtering and direct arrival muting should be considered when imaging such data set.

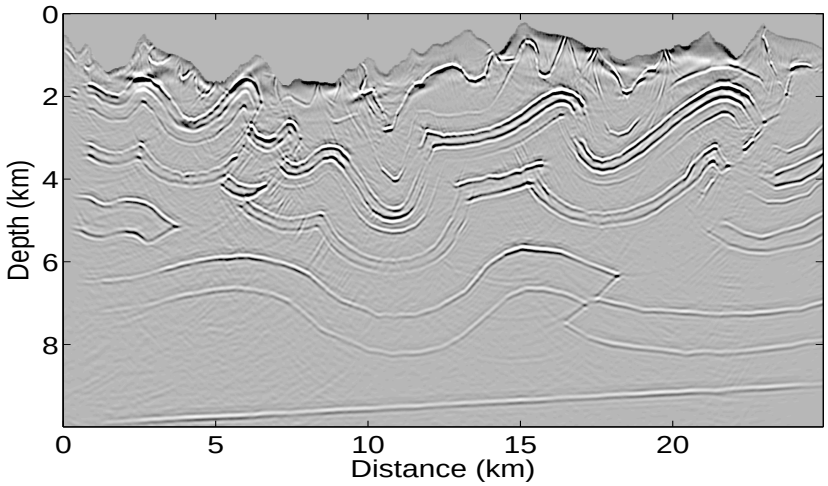


Figure 5.14: Depth images of Model 94: Gabor depth imaging with both direct-arrival muting and dip filtering.

To get the best image, I applied a dip filter in the depth imaging of the Model 94 data set with its direct arrivals muted. Low-frequency filtering is not used because it reduces the resolution of the depth image when removing high amplitudes.

With dip filtering and direct-arrival muting, we have the imaging result shown in Figure 5.14 (a). We can see that dark zones disappear in the image near surface, while the structures in both the shallow and deep parts are still clearly imaged. The most important thing is that the shallow structures near the topography show up. Figure 5.14 shows the best imaging result in this thesis using the topographic Gabor depth imaging method.

Table 5.1: Acquisition parameters of the Grizzly seismic line.

Shot interval	90 m
Group interval	15 m
Line length	11.85 km
Channels/spread	740
Near trace/Far trace offset	30 m/5565 m
Sample rate	2 ms
Energy source	Dynamite (10 kg)

5.4 Imaging 2D Real Data from Topography

5.4.1 Acquisition parameters

The 2D real seismic data set used in this thesis is from an area in northeast British Columbia, Canada (Archer, 2007). Acquisition parameters are listed in Table 5.1 (from Archer (2007)). The surveying line is about 12 km long, which can be seen also from topography profile shown in Figure 5.15.

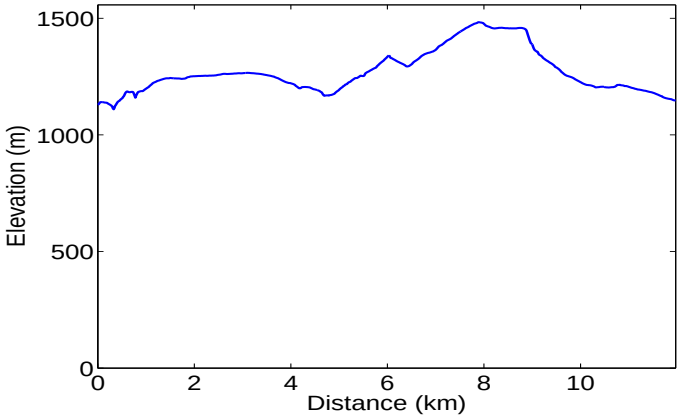


Figure 5.15: Topography profile for the 2D real data acquisition.

The data is a 2D topographic data set, called the Grizzly seismic data set. The maximum elevation difference between the topography extremes is close to 500 meters. Figure 5.15 shows the topography profile along which the seismic data were acquired.

5.4.2 Processing before migration

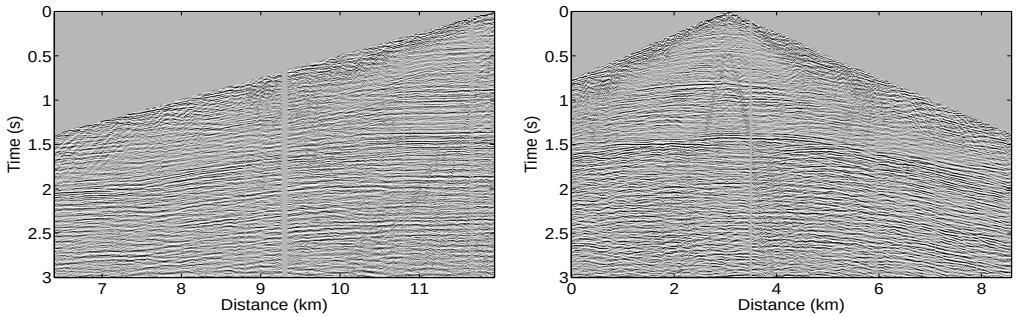
There are 134 shots in this survey and 134 shot records were used in the prestack depth migration. Figure 5.16 shows the first shot record and the 100th shot record from the 2D real data set; the first shot is at the maximum distance along the survey line (see Figure5.16 (a)).

The real data set ready for the depth imaging was pre-processed and stored as shot records in SEGY format. The SEGY shot file was converted into a binary file containing only the data (no header information) and the header (geometry) information was retrieved from Promax and saved in separate files for the depth imaging on a PC with the Gabor depth imaging software. The processes applied to shot records were 'radial filtering', 'refraction statics', 'trace muting', 'deconvolution' and automatic gain control (AGC) [1] (Archer, 2007). Coherent noise like ground roll and air blast had been removed. From shot records shown in Figure 5.16, we also see that the direct arrivals and bad traces have been muted before migration.

5.4.3 Tests with the chosen migration velocities

Velocity models are always vital in depth migration methods. The depth imaging methods used here also requires a good velocity model to do an adequate job. As for this real data set, detailed velocity model building and updating had been done by Archer (2007).

[1] AGC is usually used to remove the geometrical divergence in seismic data caused by wave-front expanding from a point source.



(a) (b)

Figure 5.16: Two shot records from the Grizzly real data set (gaps in the shot records are bad traces removed and left blank). (a) Shot record 1. (b) Shot record 100.

Velocities in this model range from 2500 m/s to about 5000 m/s, but mostly, fall between 4500 and 4800 m/s. Detailed velocity information would give better depth imaging results. However, in this test, I will use constant velocity to migrate the 2D real data set just for the initial imaging test with such a real data set. If more detailed velocity models were available, the Gabor imaging method would do a better job on this real data set from topography.

5.4.4 Imaging results from the real data set

First, let us see some shot migrations. Shot migrations from shot records 1 and 100 are used as examples. Figure 5.17 (a) and (b) show the depth migration results from these shot records using a constant velocity of 4200 m/s.

From these shot images we notice that at high angles migration ‘smiles’ dominate in the image; and they have to be muted before stacking in order to get good images, especially in the shallow part. Figure 5.17 (c) and (d) are the same shot migrations shown

in Figure 5.17 (a) and (b), respectively, except that muting functions are applied to both of them. If the original shot migrations are used to obtain the depth image, it will be degraded by those smiles stacked in (see Figure 5.18 (a) and compare it with Figure 5.18 (b)). With a careful inspection in the image area from $x = 2$ km to $x = 6$ km and depth above $z = 0.8$ km, we can see that the imaging quality in Figure 5.18 (a) is quite bad due to those 'smiles' in individual shot migrations. Similar blurring is seen from $x = 9$ km to $x = 10$ km, and depth $z > 0.9$ km in Figure 5.18 (a).

If all the shot migrations are applied with muting functions, we obtain a better imaging result. In Figure 5.18 (b), we can see improvement in the area between $x = 6$ km and $x = 8$ km, from depth $z = 1$ km $z = 2$ km, compared to the same area in Figure 5.18 (a). The depth imaging result shown in Figure 5.18 (b) is the best that has been obtained by using the Gabor depth imaging method (in this thesis) on the Grizzly seismic data from topography with a constant velocity model.

5.5 Chapter summary

In this chapter, I start with the method of adapting the Gabor imaging method for the flat-datum seismic data to seismic data acquired from topographic surfaces. Then I use the proposed method in a 2D synthetic seismic data set (Model 94). The Gabor imaging method does give reasonably good imaging results at the first try. However, high amplitudes caused by direct-arrival imaging near the topographic surface make the shallow structures almost invisible. The high-amplitude contamination degrades the total depth image of Model 94. However, we can still see that the structures in the middle and deeper part are clearly imaged.

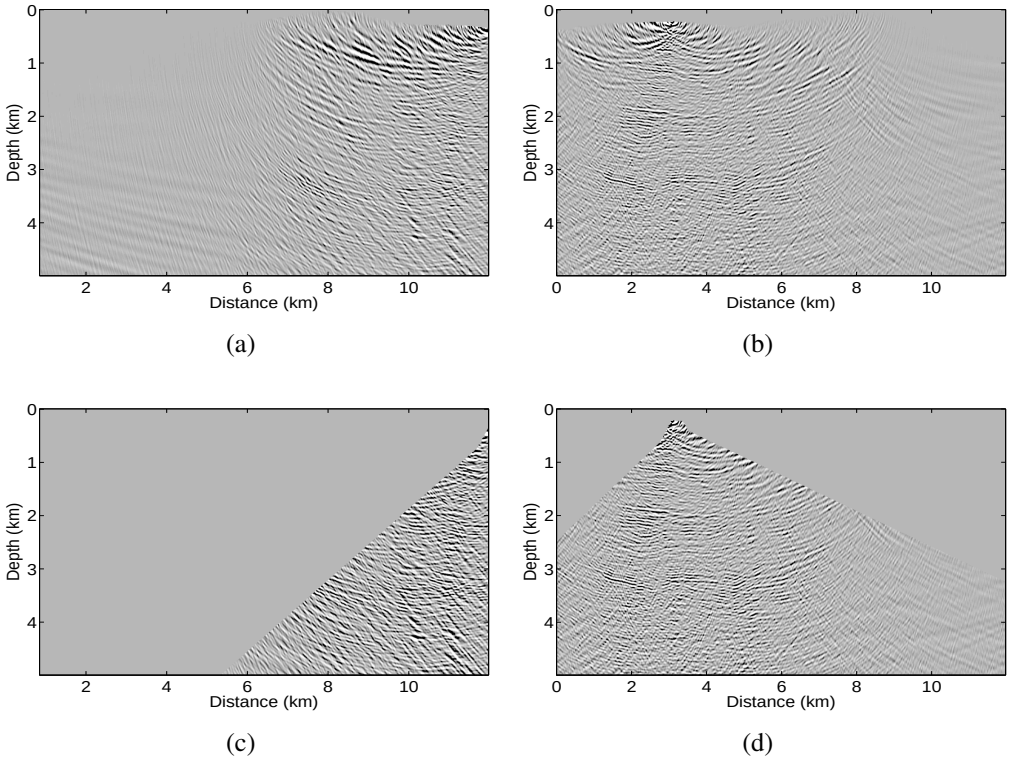
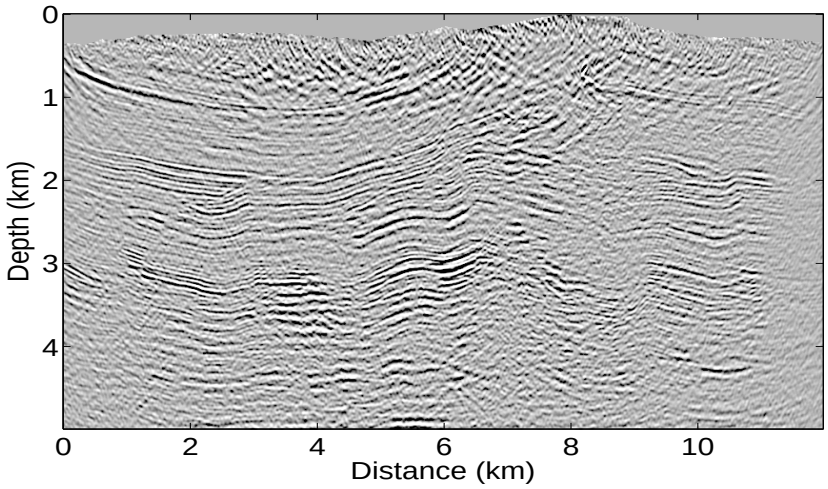
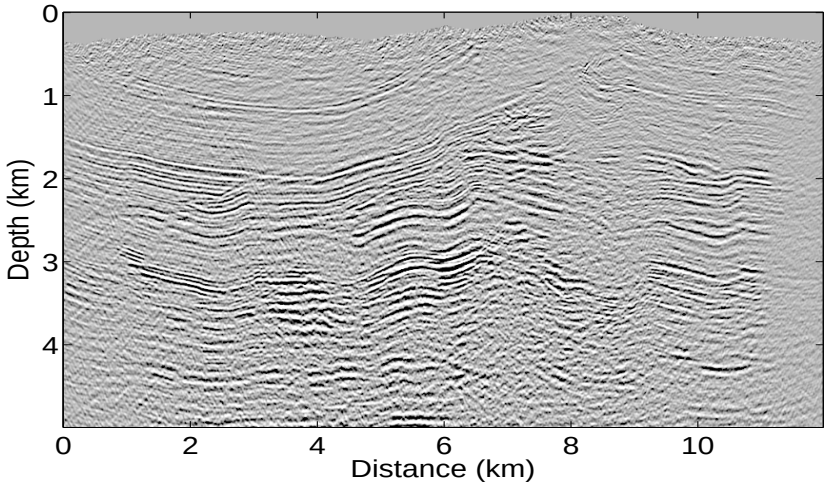


Figure 5.17: Shot migrations from the Grizzly real data set. Shot migrations from (a) shot record 1 and (b) shot record 100, no muting applied in the shot images. (c) Muted shot image corresponding to (a). (d) Muted shot image corresponding to (b).



(a)



(b)

Figure 5.18: Depth imaging results of the Grizzly data set. (a) Imaging without muting in shot migrations. (b) Imaging with muting in shot migrations.

To overcome these imaging difficulties from the topographic model, some suggestions are considered. These are dip filtering, low-frequency filtering and direct-arrival muting. Through the results of these tests, we conclude that direct-arrival muting and dip filtering help to remove high amplitudes in the near surface of depth images of Model 94. Using these two techniques, we have obtained very good images of Model 94.

The Gabor depth imaging method has been tested with a 2D real data set from a topographic surface. Unfortunately, I have no detailed velocity model available. Velocity can be updated by velocity analysis in common imaging gathers (CIG), which is out of the scope of this thesis. However, using the constant velocity (4200 m/s) only, the Gabor depth imaging method works well with the 2D real data set and it gives good imaging results, comparable to the ones from other methods.

In the next chapter I will give theories and primary research results on 3D Gabor depth imaging.

Chapter 6: 3D Gabor Depth Imaging Theory and Initial Imaging Tests

In the previous chapters, I described the Gabor depth imaging method in two dimensions (2D) and used imaging examples to show that the Gabor depth imaging method works very well with 2D seismic data sets from either flat or topographic acquisition surfaces. In the real world, we often work with three-dimensional (3D) seismic data sets, which usually give better images than 2D data sets do. In this chapter, I will develop the Gabor depth imaging method in 3D and present some initial tests for the 3D Gabor depth imaging.

Similarly as I did in Chapter 2 for 2D wavefield extrapolation using the Gabor method, at first I will develop 3D wavefield extrapolation using the phase-shift method; then, I will present a description of 3D Gabor wavefield extrapolation theory.

As we know from 2D Gabor depth imaging theory and examples, the number of partitions is directly related to the number of spatial Fourier transforms. We should keep this number small whenever possible while maintaining a desired accuracy, which is critical in 3D Gabor depth imaging in that data sets involved in 3D depth imaging are usually quite large compared to those used in 2D depth imaging. The number of partitions can be controlled using a 2D adaptive partitioning method based on a partitioning criterion and a velocity slice from a 3D velocity volume. The 2D adaptive partitioning algorithm is extended from the one using the lateral position error as a criterion introduced in Chapter 3 for adaptive partitioning in a 1D velocity profile. Partitioning examples using the 2D adaptive partitioning algorithm will be shown.

Impulse response tests are used to show that 3D Gabor extrapolator works as expected.

These initial tests are critical steps in the 3D Gabor depth imaging method. In the next section, I will describe wavefield extrapolation using the phase-shift method in 3D.

6.1 Wavefield extrapolation with the phase shift in 3D

Fourier transforms in 1D have been defined in Chapter 2. 2D Fourier transforms used in this thesis are only involved in the spatial coordinates.

The forward Fourier transform is

$$F(k_x, k_y) = \int_{\mathbb{R}^2} f(x, y) \exp\left[i(k_x x + k_y y)\right] \mathrm{d}x\mathrm{d}y, \tag{6.1}$$

and the inverse Fourier transform is

$$f(x, y) = \frac{1}{4\pi^2} \int_{\mathbb{R}^2} F(k_x, k_y) \exp\left[-i(k_x x + k_y y)\right] \mathrm{d}k_x\mathrm{d}k_y, \tag{6.2}$$

where x and y are the spatial coordinates, k_x and k_y are the wavenumber coordinates, F is the spatial Fourier transform of f, and $\mathbb{R}^2$ denotes the real plane in the spatial (x-y) or wavenumber (k_x-k_y) domain.

In this development, I start with a 3D scalar (acoustic) wave equation (e.g. Berkhout, 1981), which is written as

$$\frac{\partial^2 \psi}{\partial z^2} + \frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} - \frac{1}{v^2} \frac{\partial^2 \psi}{\partial t^2} = 0, \tag{6.3}$$

where x, y and z are the spatial coordinates, t is the time coordinate, $\psi = \psi(x, y, z, t)$ is the pressure in the spatial-time domain, and v is wave speed. Equation (6.3) is the same partial differential equation (PDE) as the one introduced in Chapter 2 except for an additional dimension y.

We face the same problem as we had in 2D (see section 2.2.3 in Chapter 2). That is, we try to obtain the wavefield at depth z ($\psi(x, y, z, t)$) provided the wavefield at the depth $z = 0$ ($\psi(x, y, 0, t)$). To address such a problem, we need to solve equation (6.3). The method to solve this equation is very similar to the one introduced in Chapter 2 for the PDE's in 2D space.

Using 1D forward temporal Fourier transforms and 2D forward spatial Fourier transforms (defined in equation (2.1) in Chapter 2 and equation (6.1), respectively) on both sides of equation (6.3) yields

$$\frac{\partial^2 \hat{\Psi}}{\partial z^2} + k_z^2 \hat{\Psi} = 0. \tag{6.4}$$

Variable k_z in equation (6.4) is a component of the total wavenumber vector with a magnitude of k, and defined as

$$k_z = \begin{cases} \sqrt{k^2 - (k_x^2 + k_y^2)}, & k^2 \geqslant k_x^2 + k_y^2 \\ i\sqrt{(k_x^2 + k_y^2) - k^2}, & k^2 < k_x^2 + k_y^2, \end{cases} \tag{6.5}$$

where

$$k = \frac{\omega}{v}. \tag{6.6}$$

Solution to equation (6.4) is (assuming upward traveling waves)

$$\hat{\Psi}(k_x, k_y, z, \omega) = \hat{\Psi}(k_x, k_y, 0, \omega) \exp(ik_z z), \tag{6.7}$$

where $\hat{\Psi}(k_x, k_y, 0, \omega)$ denotes (Dirichlet) boundary conditions, usually known as seismic data recorded in an area on the surface and transformed into the frequency-wavenumber domain. Writing $\Psi(k_x, k_y, 0, \omega)$ in an explicit formula, we have

$$\hat{\Psi}(k_x, k_y, 0, \omega) = \int_{\mathbb{R}^3} \psi(x, y, 0, t) \exp\left[i(k_x x + k_y y - \omega t)\right] \mathrm{d}t\mathrm{d}x\mathrm{d}y, \tag{6.8}$$

where $\mathbb{R}^3$ indicates a multiple integral with respect to t, x and y.

The spatial phase-shift term, $\exp(ik_z z)$, in (6.7) is defined as a 3D wavefield extrapolator using a constant velocity, which can be written as

$$\hat{W}(k_x, k_y, z, \omega) = \exp\left[ik_z(k_x, k_y, \omega)z\right]. \tag{6.9}$$

Using (6.9) in equation (6.7), we have

$$\hat{\Psi}(k_x, k_y, z, \omega) = \hat{\Psi}(k_x, k_y, 0, \omega)\hat{W}(k_x, k_y, z, \omega). \tag{6.10}$$

To get the wavefield in the spatial-frequency domain at depth z, we use 2D inverse spatial Fourier transforms in equation (6.10) and have

$$\Psi(x, y, z, \omega) = \frac{1}{4\pi^2}\int_{\mathbb{R}^2} \hat{\Psi}(k_x, k_y, 0, \omega)\hat{W}(k_x, k_y, z, \omega)\exp\left[-i(k_x x + k_y y)\right]\mathrm{d}k_x\mathrm{d}k_y. \tag{6.11}$$

Equation (6.11) is the formula for seismic wavefield extrapolation in a 3D homogeneous medium.

Similarly as shown in Chapter 2, solution to PDE (6.3) in a 2D slab (constant velocity) with a thickness of Δz can be written as

$$\Psi(x, y, \Delta z, \omega) = \frac{1}{4\pi^2}\int_{\mathbb{R}^2} \hat{\Psi}(k_x, k_y, 0, \omega)\hat{W}(k_x, k_y, \Delta z, \omega)\exp\left[-i(k_x x + k_y y)\right]\mathrm{d}k_x\mathrm{d}k_y. \tag{6.12}$$

In a 2D slab with lateral velocity variations, i.e., $v = v(x, y)$, the LHA (or GPSPI) method can be used to solve such a wavefield extrapolation problem. In the next section, I will present the 3D wavefield extrapolation using the Gabor transform, which is an approximation to the LHA formula in 3D media.

6.2 3D Gabor extrapolation theory

The Gabor extrapolator has been developed as an approximation of the LHA (GPSPI) extrapolator in 2D in Chapter 2. The LHA extrapolation formula in 3D can be written as (after Margrave et al., 2004)

$$\begin{aligned}\Psi\left(x,y,\Delta z,\omega\right) &= \frac{1}{4\pi^2}\int_{\mathbb{R}^2}\hat{\Psi}\left(k_x,k_y,0,\omega\right)\hat{W}\left(x,y,k_x,k_y,\Delta z,\omega\right)\\ &\qquad\times\exp\left[-i(k_x x+k_y y)\right]\mathrm{d}k_x\mathrm{d}k_y,\end{aligned}\tag{6.13}$$

where all variables are defined similarly as in equations in the previous section except for the 2D velocity slice (from a 3D volume), varying in the x-y plane (i.e., $v = v(x,y)$). Equation (6.13) extrapolates wavefields from depth 0 down to depth Δz in the frequency-wavenumber domain and transforms them into the frequency-space domain using the 2D inverse spatial Fourier transform.

Equation (6.13) is very similar to equation (6.12). However, they are different from each other in the definition of $\hat{W}$, the 3D wavefield extrapolator. The wavefield extrapolator in (6.12) is related to a constant velocity. In the LHA formula (equation (6.13)), $\hat{W}$ is related to a locally constant velocity at an output position (x,y) (see also section 2.4 of Chapter 2). So formula (6.13) is still a reasonable approximate solution to a PDE in a medium of non-constant velocity. The 2D slab, extending from 0 to Δz, has to be such a thin one that velocity variations in the vertical direction (z) are small enough to be treated as 'constant' or approximately 'constant', i.e., $dv(z)/dz \approx 0$.

The LHA wavefield extrapolator, $\hat{W}$ in equation (6.13), is a spatial phase shift term and defined as

$$\hat{W}\left[x,y,k_x,k_y,\Delta z,\omega\right] = \exp\left[ik_z(x,y,k_x,k_y,\omega)\Delta z\right],\tag{6.14}$$

where k_z, orthogonal to the k_x-k_y plane, is a component of the total wavenumber k. k_z is explicitly defined as

$$k_z(x, y, k_x, k_y, \omega) = \begin{cases} \sqrt{k^2(x, y, \omega) - (k_x^2 + k_y^2)}, & k^2(x, y, \omega) \geqslant k_x^2 + k_y^2 \\ i\sqrt{(k_x^2 + k_y^2) - k^2(x, y, \omega)}, & k^2(x, y, \omega) < k_x^2 + k_y^2, \end{cases} \tag{6.15}$$

and k in this case is dependent on x and y and can be written as

$$k(x, y, \omega) = \frac{\omega}{v(x, y)}. \tag{6.16}$$

The approximation to the LHA extrapolator is very similar to the one described in 2D in Chapter 2. In approximation of the LHA extrapolator in 3D with the Gabor method, we use

$$\hat{W}(x, y, k_x, k_y, \Delta z, \omega) \approx \sum_{j \in \mathbb{Z}} \Omega_j(x, y) S_j(x, y) \hat{W}_j \left(k_x, k_y, \Delta z, \omega\right), \tag{6.17}$$

where Ω_j is a set of 2D partitions forming a 2D partition of unity (POU), meaning they sum to unity.

$$\sum_{j \in \mathbb{Z}} \Omega_j(x, y) = 1. \tag{6.18}$$

Reference velocity v_j is defined in a partition $\Omega_j(x, y)$ as

$$v_j = \frac{\int_{\mathbb{R}^2} \Omega_j(x, y) v(x, y) \mathrm{d}x\mathrm{d}y}{\int_{\mathbb{R}^2} \Omega_j(x, y) \mathrm{d}x\mathrm{d}y}. \tag{6.19}$$

The split-step Fourier operator (Stoffa et al., 1990) formerly defined in Chapter 2 is extended to 2D, correcting residual phase shift from velocity fluctuations, which is written as

$$S_j(x, y) = \exp\left[i\omega\Delta z \left(\frac{1}{v(x, y)} - \frac{1}{v_j}\right)\right]. \tag{6.20}$$

Total wavenumbers k_j $(j \in \mathbb{Z})$ corresponding to partitions (Ω_j) are defined as

$$k_j = \frac{\omega}{v_j}. \tag{6.21}$$

Using the approximation (6.17) in (6.13) yields

$$\Psi(x,y,\Delta z,\omega) \approx \sum_{j\in\mathbb{Z}} \Omega_j(x,y) S_j(x,y) \int_{\mathbb{R}^2} \hat{\Psi}(k_x,k_y,0,\omega)\hat{W}_j\left(k_x,k_y,\Delta z,\omega\right) \times \exp\left[-i(k_x x + k_y y)\right]\mathrm{d}k_x\mathrm{d}k_y. \quad (6.22)$$

Equation (6.22) is the approximation formula to the LHA extrapolation, also called the 3D Gabor extrapolation formula. As known from the 2D Gabor wavefield extrapolation theory described in Chapter 2, this is one of the extreme cases when $p = 0$, simulating the PSPI method (see equation (2.42)).

Equation (6.22) is the formula used in the following sections for the 3D Gabor wavefield extrapolation.

To make an efficient 3D Gabor extrapolation, we need an adaptive partitioning algorithm to control the number of partitions. In the next section, I will describe a 2D adaptive partitioning algorithm using the lateral-position-error criterion.

6.3 2D adaptive partitioning algorithm for 3D Gabor extrapolation

The number of partitions, $\Omega_j(x,y)\ (j \in \mathbb{Z})$, is key to an efficient 3D Gabor depth imaging because it determines the number of 2D spatial Fourier transforms. This number can be optimized by using adaptive partitioning algorithms given a velocity model and an accracy criterion. In such a way, the number of reference velocities corresponding to partitions can also be determined. In this section, I introduce the method to determine 2D adaptive partitions for 3D Gabor wavefield extrapolation. Lateral position errors both in the x and y directions are involved in the adaptive partitioning.

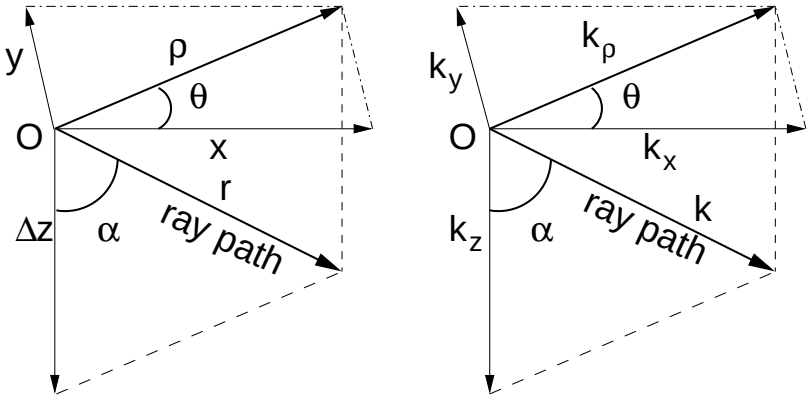
As was developed in the 1D adaptive partitioning method using the lateral-position-error criterion, we begin with a ray emanating (at angle θ in azimuth) at point $(x,y,z) =$

$(0, 0, 0)$ (see Figure 6.1 (a)). We have a geometry relation in the spatial domain as

$$\rho = \Delta z \tan \alpha, \tag{6.23}$$

where α is the angle between the vertical direction z and the ray path, which is called the *design angle* hereafter. The distance from $(0, 0)$ to (x, y) in the horizontal x-y plane at $z = 0$ is

$$\rho = \sqrt{x^2 + y^2}\,. \tag{6.24}$$



(a) (b)

Figure 6.1: Geometry of a local ray path in the (a) spatial domain and (b) wavenumber domain.

Total distance in 3D from $(0, 0, 0)$ to $(x, y, \Delta z)$ is

$$r = \sqrt{\rho^2 + (\Delta z)^2}\,. \tag{6.25}$$

Similarly, in the wavenumber domain, we have geometry relation

$$k_\rho = \sqrt{k_x^2 + k_y^2}\,, \tag{6.26}$$

where k_ρ is composed by the two components of the total wavenumber k in the horizontal k_x-k_y plane. Also from physical definitions and the geometry (see Figure 6.1 (b)), we have

$$k = \frac{k_\rho}{\sin\alpha} = \frac{\omega}{v}. \tag{6.27}$$

If the lateral position shift in x and y (δx and δy) are small, the total lateral position shift $\delta\rho$ in the x-y plane will also be small; and the small position shift in ρ results in a small perturbation in velocity (δv). If these conditions suffice, the differential calculus expression $\delta\rho = \frac{\partial\rho}{\partial v}\delta v$ exists. Taking differentials in equations (6.23) and (6.27) with respect to v gives

$$\delta\rho = \Delta z \sec^2\alpha \frac{\partial\alpha}{\partial v}\delta v, \tag{6.28}$$

and

$$\frac{\partial\alpha}{\partial v} = \frac{k_\rho}{\omega}\sec\alpha, \tag{6.29}$$

respectively.

A similar development for the relationship between velocity (2D) perturbation δv and the lateral position error $\delta\rho$ (involving position errors from x and y) can be obtained as the one that we used in Chapter 2 for 1D velocity profiles. Using equations (6.28) and (6.29) and rearranging, we have

$$\delta v = \frac{\cos^3\alpha}{\sin\alpha}\frac{v}{\Delta z}\delta\rho. \tag{6.30}$$

As for the design angle α, I also use 45° as chosen for the 1D adaptive partitioning using the lateral-position-error criterion. Design angles may need to be tested and decided based on which one is the best for the depth imaging with a 3D data set.

If the design angle α, Δz and v are known, we can use the lateral position error $\delta\rho$ as the criterion to define a complete set of reference velocities with equation (6.30). The lateral position error $\delta\rho$ is related to accuracy of the 3D Gabor depth image.

Comparing equation (6.30) to equation (3.18) in Chapter 3, we know that these two formulae are exactly the same. Hence, procedures of creating a reference velocity set are very similar to those described in Chapter 3.

Presenting these formulae used to create the reference velocity (for 2D) set, we have

$$v_{2n} = \left(\frac{2+a}{2-a}\right)^n v_{1,} \quad n = 1, 2, \ ... \tag{6.31}$$

and

$$v_{2n+1} = \left(\frac{2-a}{2+a}\right)^n v_1, \quad n = 1, 2, \ ..., \tag{6.32}$$

where $a = \cos^3 \alpha \sin^{-1} \alpha\delta\rho/\Delta z$.

Figure 6.2 (a) shows a 2D horizontal slice from a 3D velocity model to demonstrate the 2D adaptive partitioning algorithm. Partitioning examples related to this model will be shown in the following section. Figure 6.2 (b) is a 3D view of the velocity model shown in Figure 6.2 (a).

To start the adaptive partitioning, we need to find a set of reference velocities given a lateral-position-error criterion and a 2D velocity slice. The first reference velocity v_1 is chosen as the most frequently occurring velocity value. This can be done by searching the histogram of the 2D velocity slice (see Figure 6.2 (c)). Once v_1 is determined, equations (6.31) and (6.32) can be used to determine a complete set of reference velocities (see Figure 6.2 (c) and (d)).

In a 2D partitioning, each reference velocity is represented as a 2D plane with a constant value and it goes through the 2D velocity model (shown in 3D-view as a 'topography' corresponding to velocity values). These planes (see Figure 6.2 (d)) can be used as scales

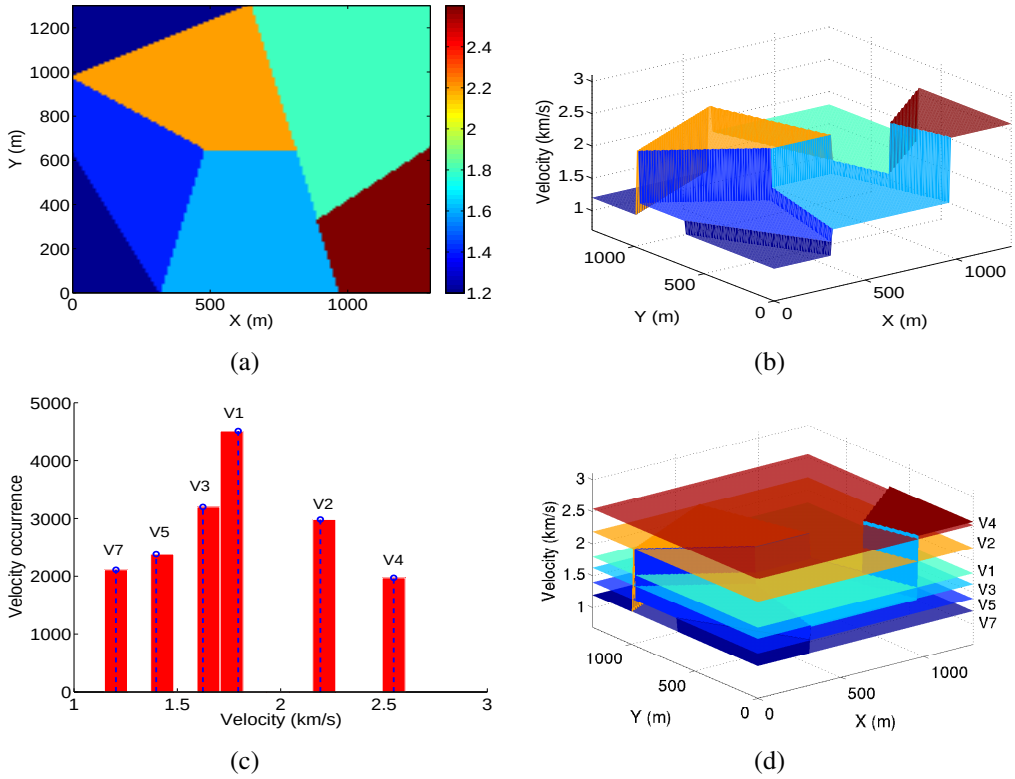


Figure 6.2: Reference velocities determined by the 2D adaptive partitioning algorithm using the lateral position error. (a) A 2D velocity slice. The colour bar on the right codes velocities in km/s. (b) A 3D view of the velocity model shown in (a). (c) Statistics of velocities from the model in (a) with reference velocities (blue-dashed lines) superimposed. (d) A 3D view of the velocity with 2D reference planes (velocities) superimposed.

to create the 2D indicator function $I_j(x, y)$, which can be defined as

$$I_j\,(x, y) = \begin{cases} 1, & |v(x, y) - v_j\,| = \text{minimum}, (\text{over } j) \\ 0, & \text{otherwise}, \end{cases} \tag{6.33}$$

where $v(x, y)$ denotes exact velocities and $v_j\,(j = 1, 2, ..., n)$ are reference velocities. Equation (6.33) says that the indicator function $I_j(x, y)$ of reference velocity v_j is created by comparing v_j to $v(x, y)$. If a velocity value $v(x, y)$ at a spatial position (x, y) is the closest one to v_j, the indicator function at this position is assigned a "1", otherwise, a "0". There is a possibility that $v(x, y)$ at a position is mid-way between two reference velocities (see horizontal planes in Figure 6.2 (d)). If so, it can be assigned to either side. However, to get rid of the ambiguity, we can always assign $v(x, y)$ to the side of the smaller one of two reference velocities.

A normalized convolution of the indicator function $I_j(x, y)$ and a 2D atomic window (see Figure 6.3) is used to create 2D partitions (forming a 2D POU). These partitions correspond to reference velocities in the Gabor wavefield extrapolation. This is given as

$$\Omega_j(x, y) = (I_j * \Theta)\,(x, y), \tag{6.34}$$

where Θ is the 2D atomic window, $*$ stands for the normalized convolution. The 2D atomic window can be any type of 2D bump function, such as the 2D Gaussian, with localizing property.

In the next section, I will demonstrate some examples of the 2D adaptive partitioning using the lateral-position-error criterion.

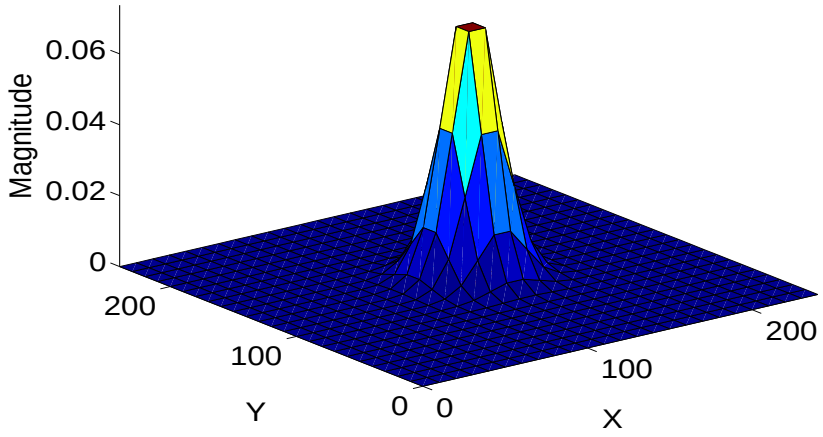
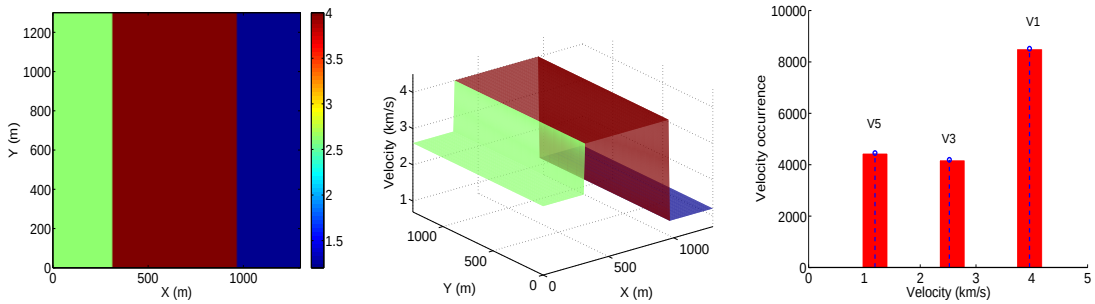


Figure 6.3: 2D atomic window (Gaussian).

6.4 2D adaptive partitioning examples

The first partitioning example is a simple 2D bump velocity model. The velocity only varies in the x direction (see Figure 6.4 (a)). The velocity model starts at a plateau with an 'altitude' (constant velocity) of 2.6 km/s and then climbs up to a highest plateau of 4.0 km/s and finally goes down to a level of 1.2 km/s (see Figure 6.4 (b)). Once the velocity model is known, we can use statistics of velocities in the model, which are shown as the histogram of the model. The most frequently occurring velocity can be easily found in the histogram and it is set as the first reference velocity (v_1), from where searching for other reference velocities is started using equations (6.30), (6.31) and (6.32).

When reference velocities are found, indicator functions corresponding to them are created using equation (6.33). Using these indicator functions and the 2D atomic window (see Figure 6.3), we can build the 2D adaptive partitions for the model. The partitions in



(a) (b) (c)

Figure 6.4: A set of reference velocities created by the 2D adaptive partitioning algorithm using a lateral position error of 3 m. (a) A top view of the 2D bump velocity model. Velocities are coded in km/s. (b) A 3D view of the 2D bump velocity slice. (c) Statistics of the velocities in the velocity model shown in (a) or (b) with references velocities superimposed.

this example are created by the 2D adaptive partitioning algorithm using a lateral position error of 3 m. These partitions are shown in Figure 6.5 ((a)-(c)). Figure 6.5 (d) shows all the partitions superimposed together. Figure 6.5 (e) shows the summation of all the partitions and we can see that they sum to 1, giving a 2D POU. Figure 6.5 (f) shows a profile in x direction at y = 650 m through all the partitions (see Figure 6.5 (d)). We can see that this slice also suffices to define a 1D POU (see the black solid line on top of the partitions).

The second model is also a bump velocity model with a high plateau in the middle (4.0 km/s), but the plateaus on both sides share the same velocity value (2.6 km/s). Also, random fluctuations in velocity have been added into the model, making the velocity model more complex compared to the first one. Figure 6.6 (a) shows the plane-view of the second model. A 3D view of the model is shown in Figure 6.6 (b). The reference velocity set given by the partitioning algorithm using a lateral position error of 3 m is shown in Figure 6.6 (c). In this case, four reference velocities have been selected by the 2D adaptive partitioning

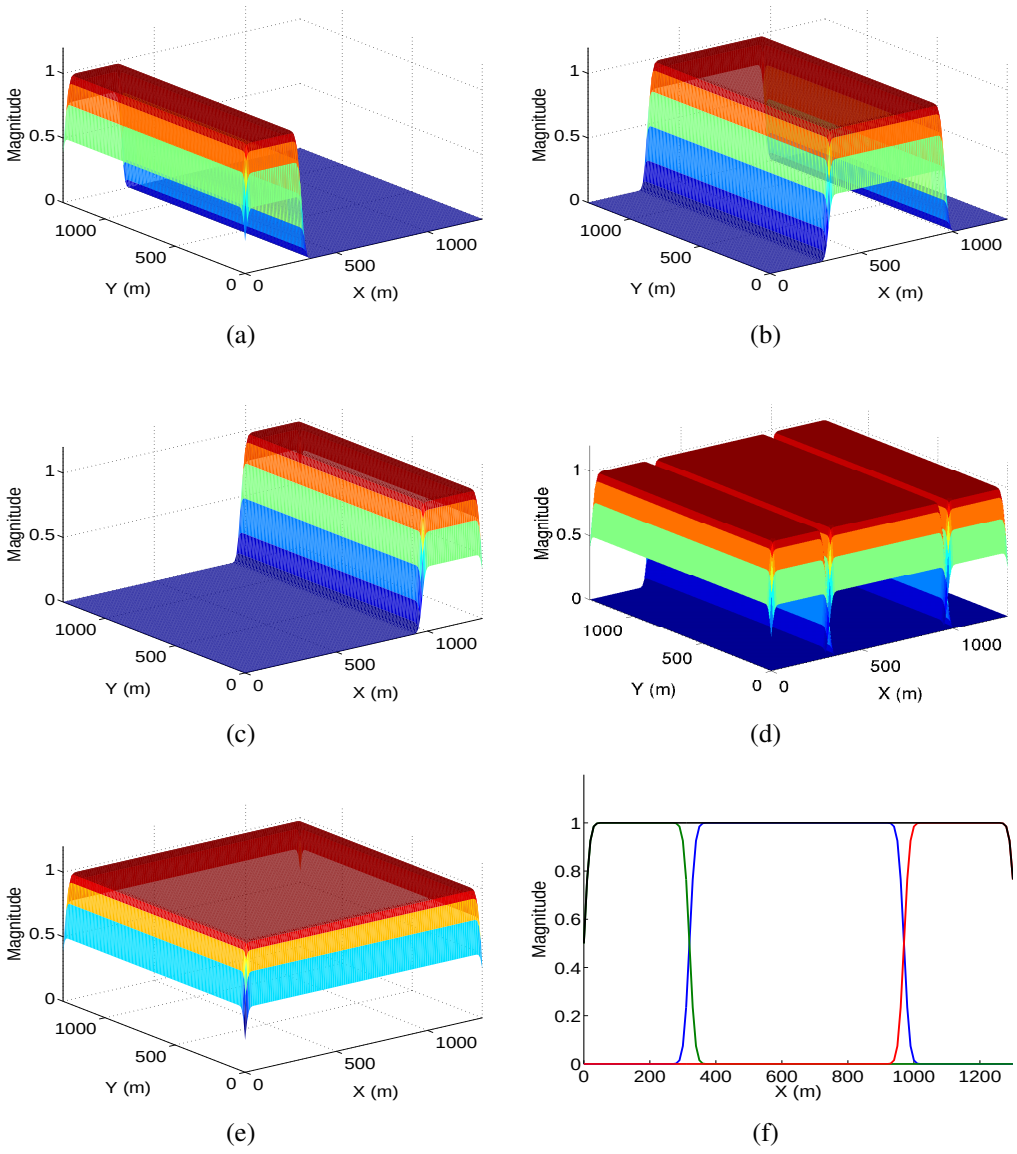
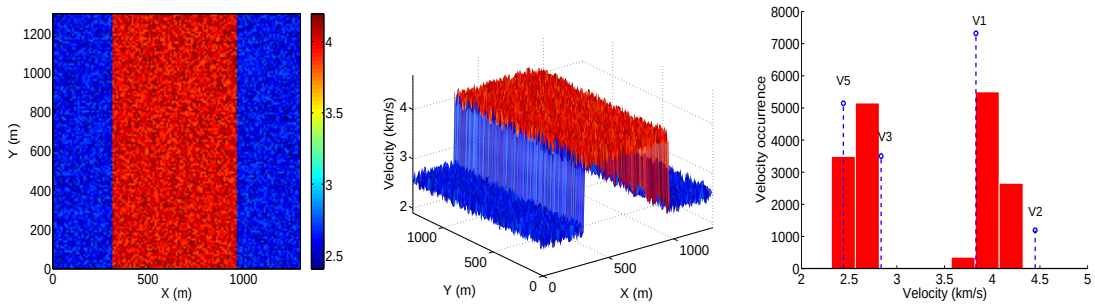


Figure 6.5: Adaptive partitioning on a bump velocity model using a lateral position error of 3 m. (a) The first partition related to the left plateau (see Figure 6.4 (b)). (b) The second partition related to the plateau in the middle. (c) The third partition related to the right plateau. (d) All partitions superimposed. (e) 2D POU. (f) 1D POU in the x direction.

algorithm. If there is no random velocity fluctuations in those plateaus, the partitioning would select two reference velocities corresponding to two distinct velocity values in the model.



(a) (b) (c)

Figure 6.6: Adaptive partitioning on a bump velocity model with random fluctuations. (a) A 2D view of the velocity model. (b) A 3D view of the velocity model. (c) Velocity statistics with the reference velocities selected by the partitioning algorithm superimposed.

The adaptive partitions given by the 2D partitioning algorithm are shown in Figure 6.7. With random fluctuations in the velocity model, the 2D adaptive partitioning algorithm gives partitions with complicated shapes, which means that the algorithm is suitable for partitioning velocity models with complex structures. Superimposing these partitions together, we have a picture shown in Figure 6.7 (e), from which we can not tell whether 2D or 1D POU is satisfied. However, summing the partitions together, we have a 2D POU shown in Figure 6.7 (f).

Taking two vertical profiles in the x and y directions, respective, we have two 1D POU's shown in Figure 6.8 (a) and (b).

The third adaptive partitioning example is set up with a few velocity chunks (shown in various colours in Figure 6.2 (a) and Figure 6.9 (a)). The 2D partitioning algorithm is

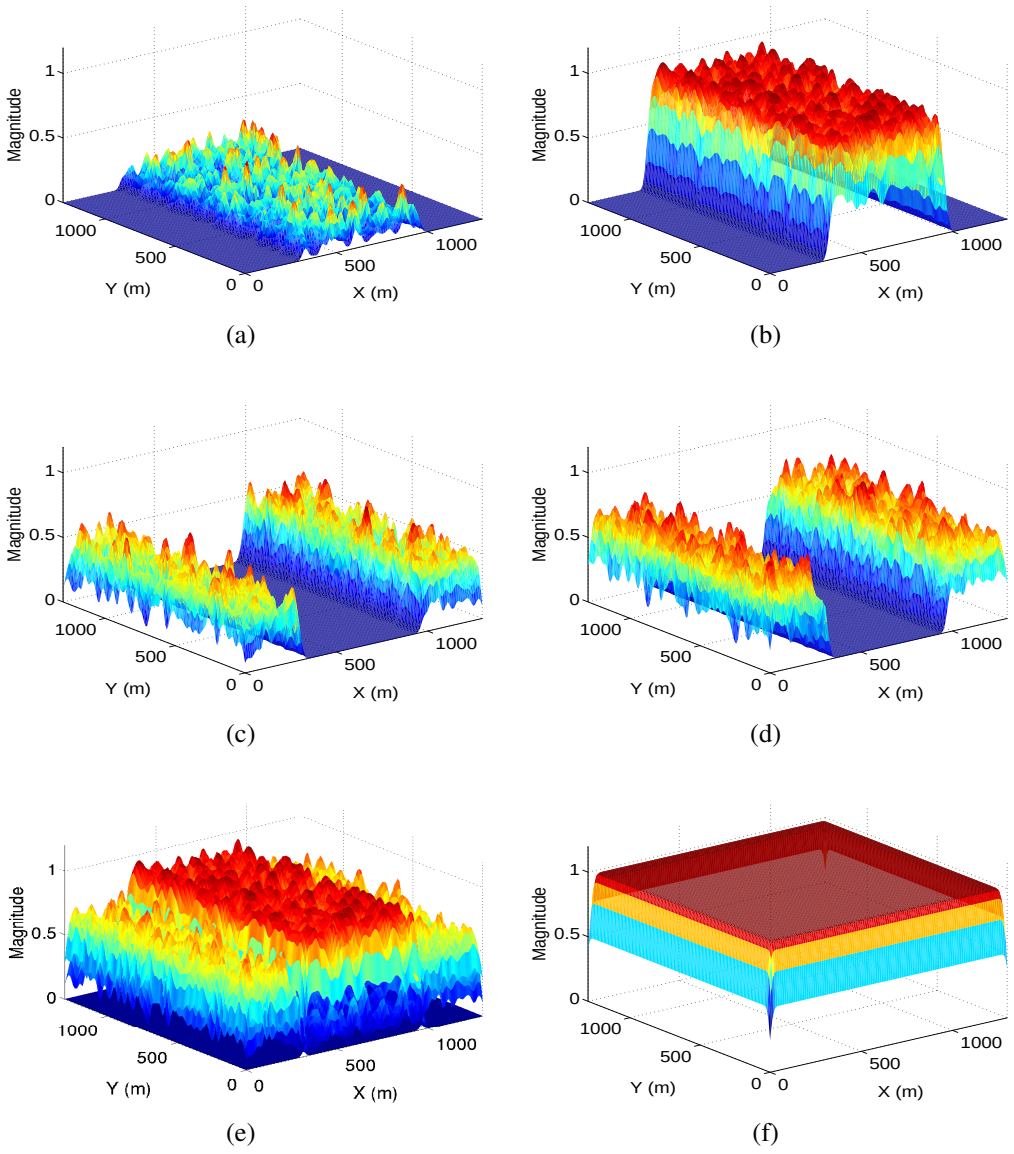
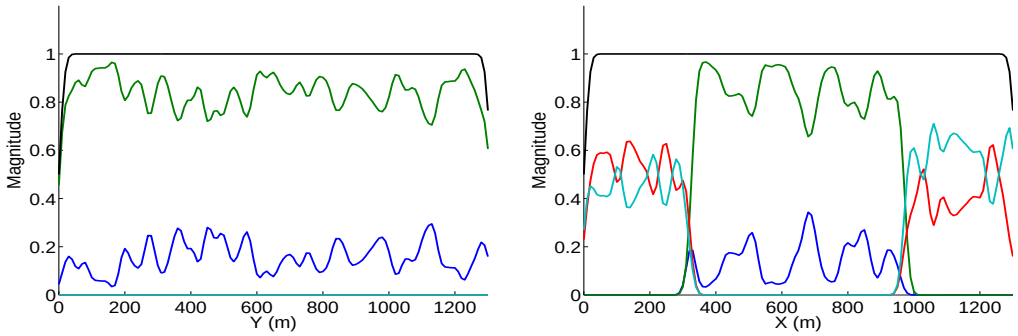


Figure 6.7: Adaptive partitioning on a bump velocity model with random fluctuations using a lateral position error of 3 m. (a) The first partition. (b) The second partition. (c) The third partition. (d) The fourth partition. (e) Partitions superimposed. (f) 2D POU from summing all partitions.



(a) (b)

Figure 6.8: 1D POU's. (a) in the y direction at x = 650 m. (b) 1D POU in the x direction at y = 300 m.

expected to give partitions corresponding to those velocity chunks, each with a distinct velocity value. This velocity model has been introduced in section 6.3 to show how to select reference velocities in the partitioning algorithm. For the 3D view of the velocity model and the reference velocities, see Figure 6.2 in section 6.3.

The partitions created by the 2D partitioning algorithm using a lateral position error of 1 m are shown in Figure 6.9 (b) to (g). Two triangular chunks (in dark blue) share the same velocity value. Other chunks have different velocity values from each other. As expected, the two triangular chunks are related to a partition (see Figure 6.9 (b)) with two maxima (or plateaus). Other chunks are associated with partitions with single maximum (or plateau) (see Figure 6.9, from (c) to (g)).

Superimposing these partitions together, we can see there are 'gaps' between them (see Figure 6.9 (h)), which are related to boundaries between the velocity chunks. Summing the partitions gives a 2D POU shown in Figure 6.9 (i).

Two vertical profiles are taken in the x and y directions and shown in Figure 6.10 (a)

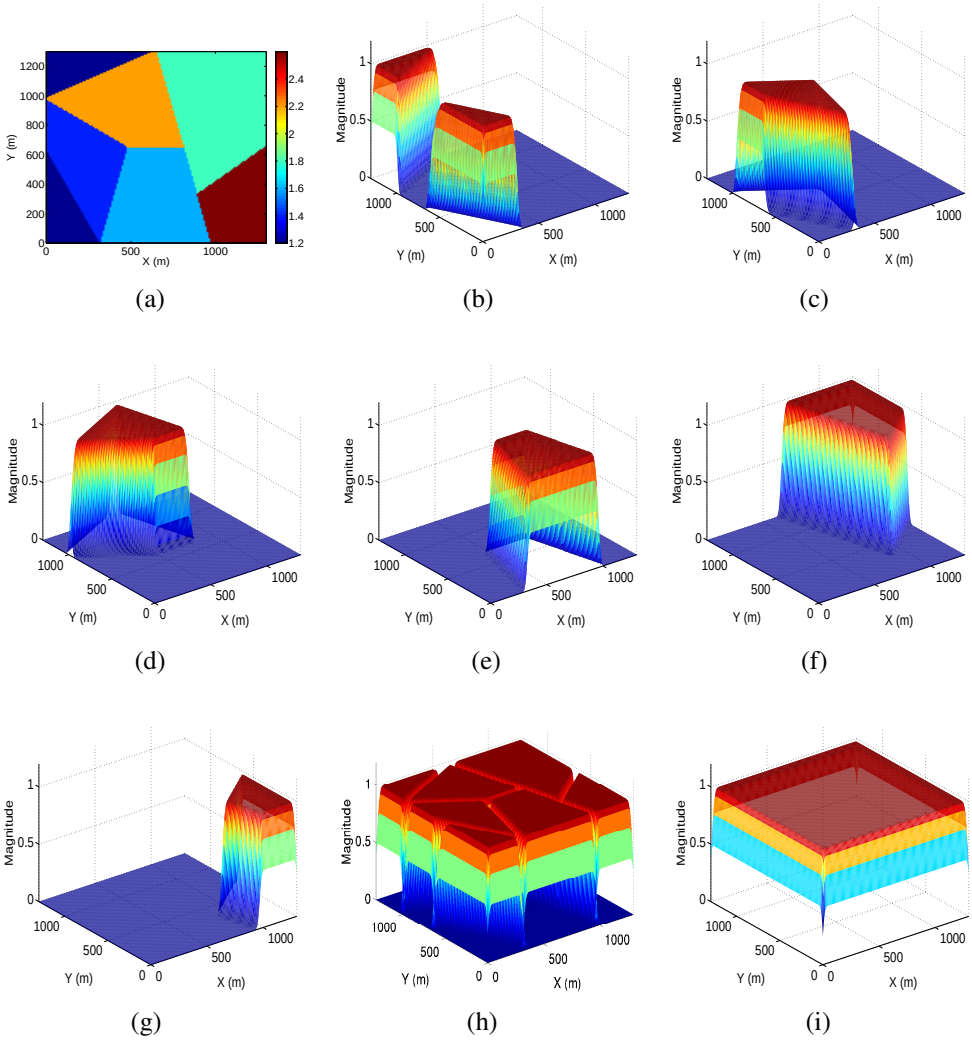
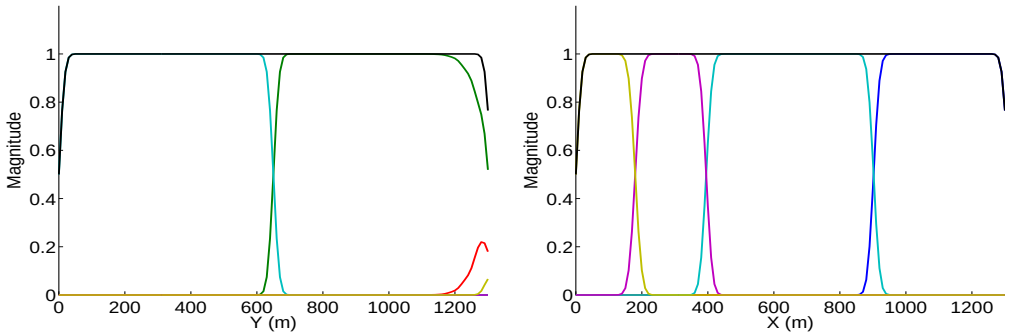


Figure 6.9: Adaptive partitioning on the mosaic velocity model using a lateral position error of 1 m. (a) The mosaic velocity model. (b) The first partition: triangles in dark blue in (a). (c) The second partition: sky blue chunk adjacent to the lower triangle in (a). (d) The third partition: yellow chunk in (a). (e) The fourth partition: light-blue chunk at the bottom of (a). (f) The fifth partition: light-green chunk in the upper right corner of (a). (g) The sixth partition: dark-red chunk in (a). (h) All the partitions superimposed together. (i) 2D POU from summing of the 6 partitions.



(a) (b)

Figure 6.10: 1D POU from the 2D partitioning of the mosaic velocity model. (a) 1D POU in the y direction at x = 650 m. (b) 1D POU in the x direction at y = 300 m.

and (b). We can see from these profiles that POU also holds in the x and y directions.

Generating 2D adaptive partitions is a key step to an efficient 3D Gabor depth imaging. The 2D adaptive partitioning examples have shown an effective partitioning method for 3D Gabor depth migrations. Another step is to test the 3D Gabor extrapolator. In the next section, I will show impulse responses using the 3D Gabor extrapolator.

6.5 3D impulse response

The dimensions of the 3D velocity model used in impulse response tests are set as $x \times y \times z = 640\,\mathrm{m} \times 640\,\mathrm{m} \times 640\,\mathrm{m}$. Grid sizes in three coordinates are the same as each other. i.e., $dx = dy = dz = 10\,\mathrm{m}$. Figure 6.11 shows the dimensions of the model.

3D impulse tests are started with a simple impulse test. In this simple case, a single impulse is imparted to a homogeneous medium with dimensions shown in Figure 6.11 and a constant velocity of 2000 m/s. The impulse (point source) is seeded at $(x, y, z) = (320, 320, 0)$, i.e., centre of the top surface of the model. A zero-phase seismic wavelet

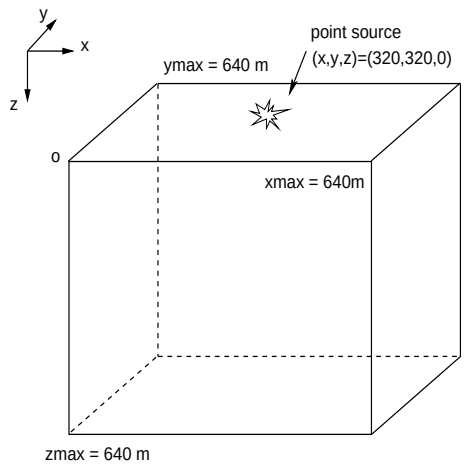


Figure 6.11: Modelling geometry and the velocity model: the point source is positioned at the centre of the surface.

(Ormsby) with frequencies ranging from 5 Hz to 60 Hz is used as the impulse in the wavefield extrapolation.

Starting from the surface, the source field is extrapolated at depths with a step size of 10 m. In this example and the following ones, the source field (from one or more point sources) has been extrapolated 100 m downwards from the surface, using 10 extrapolation steps. Snapshots are taken every 2 steps (20 m apart). Figure 6.12 shows 3D-view snapshots of impulse responses at various depths. The first one (Figure 6.12 (a)) shows the point source before extrapolation. As extrapolated downwards at depths, the shape of the point source gradually becomes a hyperbolic cone shape, expanding both in time (from reader into the page) and the lateral directions (x, y). Figure 6.12 (a) to (f) show the proceeding of the source field extrapolated by the 3D Gabor wavefield extrapolator in the homogeneous medium. In terms of time, difference between apexes of each pair of adjacent hyperbolic cones is 10 ms (calculated from 20 m / 2000 m/s).

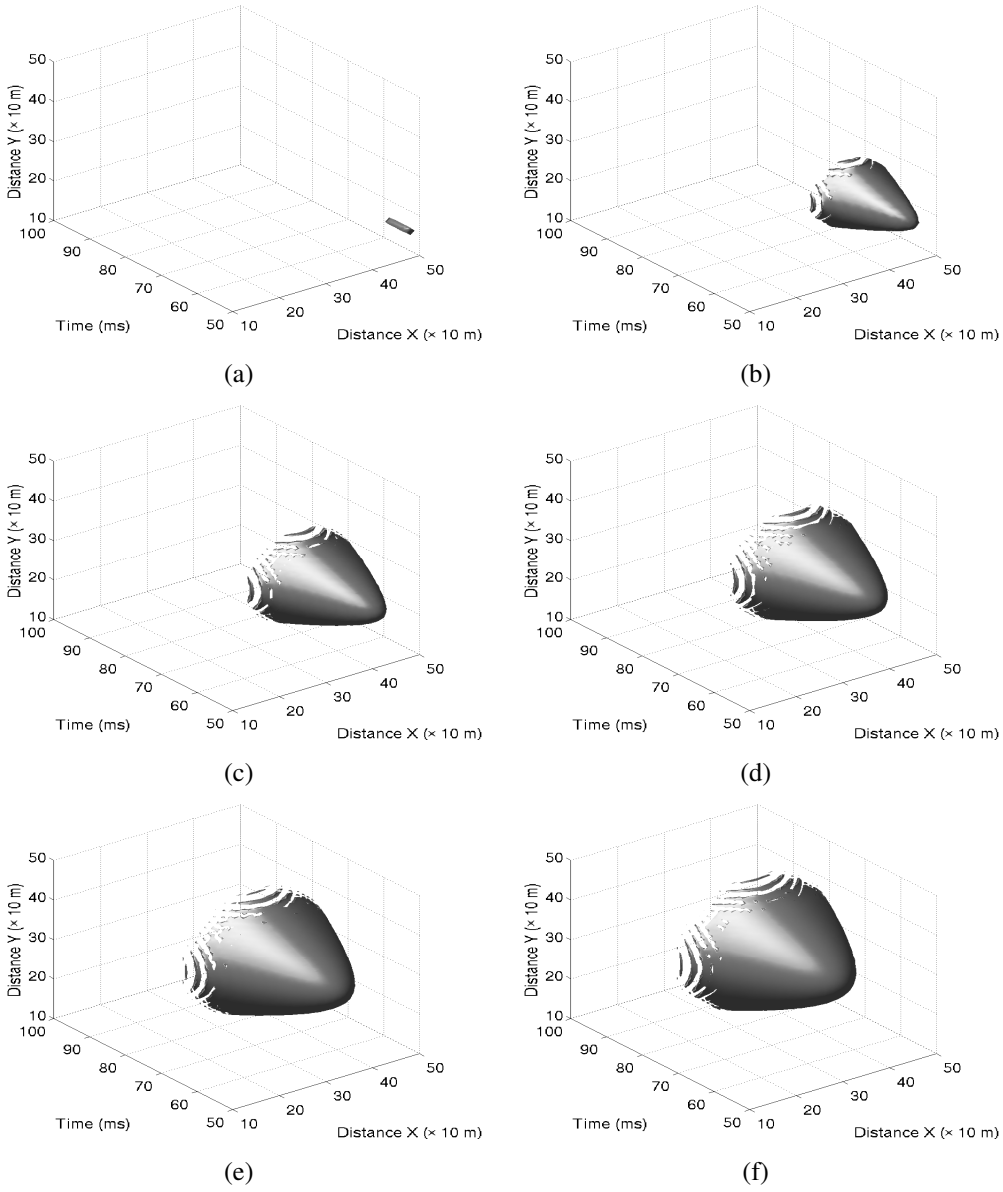


Figure 6.12: 3D impulse responses in a homogeneous medium. (a) Impulse before extrapolation. Impulse responses when extrapolated from surface to depths: (b) 20 m, (c) 40 m, (d) 60 m, (e) 80 m and (f) 100 m.

To see the time progression of impulse responses more clearly, we can take vertical profiles through the source position in x direction, for example. Figure 6.13 shows the vertical profiles corresponding to the 3D snapshots of the extrapolated impulse shown in Figure 6.12. Careful inspection on Figure 6.13 tells us that the time difference between two adjacent apexes of hyperbolae is 10 ms (e.g., compare Figure 6.13 (c) and (d)).

More details can be seen using vertical profiles with varying offsets from the point source. The extrapolated impulse shown in Figure 6.12 (f) is selected to created 2D vertical profiles with varying offsets. Also, time slices are taken to show the wavefield propagating in the horizontal directions. Figure 6.14 shows these profiles and slices.

In the second impulse response example, the velocity model remains the same one as in the first example; and five impulses are used instead of one. The impulse used in the previous example stays at $(x, y, z) = (320, 320, 0)$ as one of the five impulses. This impulse, referred to as the central impulse, is surrounded by four identical impulses, each of which is 200 meters away from the central one either in the x or y direction on the top surface. Figure 6.15 shows 3D-view impulse responses at depths. We can see from these impulse responses that these five point sources move at the same pace as each other, as expected. This test also tells us that the 3D wavefield extrapolator is able to handle many impulses at the same time.

Figure 6.16 shows the corresponding 2D vertical profiles through the central impulse in the x direction. In this case, we always see three extrapolated impulses (see Figure 6.16 (a)). If we take a vertical profile in y direction through the central impulse, we can also see three impulses. The central impulse sits at the cross point of the two vertical profiles, including those four impulses surrounding it.

Also, we can see that starting from Figure 6.16 (c) to (f), there is a hyperbola under the

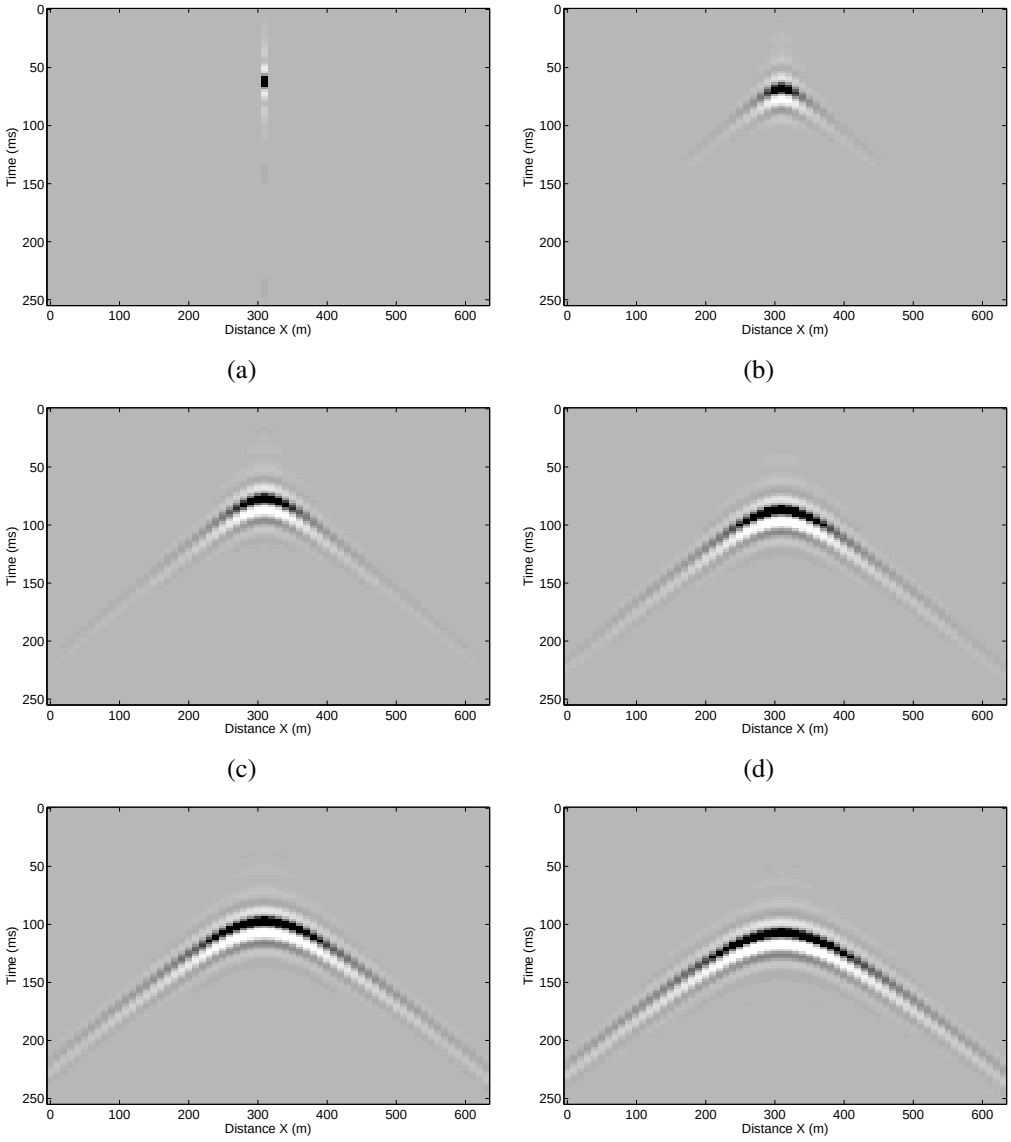


Figure 6.13: 2D vertical profiles through the point source ($y = 0$) taken from 3D impulse responses shown in Figure 6.12. (a) Impulse before extrapolation. Impulse responses when extrapolated from surface to depths: (b) 20 m, (c) 40 m, (d) 60 m, (e) 80 m and (f) 100 m.

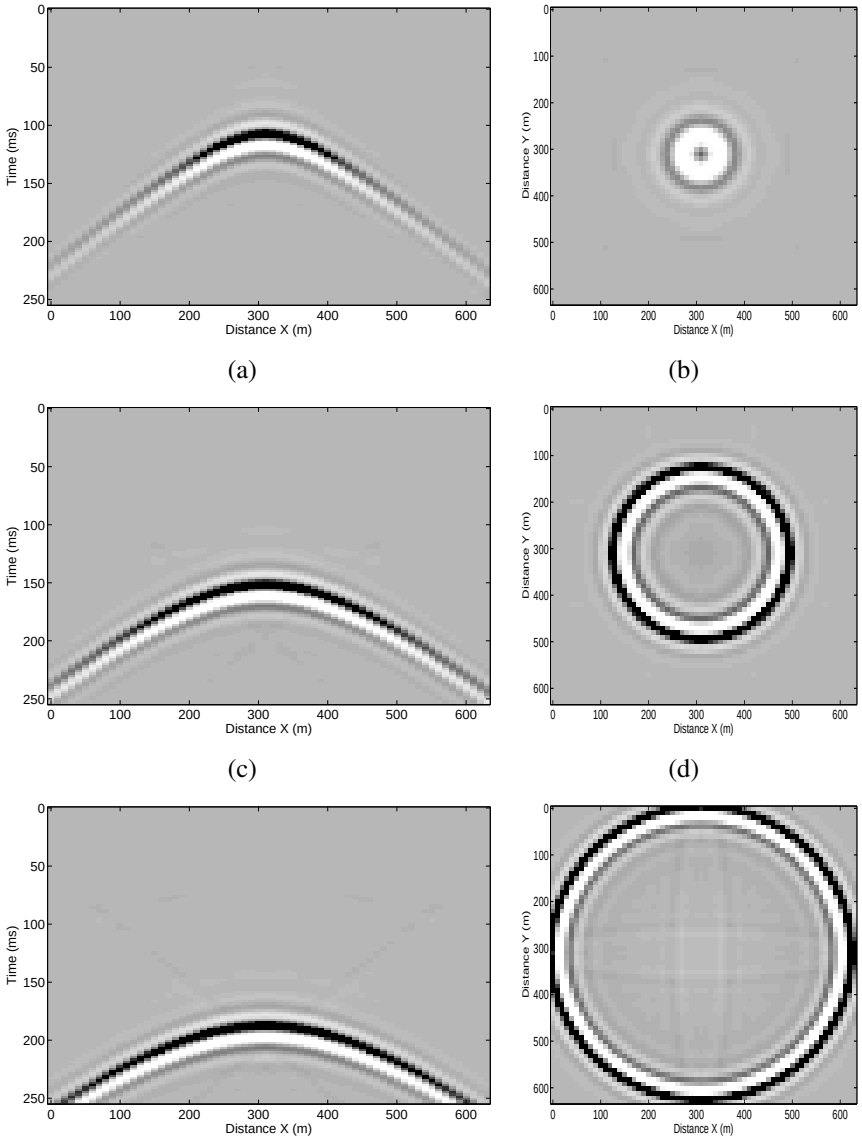


Figure 6.14: 2D vertical profiles with varying offsets from the source point and time slices at various times from 3D impulse response shown in Figure 6.12 (f). (a) Vertical profile at offset 0 m. (b) Time slice at 52 ms. (c) Vertical profile at offset 160 m. (d) Time slice at 82 ms. (e) Vertical profile at offset 240 m. (f) Time slice at 112 ms.

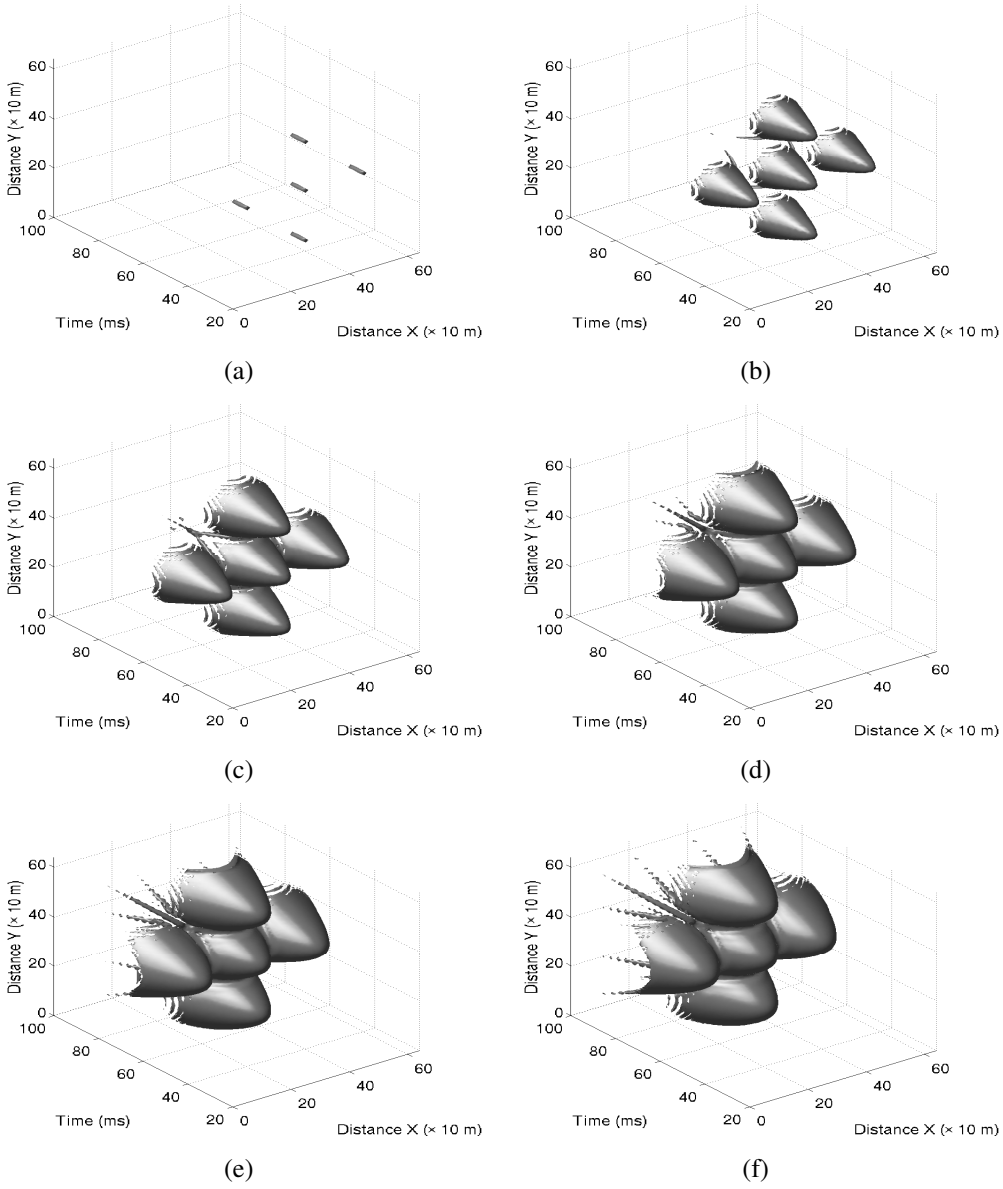


Figure 6.15: 3D impulse responses using five impulses in a homogeneous medium. (a) Impulses before extrapolation. Impulse responses at depths: (b) 20 m, (c) 40 m, (d) 60 m, (e) 80 m and (f) 100 m.

central point in each profile. This is the one related to the response from the side impulses in the y direction (see Figure 6.15). Since the responses from the two side impulses in the y direction coincide with each other, they overlap as one.

Similarly, vertical profiles at various offsets from the central point source and time slices at various times from the impulses shown in Figure 6.15 (f) are created in this example. Figure 6.17 shows these vertical profiles and time slices. The centres of those circles in the time slices also tell the positions of those five impulses. These time slices show expanding wavefronts with time in the lateral directions.

In the third impulse response test, five impulses used in the second example are still employed. A more complex velocity model (with the same dimensions as shown in Figure 6.11) is used in this example. That is, there are lateral variations in velocity. In the central part, a region with a velocity of 2000 m/s is surrounded by a region with a velocity of 4000 m/s. Figure 6.18 shows a horizontal slice from the 3D velocity model. There is no velocity variation in the z direction in this model.

Figure 6.19 shows the propagation of the impulses extrapolated at depths. After a few steps of extrapolation, we can see that the central impulse, originally sitting in a region of a low velocity (2000 m/s), departs from other impulses around it. This is expected since this impulse is extrapolated at a lower velocity.

More details about the hyperbolic shapes related to the 3D impulse responses can be seen from the vertical profiles corresponding to the 3D-view impulse response shown in Figure 6.19.

Figure 6.20 (a) presents a 2D vertical profile corresponding to Figure 6.19 (f), at a zero offset from the central point source in the x direction. A similar one taken from Figure 6.15 (f) is shown in Figure 6.20 (b) for comparison.

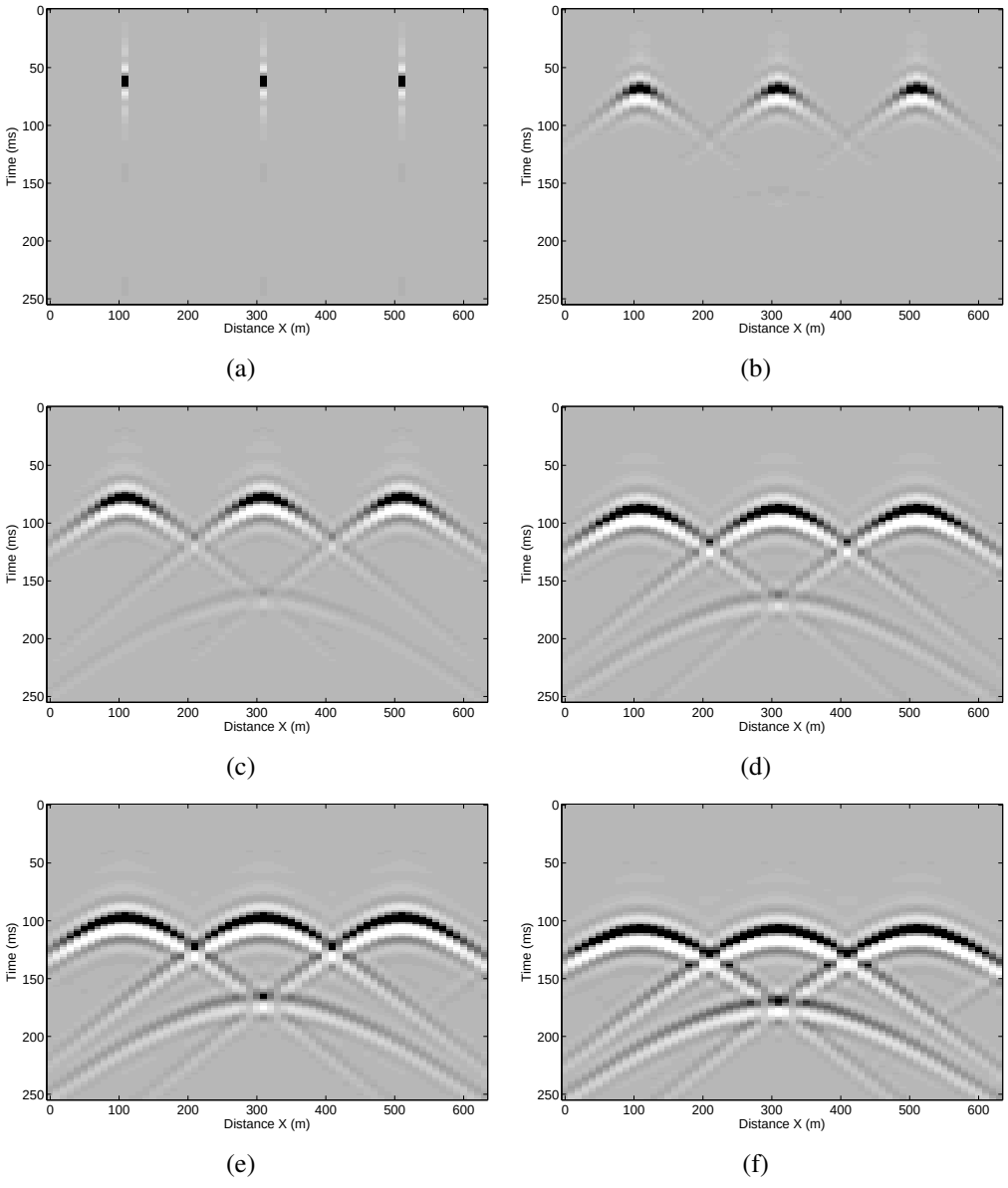


Figure 6.16: 2D vertical profiles through the central point source ($y = 0$) taken from 3D impulse responses shown in Figure 6.15. (a) Impulses before extrapolation. Impulse responses when extrapolated from surface to depths: (b) 20 m, (c) 40 m, (d) 60 m, (e) 80 m and (f) 100 m.

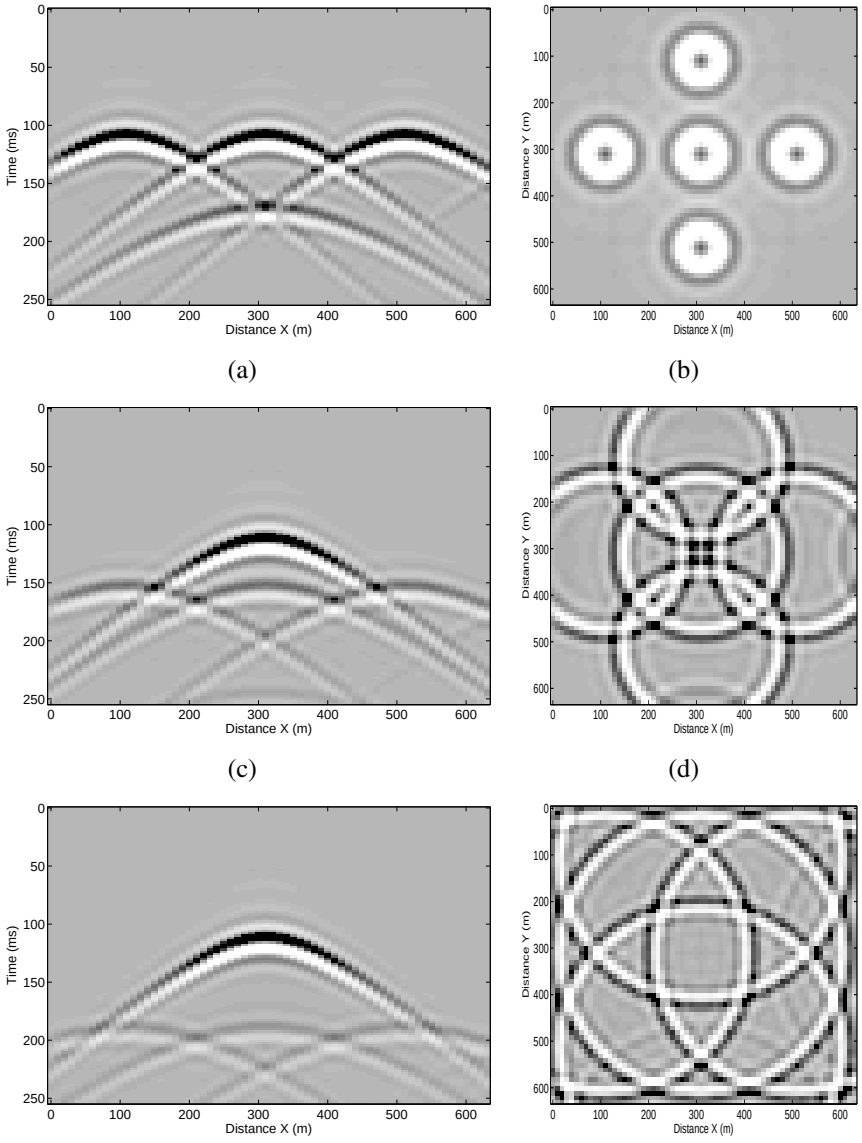


Figure 6.17: 2D vertical profiles at varying offsets from the central source point and time slices at various times from 3D impulse responses shown in Figure 6.15 (f). (a) Vertical profile at offset 0 m. (b) Time slice at 52 ms. (c) Vertical profile at offset 160 m. (d) Time slice at 82 ms. (e) Vertical profile at offset 240 m. (f) Time slice at 112 ms.

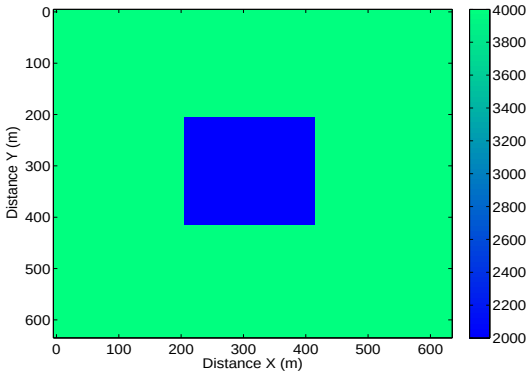


Figure 6.18: A horizontal slice from the 3D velocity volume. In the x direction, v = 2000 m/s, when $x \in$(210,420); otherwise, v = 4000 m/s. In the y direction, v = 2000 m/s, when $y \in$(210,420); otherwise, v = 4000 m/s.

We can see that in the central regions of Figure 6.20 (a) and (b) (both with a velocity of 2000 m/s), the responses are very close to each other. Outside of the central region ($x \in [210, 420]$), the impulse responses are different from each other due to the velocity changing (compare arrival times in Figures 6.20 (a) and (b)).

From the impulse response examples shown in this section, we can see that the 3D Gabor extrapolator works correctly and well either in homogeneous media or in simple inhomogeneous media. The 3D extrapolator can also handle multiple impulses at the same time.

The initial tests of the 3D Gabor wavefield extrapolator and the 2D adaptive partitioning algorithm have shown a promising application to 3D depth migrations.

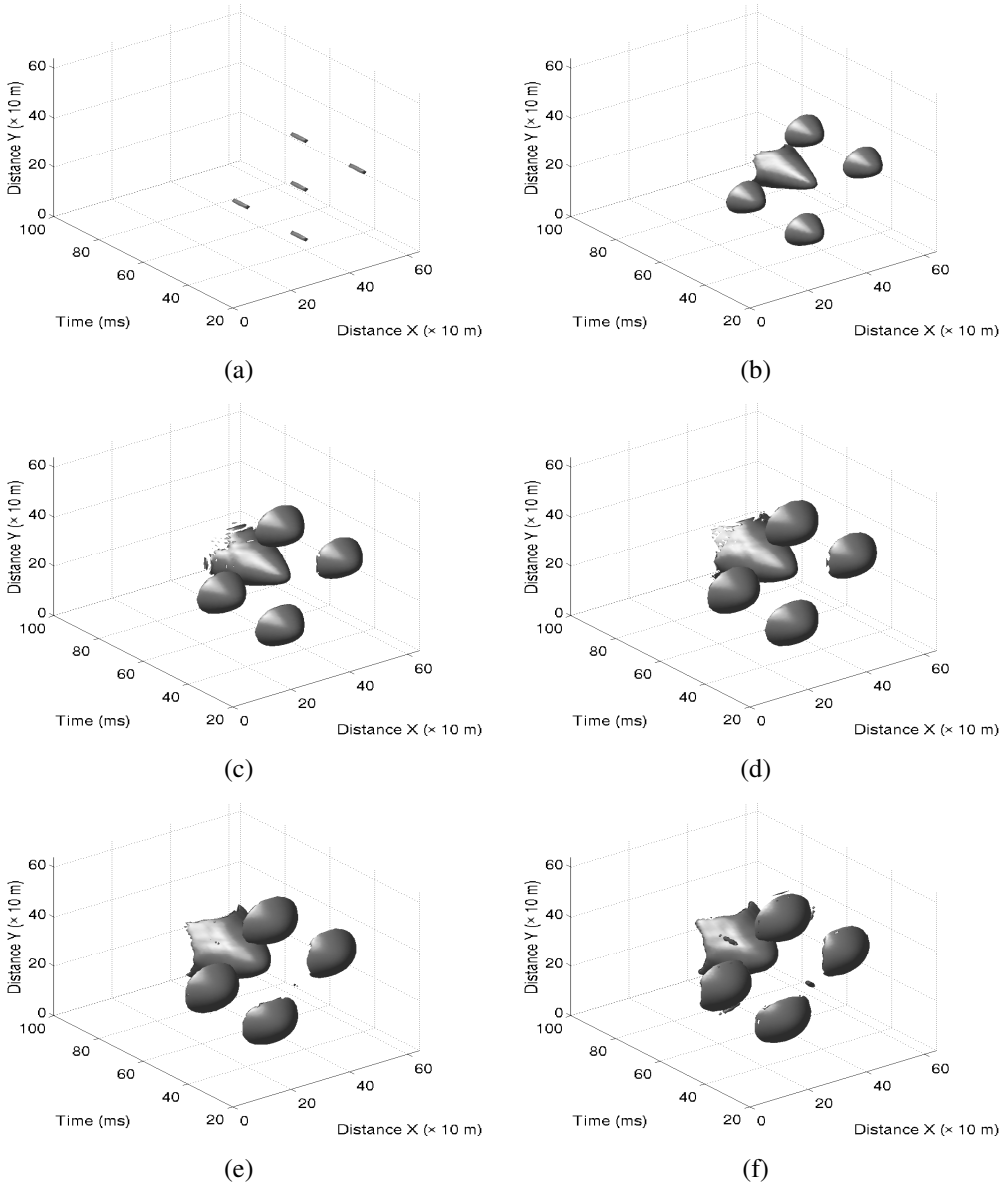
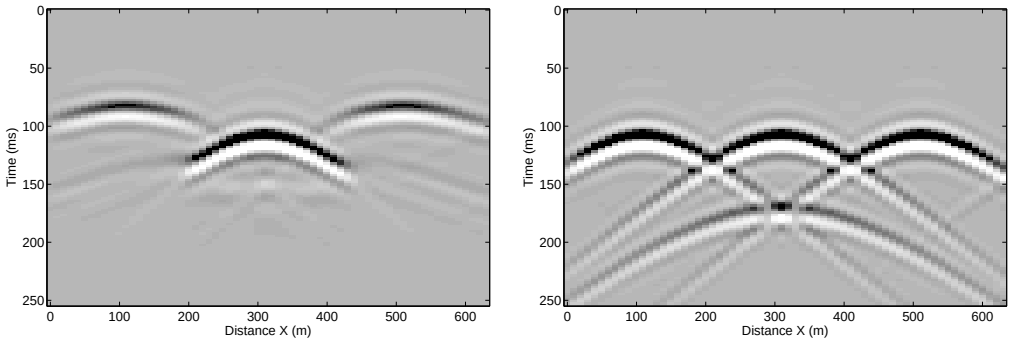


Figure 6.19: 3D impulse responses in an inhomogeneous medium with two distinct velocity regions indicated in Figure 6.18. Notice that the central impulse responses depart from the ones surrounding it.



(a) (b)

Figure 6.20: 2D vertical profiles from 3D impulse responses at a zero offset from the central point source (a) in an inhomogeneous model and (b) In a homogeneous model.

6.6 Chapter summary

This chapter started with the introduction on 3D wavefield extrapolation using the phase shift method. Then the 3D Gabor wavefield extrapolation theory was described as one of those phase-shift methods for wavefield extrapolation in 3D.

To obtain an efficient 3D Gabor wavefield extrapolation, a 2D adaptive partitioning algorithm was derived using the lateral position error as the partitioning criterion, which was extended from the corresponding 1D partitioning algorithm developed in Chapter 3. This partitioning algorithm can be extended into higher dimensions without much difficulty if needed.

Examples given by the 2D adaptive partitioning algorithm have shown that the 2D partitioning method works in a correct way as expected. Most importantly, it can handle adaptive partitioning in complex velocity models. This method is key to an efficient 3D Gabor wavefield extrapolation.

Impulse response tests using the 3D Gabor wavefield extrapolator show that the 3D

wavefield extrapolator can be used for 3D depth imaging.

In the next chapter. I will conclude my treatment of the Gabor depth imaging theory and its applications in seismic depth imaging as related throughout the thesis.

Chapter 7: Conclusions and Future Directions

7.1 Conclusions

In the previous chapters, Gabor depth imaging theories have been thoroughly explored. Imaging examples using the Gabor wavefield extrapolation theories have demonstrated a very good depth imaging method. Based on the description of these theories and the imaging examples, I conclude in the following sections.

7.1.1 Gabor depth imaging theories

Development of Gabor depth imaging theories were originally initiated by Grossman et al. (2002a,b). These theories have been fully explored both in 2D and in 3D in this thesis (see Chapters 2 and 6). One of the most innovative contributions to the Gabor depth imaging theories in this thesis is that an adaptive positioning algorithm has been invented. The new partitioning algorithm uses the lateral position error as a criterion to create adaptive partitions, thus it can be used to trade off between imaging accuracy and efficiency. The adaptive partitioning algorithm makes a fast Gabor depth imaging method possible; and the partitioning algorithm is very straightforward to extend from 1D (for 2D depth imaging) to a higher dimension (e.g., 2D for 3D imaging) compared to other adaptive partitioning methods described in this thesis. Also, with the integration of spatial resampling techniques in Gabor depth imaging theories, a further improvement in imaging efficiency is achieved (see Chapter 3).

Another important development in Gabor depth imaging theories is to generalize the Gabor transform by using intermediate values of p between 0 and 1 (see Chapter 2). Pa-

rameter p gives infinite ways of freedom in the Gabor transform, resulting in infinitely many Gabor depth imaging schemes in addition to two extreme cases when $p = 1$ (simulating the NSPS) and $p = 0$ (simulating the PSPI).

7.1.2 Accuracy and efficiency of the Gabor depth migration

The Gabor depth imaging method can be used as an accurate wave equation depth migration method. Not only is it accurate, but it can be efficient also. This depth imaging method has been developed as an efficient depth migration method; it can be improved tens of times faster by using adaptive partitioning algorithms during the wavefield extrapolation with minimal loss of accuracy. The 2D imaging examples (e.g., from the Marmousi data set) have shown high quality images with acceptable run time to obtain them.

The adaptive Gabor imaging method provides us with the flexibility of choosing between imaging speed and imaging quality. As shown in Chapter 3, the efficiency improvement due to adaptive partitioning comes from a reduction of unnecessary partitions, given certain accuracy criteria. Since the computational cost is proportional to the number of partitions, Gabor depth imaging method using adaptive partitioning algorithms thus gives faster imaging processes when the number of partitions is reduced.

Gabor depth imaging can be made even faster by using spatial resampling during recursive wavefield extrapolations (see theories in Chapter 3 and examples in Chapter 4). Spatial resampling obtains an advantage from needing fewer spatial sampling points at low frequencies, which gives shorter spatial Fourier transforms. A shorter Fourier transform means a faster process. Therefore, Gabor wavefield extrapolation can be accelerated at low frequencies. Since Gabor wavefield extrapolation is performed in the frequency domain, spatial resampling helps to save a significant amount of calculation.

7.1.3 2D Gabor depth migration results

Depth imaging results from the Marmousi seismic data show that Gabor depth imaging method can produce high-quality images. Given an appropriate accuracy criterion, imaging quality can be fairly good while keeping imaging speed acceptable. In the Marmousi depth imaging, we can set lateral position errors as large as 5 m and 10 m and still obtain good imaging qualities but use only 3/4 and 1/2 run time compared to the one obtained by using a higher accuracy criterion (e.g., a lateral position error of 2.5 m in the LPEAP algorithm, see Figure 4.9 in section 4.5, Chapter 3).

Gabor depth imaging method also works for seismic data acquired from topographic terrains. Depth imaging results from the Model 94 data set have shown images with very high quality. One of the most important achievements of the Gabor imaging with Model 94 is that the near surface structures (usually hard to image) are clearly imaged. The application in a real 2D topographic data set also gives very reasonable imaging results based on a constant velocity model.

The depth imaging results from the 2D data sets (from either flat or topographic surface) shows that the Gabor depth imaging method is a very competitive wave equation migration method.

7.1.4 3D Gabor depth migration tests

3D seismic data have been becoming dominant in current industry data libraries. Thus all depth imaging methods should eventually come to applications in 3D. The Gabor depth imaging method is not exempt from this. To have a reasonable run time with acceptable accuracy in 2D imaging, adaptive partitioning algorithms are needed; this is also the case in 3D Gabor depth imaging.

Adaptive partitioning is critically important in 3D Gabor depth imaging because we deal with many 2D spatial Fourier transforms during a 3D Gabor depth imaging process, and because large amounts of data are involved in 3D depth migrations. In Chapter 6, the 3D adaptive Gabor depth imaging method uses a 2D lateral-position-error adaptive partitioning algorithm, which will be very useful in 3D Gabor depth imaging method. 2D partitioning examples in Chapter 6 have clearly shown that the 2D adaptive partitioning algorithm is capable of creating partitions adapted to 2D velocity slices with complex variations.

Another initial experiment is the impulse test before performing the 3D Gabor depth imaging method. In Chapter 6, the impulse response with a 3D Gabor wavefield extrapolator is tested. The impulse responses have shown that 3D Gabor wavefield extrapolator behaves as expected and is ready for 3D Gabor wavefield extrapolation.

7.2 Future directions

Gabor depth imaging theories have been well developed. This imaging method has produced high-quality 2D images from seismic data acquired from both flat and topographic surfaces. However, there are still aspects that may be worth exploring. Some ideas to be introduced may be useful to improve imaging efficiency further or give better imaging results compared to those obtained in this thesis. An immediate extension to the 3D Gabor imaging (Chapter 6) will be 3D Gabor depth imaging with any 3D seismic data set. My suggestions to the future work on Gabor depth imaging method are listed in the following sections.

7.2.1 Adaptive partitioning algorithms

Three adaptive partitioning methods has been introduced in Chapter 3. Among them, the LPEAP algorithm is the best one and was used in the 2D depth imaging examples in Chapters 4 and 5. As shown in Chapter 3, the LPEAP algorithm gives better partitioning results and is easier to extend to high dimensions (e.g., 2D in Chapter 6).

It is possible that the LPEAP algorithm can be made more elegant by using clustering algorithms to find reference velocities. That is, instead of starting to search from a central (or called the most frequent velocity in Chapter 3) reference velocity to build a complete set of reference velocities, the search process can be initiated from more than one centre, each of which is the centre of a cluster [1]. Of course, the suggested method may not work quite well or it can be very complicated and slow down the partitioning process. However, it may be worthy of testing.

7.2.2 3D Gabor depth migration

One immediate application in 3D Gabor depth imaging method developed in Chapter 6 is to migrate a 3D synthetic or real data set. Also, 3D Gabor depth imaging method may be made available for 3D depth migrations with topographic seismic data.

7.3 Final remarks

The Gabor imaging theories have been developed to a new level by adding in the new adaptive partitioning algorithms and so on, and they have been more thoroughly explored in the application side in this thesis. With the significant improvement in imaging effi-

[1] A cluster is a group of velocities that are closely gathered.

ciency, this imaging method shows a potential wider range of applications in the seismic depth imaging. It is encouraging to bring this research to new areas in either theories or practical applications.

Bibliography

Archer, J., 2007, Palinspastic Restoration of Velocity Models Used in Prestack Depth Migration: Example from Northeast British Columbia: M. Sc. thesis, University of Calgary.

Arfken, G., 1970, Mathematical methods for physicists: Academic Press.

Audebert, F., D. Nicholsz, T. Rekdal, B. Biondi, D. E. Lumley, and H. Urdaneta, 1999, Imaging complex geologic structure with single-arrival Kirchhoff prestack depth migration: Geophysics, **62**, 1533–1543.

Baker, B. B. and E. T. Copson, 1987, The mathematical theory of huygens principle: AMS Bookstore.

Baysal, E., D. Kosloff, and J. W. C. Sherwood, 1983, Reverse time migration: Geophysics, **48**, 1514–1524.

Beasley, C. and W. Lynn, 1992, The zero-velocity layer: Migration from irregular surfaces: Geophysics, **57**, 1435–1443.

Bednar, J. B., 2005, A brief history of seismic migration: Geophysics, **70**, 3MJ–20MJ.

Berkhout, A. J., 1981, Wave field extrapolation techniques in seismic migration, a tutorial: Geophysics, **46**, 1638–1656.

Bevc, D., 1997, Flooding the topography: Wave-equation datuming of land data with rugged acquisition topography: Geophysics, **62**, 1558–1569.

Biondi, B., 2002, Prestack imaging of overturned and prismatic reflections by reverse time migration: SEP-111, 123–139.

Bourgeois, A., M. Bourget, P. Lailly, M. Poulet, P. Ricarte, and R. Versteeg, 1991, Marmousi, Model and data, *in* R. Versteeg and G. Grau, eds., The Marmousi experience:

Proceedings of the 1990 EAGE workshop on practical aspects of seismic data inversion, 5–16.

Brown, R. J. S., 1969, Normal-moveout and velocity relations for flat and dipping beds and for long offset: Geophysics, **34**, 180–195.

Červený, V., 2001, Seismic ray theory: Cambridge University Press.

Chang, W.-F. and G. A. McMechan, 1987, Elastic reverse-time migration: Geophysics, **52**, 1365–1375.

———, 1994, 3-D elastic prestack, reverse-time depth migration: Geophysics, **59**, 597–609.

Claerbout, J. F., 1971, Toward a unified theory of reflector mapping: Geophysics, **36**, 467–481.

Claerbout, J. F. and S. M. Doherty, 1972, Downward continuation of moveout-corrected seismograms: Geophysics, **37**, 741–768.

Clayton, R. W. and B. Engquist, 1980, Absorbing boundary conditions for wave-equation migration: Geophysics, **45**, 895–904.

Courant, R. and D. Hilbert, 1962, Methods of Mathematical Physics, Partial Differential Equations (Volume 2): John Wiley & Sons.

de Bazelaire, E., 1988, Normal moveout revisited: Inhomogeneous media and curved interfaces: Geophysics, **53**, 143–157.

Dong, Z. and G. A. McMechan, 1993, 3-D prestack migration in anisotropic media: Geophysics, **58**, 79–90.

Etgen, J. T., 1994, Stability analysis of explicit depth extrapolation through laterally-varying media: 64th Annual International Meeting, SEG, Expanded Abstracts, 1266–1269.

Fei, T. W. and C. L. Liner, 2008, Hybrid Fourier finite-difference 3D depth migration for anisotropic media: Geophysics, **73**, S27–S34.

Ferguson, R. J. and G. F. Margrave, 2005, Planned seismic imaging using explicit one-way operators: Geophysics, **70**, S101–S109.

Fishman, L. and J. J. McCoy, 1985, A new class of propagation models based on a factorization of the Helmholtz equation: Geophysics Journal of the Royal Astronomical Society, **80**, 439–461.

French, W. S., 1974, Two-dimensional and three-dimensional migration of model-experiment reflection profiles: Geophysics, **39**, 265–277.

———, 1975, Computer migration of oblique seismic reflection profiles: Geophysics, **40**, 961–980.

Gazdag, J., 1978, Wave equation migration with the phase-shift method: Geophysics, **43**, 1342–1351.

Gazdag, J. and P. Sguazzero, 1984, Migration of seismic data by phase shift plus interpolation: Geophysics, **49**, 124–131.

Geoltrain, S. and J. Brac, 1993, Can we image complex structures with first-arrival traveltime?: Geophysics, **58**, 564–575.

Gray, S. H., J. Etgen, J. Dellinger, and D. Whitmore, 2001, Y2k review article seismic migration problems and solutions: Geophysics, **66**, 1622–1640.

Gray, S. H. and K. J. Marfurt, 1995, Migration from topography: Improve the near-surface image: Canadian Jorunal of Exploration Geophysics, **31**, 18–24.

Gray, S. H. and W. P. May, 1994, Kirchhoff migration using eikonal equation traveltimes: Geophysics, **59**, 810–817.

Gröchenig, K., 2001, Foundations of Time-Frequency Analysis: Birkhäuser.

Grossman, J. P., G. F. Margrave, and M. P. Lamoureux, 2002a, Constructing adaptive, nonuniform Gabor frames from partitions of unity: CREWES Research Report, **14**.

———, 2002b, Fast wavefield extrapolation by phase-shift in the nonuniform Gabor domain: CREWES Research Report, **14**.

Hagedoorn, J. G., 1954, A process of seismic reflection interpretation: Geophysical Prospecting, **06**, 449–453.

Hale, D., 1984, Dip-moveout by Fourier transform: Geophysics, **49**, 741–757.

———, 1991, Stable explicit depth extrapolation of seismic wavefields: Geophysics, **56**, 1770–1777.

Hale, D. and J. F. Claerbout, 1983, Butterworth dip filters: Geophysics, **48**, 1033–1038.

Ji, J. and B. Biondi, 1991, Depth Migration by an Unconditionally Stable Explicit Finite-Difference Method: SEG Expanded Abstract, 1122–1125.

Jin, S. and R. Wu, 1998, Depth migration using the windowed generalized screen propagators: 68th Annual International Meeting, SEG, Expanded Abstracts, 1843–1846.

Jin, S., S. Xu, and C. C. Mosher, 2002, Migration with a local phase screen propagator: 72th Annual International Meeting, SEG, Expanded Abstracts, 1164–1167.

Kohn, J. and L. Nirenberg, 1965, An algebra of pseudo-differential operators: Commun. Pure Appl. Math., **18**, 269–305.

Kosloff, D. D. and E. Baysal, 1983, Migration with the full acoustic wave equation: Geophysics, **48**, 677–687.

Lee, D., lain M. Mason, and G. M. Jackson, 1991, Split-step Fourier shot-record migration with deconvolution imaging: Geophysics, **56**, 1786–1793.

Li, Z., 1991, Compensating finite-difference errors in 3-D migration and modeling: Geophysics, **56**, 1650–1660.

Ma, Y. and G. F. Margrave, 2005, Prestack depth migration with the Gabor transform: CREWES Research Report, **17**.

———, 2006a, Adaptive partitioning for 3D Gabor wavefield extrapolation: CREWES Research Report, **18**.

———, 2006b, Prestack depth imaging with the Gabor transform: 76th Annual International Meeting, SEG, Expanded Abstracts, 2504–2507.

———, 2006c, Prestack depth migration with the Gabor transform: CSEG National Conference Abstracts.

———, 2007a, Adaptive Gabor imaging using the lateral position error criterion: 77th Annual International Meeting, SEG, Expanded Abstracts.

———, 2007b, Fast Gabor imaging with the spatial resampling: CREWES Research Report, **19**.

———, 2008, Seismic depth imaging with the Gabor transform: Geophysics, **73**, S91–S97.

Margrave, G. and Z. Yao, 2000, Downward extrapolation from topography with a laterally variable depth step: 2000 Annual International Meeting, SEG, Expanded Abstracts.

Margrave, G. F., S. M. Al-Saleh, H. D. Geiger, and M. P. Lamoureux, 2004, The FOCI algorithm for seismic depth migration: CREWES Research Report, **16**.

Margrave, G. F. and R. J. Ferguson, 1999, Wavefield extrapolation by nonstationary phase shift: Geophysics, **64**, 1067–1078.

Margrave, G. F., H. D. Geiger, S. M. Al-Saleh, and M. P. Lamoureux, 2006, Improving explicit seismic depth migration with a stabilizing wiener filter and spatial resampling: Geophysics, **71**, S111–S120.

Margrave, G. F. and M. P. Lamoureux, 2001, Gabor deconvolution: CREWES Research Report, **13**.

Miller, D., M. Oristaglio, and G. Beylkin, 1987, A new slant on seismic imaging: Migration and integral geometry: Geophysics, **52**, 943–964.

Pann, K., Y. Shin, and E. Eisner, 1979, A collocation formulation of wave equation migration: Geophysics, **44**, 712–721.

Papoulis, A., 1962, The Fourier Integral and its Applications: McGraw-Hill Book company, Inc.

Reshef, M., 1991, Depth migration from irregular surfaces with depth extrapolation methods: Geophysics, **56**, 119–122.

Ristow, D. and T. Rühl, 1997, 3-D implicit finite-difference migration by multiway splitting: Geophysics, **62**, 554–567.

Roberts, P., L. Huang, C. Burch, M. Fehler, and S. Hildebrand, 1997, Prestack depth migration for complex 2D structure using phase-screen propagators: 67th Annual International Meeting, SEG, Expanded Abstracts, 1282–1285.

Rousseau, J. and M. V. de Hoop, 2001, Modeling and imaging with the scalar generalized-screen algorithms in isotropic media: Geophysics, **66**, 1551–1568.

Schleicher, J., J. C. Costa, and A. Novais, 2008, Time-migration velocity analysis by image-wave propagation of common-image gathers: Geophysics, **73**, VE161–VE171.

Schneider, W. A., 1971, Developments in seismic data processing and analysis (1968-1970): Geophysics, **36**, 1043–1073.

———, 1978, Integral formulation for migration in two and three dimensions: Geophysics, **43**, 49–76.

Schultz, P. S. and J. W. C. Sherwood, 1980, Depth migration before stack: Geophysics, **45**, 376–393.

Sneddon, I. N., 1951, Fourier Transforms: McGraw-Hill Book Company, Inc.

Stoffa, P. L., J. T. Fokkema, R. M. de Luna Freire, and W. P. Kessinger, 1990, Split-step Fourier migration: Geophysics, **55**, 410–421.

Stolt, R. H., 1978, Migration by Fourier transform: Geophysics, **43**, 23–48.

Valenciano, A. A., 2002, Damped imaging condition for reverse-time migration: SEP-112, 117–123.

Versteeg, R., 1994, The Marmousi experience: Velocity model determination on a synthetic complex data set: The Leading Edge, **13**, 927–936.

Wu, R. and L. Huang, 1992, Scattered calculation in heterogeneous media using a phase-screen propagator: 62th Annual International Meeting, SEG, Expanded Abstracts, 1289–1292.

Yilmaz, O., 2001, Seismic Data Analysis: Soc. Expl. Geophys.